kita Siapkan dulu sebuah tools sebagai berikut:

a. xampp
b. CodeIgniter_2.1.4
c. Browser
d. Sublime Text

Yang pertama kita install terlebih dahulu xampp setelah selesai install xampp dan sublime text proses selanjutnya extrak file CodeIgniter_2.1.4 didalam file htdoc setelah dinstall proses selanjutnya kita bikin sebuah aplikasi webnya di application proses selanjutnya dibikin sebuah kofigurasi data base dan konfigurasi url berikut ini cara konfigurasi database dan konfigurasi url didalam folder config untuk konfigurasi sebuah data base dan konfigurasi sebuah url yang akan dipanggil di browser tersebut.

Berikut ini sorce code yang akan untuk mengkoneksikan sebuah data base sebagai berikut ini:

Pada gambar 1.1 yang harus kita buat untuk koneksi data basenya.

autoload	2/16/2015 3:37 PM	PHP File	4 KB
config	11/28/2014 11:41 ...	PHP File	13 KB
constants	7/29/2013 3:54 PM	PHP File	2 KB
database	2/12/2015 12:00 PM	PHP File	4 KB
doctypes	7/29/2013 3:54 PM	PHP File	2 KB
foreign_chars	7/29/2013 3:54 PM	PHP File	2 KB
hooks	7/29/2013 3:54 PM	PHP File	1 KB
index	7/29/2013 3:54 PM	HTML Document	1 KB
migration	7/29/2013 3:54 PM	PHP File	2 KB
mimes	7/29/2013 3:54 PM	PHP File	5 KB
profiler	7/29/2013 3:54 PM	PHP File	1 KB
routes	7/29/2013 3:54 PM	PHP File	2 KB
smileys	7/29/2013 3:54 PM	PHP File	4 KB
user_agents	7/29/2013 3:54 PM	PHP File	6 KB

Gambar 1.1

Nama File database.php

```
$active_group = 'default';
$active_record = TRUE;
$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'root';
$db['default']['password'] = '';
$db['default']['database'] = 'roti';
$db['default']['dbdriver'] = 'mysql';
$db['default']['dbprefix'] = '';
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = '';
$db['default']['char_set'] = 'utf8';
$db['default']['dbcollat'] = 'utf8_general_ci';
$db['default']['swap_pre'] = '';
$db['default']['autoinit'] = TRUE;
$db['default']['stricton'] = FALSE;
```

Konfigurasi sebuah urlnya sebagai berikut ini cara konfigurasi sebuah urlnya

autoload	2/16/2015 3:37 PM	PHP File	4 KB
config	11/28/2014 11:41 ...	PHP File	13 KB
constants	7/29/2013 3:54 PM	PHP File	2 KB
database	2/12/2015 12:00 PM	PHP File	4 KB
doctypes	7/29/2013 3:54 PM	PHP File	2 KB
foreign_chars	7/29/2013 3:54 PM	PHP File	2 KB
hooks	7/29/2013 3:54 PM	PHP File	1 KB
index	7/29/2013 3:54 PM	HTML Document	1 KB
migration	7/29/2013 3:54 PM	PHP File	2 KB
mimes	7/29/2013 3:54 PM	PHP File	5 KB
profiler	7/29/2013 3:54 PM	PHP File	1 KB
routes	7/29/2013 3:54 PM	PHP File	2 KB
smileys	7/29/2013 3:54 PM	PHP File	4 KB
user_agents	7/29/2013 3:54 PM	PHP File	6 KB

Gambar 1.2

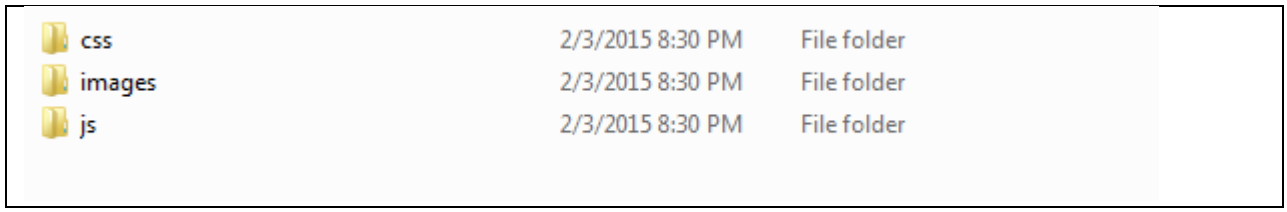
```
$config['base_url']   = 'http://localhost/toko-roti/';
```

Selanjutnya proses pembuatan tampilan terlebih dahulu proses pembuatan tampilannya di folder assets digunakan untuk membuat sebuah tampilan css dan penyimpanan gambar data produk roti dan jquerynya.

application	2/16/2015 10:38 PM	File folder	
assets	2/3/2015 8:30 PM	File folder	
system	2/3/2015 8:30 PM	File folder	
user_guide	2/3/2015 8:30 PM	File folder	
.gitignore	2/16/2015 3:37 PM	GITIGNORE File	1 KB
.htaccess	2/16/2015 3:37 PM	HTACCESS File	1 KB
.travis.yml	2/16/2015 3:37 PM	YML File	1 KB
index	7/29/2013 3:54 PM	PHP File	7 KB
license	7/29/2013 3:54 PM	Text Document	3 KB
toko-roti	2/16/2015 10:39 PM	WinRAR ZIP archive	11,663 KB

Gambar 1.3

Didalam folder assets kita bikin css dan jquerynya dan data gambar produknya gambar 1.3 dibawah ini:



Gambar 1.4

Berikut ini sorce code dalam pembuat cssnya:

Nama file style.css

```
body {
font : 73% Tahoma, "Trebuchet MS", Arial, Helvetica, sans-serif;
color : #666;
background : #520934 url(../images/bg.gif) repeat-y top center;
}
                    /*WRAP*/
#container {
width : 750px;
margin : 0 auto;
padding : 0;
line-height : 1.7em;
background : #eee;
color : #666;
}
                    /*TOP BANNER*/
#top {
margin : 0;
padding : 0;
height : 150px;
background : #520934 url(../images/1.jpg) no-repeat right center;
border-bottom : 2px solid #fff;
}
#top h1 {
margin : 0;
font : 400% georgia, Helvetica, "Trebuchet MS", Tahoma, sans-serif;
letter-spacing : 10px;
color : blue;
background : transparent;
}
```

```
#top p {
padding : 10px 0 0 10px;
margin : 0;
}
#top a:link {
text-decoration : none;
color : #eee;
background : inherit;
}
#top a:hover {
text-decoration : underline;
color : #fff;
background : inherit;
}
                    /*TOP MENU*/
/*#menu {
margin : 0;
padding : 0;
height : 30px;
}*/
                    /*LEFT MENU*/
#leftnav {
float : left;
width : 160px;
margin : 0;
padding : 0;
border-right : 2px solid #fff;
color : #333;
background : #eee;
}
#leftnav img{
        width: 70px;
        height: 70px;

}
#leftnav ul li{
        display: block;
        margin-left: -29px;
```

```
}
#pembayaran-left{
        margin-top: 60px;
}
                        /*MAIN CONTENT*/
#content {
margin : 0 0 0 170px;
padding : 0;
}
                        /*FOOTER*/
#footer {
clear : both;
margin : 0;
padding : 0.5em;
border-top : 5px solid #cb274c;
color : #fff;
background : #520934;
}
#footer img {
float : right;
}
* > html #footer img {
margin : 0;
}
#footer a:link, #footer a:visited {
text-decoration : none;
color : #eee;
background : transparent;
text-decoration : none;
}
#footer a:hover {
text-decoration : none;
color : #90b905;
background : inherit;
}
#footer a:active {
text-decoration : none;
background : inherit;
```

```
color : #aaa;
}
                    /*TYPOGRAPHY*/
h2 {
margin : 0;
padding : 0 0 0 5px;
font : 130% georgia, Helvetica, "Trebuchet MS", Tahoma, sans-serif;
letter-spacing : 5px;
border-bottom : 1px solid #eee;
color : #520934;
background : inherit;
}
h3 {
margin : 0;
padding : 0 0 5px 10px;
font : 130% georgia, Helvetica, "Trebuchet MS", Tahoma, sans-serif;
text-transform : uppercase;
color : #520934;
background : inherit;
}
p {
padding : 10px;
}
blockquote {
font-weight : bold;
font-style : italic;
color : #b29b35;
}
                    /*LINKS*/
a:link, a:visited {
text-decoration : none;
color : #cb274c;
background : transparent;
text-decoration : none;
}
a:hover {
text-decoration : none;
color : #520934;
```

```
background : inherit;
}
a:active {
text-decoration : none;
background : inherit;
color : #aaa;
}
a img {
border : none;
}
#navlist li {
list-style-type : square;
margin : 0 15px 0 0;
padding : 0;
background : inherit;
color : #520934;
}
.clear {
clear : both;
overflow : hidden;
width : 0;
height : 1px;
}
.quote {
margin : 10px;
border : 1px solid #cb274c;
padding : 15px;
color : #cb274c;
background : transparent;
}

/*admin*/
.gambaradmin{
	width: 100px;
}
/*halaman home */
.content-center{
	float: left;
```

```
        width: 585px;
        height: 100%;
}
.img-home{
        width: 237px;
        margin: 10px;
        float: left;
}

.img-home img{
        width: 116px;
        height: 64px;
}
.img-promo img{
        width: 62px;
        margin-top: -64px;
        margin-left: 54px;
}
.nama-produk{
        text-align: center;
}
.harga{
        text-align: center;
        margin: 10px;
}
.content-home{
        float: left;
        height: 210px;
        width: 136px;
        margin: 15px;
        padding: 10px;
        background: #fff;
        border: 1px solid #000;
}



/*halaman detail produk*/
```

```
.nama-produk-detail{
	text-align: center;
	font-size: 20px;
	margin: 10px;
}
.img-detail img{
	width: 576px;
}
.keterangan{
	padding: 10px;
	text-align: justify;
}
.btn-pesan{
	width: 60px;
	padding:  5px 15px 5px 15px;
	background: blue;
	margin-bottom: 10px
}
#btn-beli{
	float: right;
	margin-top: -20px;
}
.btn-detail{
	border: 1px solid #000;
width: 33px;
padding: 1px 4px 1px 4px;
background: #E7E7E7;
color: #000;
border-radius: 3px;
margin-top: 12px;
}
```

Untuk pembuatan sebuah ukuran tampilan produk sebagai berikut ini css:

Dengan Nama File menu.css

```
/* Begin CSS Popout Menu */
```

```
#menuh-container
	{
	position:  relative;
	margin: 0;
	padding: 0;
	width:750px;
	}

#menuh
	{
	font-size: 90%;
	font-family: helvetica, "Trebuchet MS",arial, sans-serif;
	width:100%;
	float:left;
	margin-bottom:20px;
	color: #333;
	background:  #CB274C;
	border-bottom : 2px solid #fff;
	}

#menuh a
	{
	text-align: left;
	display:block;
	height:16px;
	border-right:1px solid #fff;
	border-bottom : 1px solid #fff;
	white-space:nowrap;
	margin:0;
	padding: 0.6em 0.2em 0.6em 0.4em;
	}

#menuh a, #menuh a:visited/* menu at rest */
	{
	color: #fff;
	background: #CB274C;
	text-decoration:none;
```

```
	}

#menuh a:hover /* menu at mouse-over  */
	{
	color: #000;
	background: #CC96A4;
	}

#menuh a.top_parent, #menuh a.top_parent:hover  /* attaches down-
arrow to all top-parents */
	{text-transform: uppercase;
	background-image: url(../images/navdown_white.gif);
	background-position: right center;
	background-repeat: no-repeat;
	}

#menuh a.parent, #menuh a.parent:hover	/* attaches side-arrow to all
parents */
	{
	background-image: url(../images/nav_white.gif);
	background-position: right center;
	background-repeat: no-repeat;
	}

#menuh ul
	{
	list-style:none;
	margin:0;
	padding:0;
	float:left;
	width:125px;	/* width of all menu boxes */
	}

#menuh li
	{
	position:relative;
	}
```

```
#menuh ul ul
	{
	position:absolute;
	z-index:500;
	top:auto;
	display:none;
	padding: 1em;
	margin:-1em 0 0 -1em;
	}

#menuh ul ul ul
	{
	top:0;
	left:100%;
	}

div#menuh li:hover
	{
	cursor:pointer;
	z-index:100;
	}

div#menuh li:hover ul ul,
div#menuh li li:hover ul ul,
div#menuh li li li:hover ul ul,
div#menuh li li li li:hover ul ul
{display:none;}

div#menuh li:hover ul,
div#menuh li li:hover ul,
div#menuh li li li:hover ul,
div#menuh li li li li:hover ul
{display:block;}

/* End CSS Popout Menu */
```

Proses selanjunya pembuatan jquery-ui

```
/*! jQuery UI - v1.11.1 - 2014-08-13
* http://jqueryui.com
* Includes: core.css, accordion.css, autocomplete.css, button.css,
datepicker.css, dialog.css, draggable.css, menu.css, progressbar.css,
resizable.css, selectable.css, selectmenu.css, slider.css, sortable.css,
spinner.css, tabs.css, tooltip.css, theme.css

* To view and modify this theme, visit
http://jqueryui.com/themeroller/?ffDefault=Verdana%2CArial%2Csans-
serif&fwDefault=normal&fsDefault=1.1em&cornerRadius=4px&bgColorHeade
r=cccccc&bgTextureHeader=highlight_soft&bgImgOpacityHeader=75&border
ColorHeader=aaaaaa&fcHeader=222222&iconColorHeader=222222&bgColor
Content=ffffff&bgTextureContent=flat&bgImgOpacityContent=75&borderCol
orContent=aaaaaa&fcContent=222222&iconColorContent=222222&bgColorD
efault=e6e6e6&bgTextureDefault=glass&bgImgOpacityDefault=75&borderCo
lorDefault=d3d3d3&fcDefault=555555&iconColorDefault=888888&bgColorHo
ver=dadada&bgTextureHover=glass&bgImgOpacityHover=75&borderColorH
over=999999&fcHover=212121&iconColorHover=454545&bgColorActive=ffff
ff&bgTextureActive=glass&bgImgOpacityActive=65&borderColorActive=aaaa
aa&fcActive=212121&iconColorActive=454545&bgColorHighlight=fbf9ee&bg
TextureHighlight=glass&bgImgOpacityHighlight=55&borderColorHighlight=fc
efa1&fcHighlight=363636&iconColorHighlight=2e83ff&bgColorError=fef1ec&
bgTextureError=glass&bgImgOpacityError=95&borderColorError=cd0a0a&fc
Error=cd0a0a&iconColorError=cd0a0a&bgColorOverlay=aaaaaa&bgTextureO
verlay=flat&bgImgOpacityOverlay=0&opacityOverlay=30&bgColorShadow=a
aaaaa&bgTextureShadow=flat&bgImgOpacityShadow=0&opacityShadow=30
&thicknessShadow=8px&offsetTopShadow=-8px&offsetLeftShadow=-
8px&cornerRadiusShadow=8px
* Copyright 2014 jQuery Foundation and other contributors; Licensed MIT */

/* Layout helpers
----------------------------------*/
.ui-helper-hidden {
	display: none;
}
.ui-helper-hidden-accessible {
	border: 0;
	clip: rect(0 0 0 0);
```

```
	height: 1px;
	margin: -1px;
	overflow: hidden;
	padding: 0;
	position: absolute;
	width: 1px;
}
.ui-helper-reset {
	margin: 0;
	padding: 0;
	border: 0;
	outline: 0;
	line-height: 1.3;
	text-decoration: none;
	font-size: 100%;
	list-style: none;
}
.ui-helper-clearfix:before,
.ui-helper-clearfix:after {
	content: "";
	display: table;
	border-collapse: collapse;
}
.ui-helper-clearfix:after {
	clear: both;
}
.ui-helper-clearfix {
	min-height: 0; /* support: IE7 */
}
.ui-helper-zfix {
	width: 100%;
	height: 100%;
	top: 0;
	left: 0;
	position: absolute;
	opacity: 0;
	filter:Alpha(Opacity=0); /* support: IE8 */
}
```

```
.ui-front {
	z-index: 100;
}


/* Interaction Cues
----------------------------------*/
.ui-state-disabled {
	cursor: default !important;
}


/* Icons
----------------------------------*/

/* states and images */
.ui-icon {
	display: block;
	text-indent: -99999px;
	overflow: hidden;
	background-repeat: no-repeat;
}


/* Misc visuals
----------------------------------*/

/* Overlays */
.ui-widget-overlay {
	position: fixed;
	top: 0;
	left: 0;
	width: 100%;
	height: 100%;
}
.ui-accordion .ui-accordion-header {
	display: block;
```

```
	cursor: pointer;
	position: relative;
	margin: 2px 0 0 0;
	padding: .5em .5em .5em .7em;
	min-height: 0; /* support: IE7 */
	font-size: 100%;
}
.ui-accordion .ui-accordion-icons {
	padding-left: 2.2em;
}
.ui-accordion .ui-accordion-icons .ui-accordion-icons {
	padding-left: 2.2em;
}
.ui-accordion .ui-accordion-header .ui-accordion-header-icon {
	position: absolute;
	left: .5em;
	top: 50%;
	margin-top: -8px;
}
.ui-accordion .ui-accordion-content {
	padding: 1em 2.2em;
	border-top: 0;
	overflow: auto;
}
.ui-autocomplete {
	position: absolute;
	top: 0;
	left: 0;
	cursor: default;
}
.ui-button {
	display: inline-block;
	position: relative;
	padding: 0;
	line-height: normal;
	margin-right: .1em;
	cursor: pointer;
	vertical-align: middle;
```

```
	text-align: center;
	overflow: visible; /* removes extra width in IE */
}
.ui-button,
.ui-button:link,
.ui-button:visited,
.ui-button:hover,
.ui-button:active {
	text-decoration: none;
}
/* to make room for the icon, a width needs to be set here */
.ui-button-icon-only {
	width: 2.2em;
}
/* button elements seem to need a little more width */
button.ui-button-icon-only {
	width: 2.4em;
}
.ui-button-icons-only {
	width: 3.4em;
}
button.ui-button-icons-only {
	width: 3.7em;
}

/* button text element */
.ui-button .ui-button-text {
	display: block;
	line-height: normal;
}
.ui-button-text-only .ui-button-text {
	padding: .4em 1em;
}
.ui-button-icon-only .ui-button-text,
.ui-button-icons-only .ui-button-text {
	padding: .4em;
	text-indent: -9999999px;
}
```

```
.ui-button-text-icon-primary .ui-button-text,
.ui-button-text-icons .ui-button-text {
	padding: .4em 1em .4em 2.1em;
}
.ui-button-text-icon-secondary .ui-button-text,
.ui-button-text-icons .ui-button-text {
	padding: .4em 2.1em .4em 1em;
}
.ui-button-text-icons .ui-button-text {
	padding-left: 2.1em;
	padding-right: 2.1em;
}
/* no icon support for input elements, provide padding by default */
input.ui-button {
	padding: .4em 1em;
}

/* button icon element(s) */
.ui-button-icon-only .ui-icon,
.ui-button-text-icon-primary .ui-icon,
.ui-button-text-icon-secondary .ui-icon,
.ui-button-text-icons .ui-icon,
.ui-button-icons-only .ui-icon {
	position: absolute;
	top: 50%;
	margin-top: -8px;
}
.ui-button-icon-only .ui-icon {
	left: 50%;
	margin-left: -8px;
}
.ui-button-text-icon-primary .ui-button-icon-primary,
.ui-button-text-icons .ui-button-icon-primary,
.ui-button-icons-only .ui-button-icon-primary {
	left: .5em;
}
.ui-button-text-icon-secondary .ui-button-icon-secondary,
.ui-button-text-icons .ui-button-icon-secondary,
```

```
.ui-button-icons-only .ui-button-icon-secondary {
	right: .5em;
}

/* button sets */
.ui-buttonset {
	margin-right: 7px;
}
.ui-buttonset .ui-button {
	margin-left: 0;
	margin-right: -.3em;
}

/* workarounds */
/* reset extra padding in Firefox, see h5bp.com/l */
input.ui-button::-moz-focus-inner,
button.ui-button::-moz-focus-inner {
	border: 0;
	padding: 0;
}
.ui-datepicker {
	width: 17em;
	padding: .2em .2em 0;
	display: none;
}
.ui-datepicker .ui-datepicker-header {
	position: relative;
	padding: .2em 0;
}
.ui-datepicker .ui-datepicker-prev,
.ui-datepicker .ui-datepicker-next {
	position: absolute;
	top: 2px;
	width: 1.8em;
	height: 1.8em;
}
.ui-datepicker .ui-datepicker-prev-hover,
.ui-datepicker .ui-datepicker-next-hover {
```

```
	top: 1px;
}
.ui-datepicker .ui-datepicker-prev {
	left: 2px;
}
.ui-datepicker .ui-datepicker-next {
	right: 2px;
}
.ui-datepicker .ui-datepicker-prev-hover {
	left: 1px;
}
.ui-datepicker .ui-datepicker-next-hover {
	right: 1px;
}
.ui-datepicker .ui-datepicker-prev span,
.ui-datepicker .ui-datepicker-next span {
	display: block;
	position: absolute;
	left: 50%;
	margin-left: -8px;
	top: 50%;
	margin-top: -8px;
}
.ui-datepicker .ui-datepicker-title {
	margin: 0 2.3em;
	line-height: 1.8em;
	text-align: center;
}
.ui-datepicker .ui-datepicker-title select {
	font-size: 1em;
	margin: 1px 0;
}
.ui-datepicker select.ui-datepicker-month,
.ui-datepicker select.ui-datepicker-year {
	width: 45%;
}
.ui-datepicker table {
	width: 100%;
```

```
	font-size: .9em;
	border-collapse: collapse;
	margin: 0 0 .4em;
}
.ui-datepicker th {
	padding: .7em .3em;
	text-align: center;
	font-weight: bold;
	border: 0;
}
.ui-datepicker td {
	border: 0;
	padding: 1px;
}
.ui-datepicker td span,
.ui-datepicker td a {
	display: block;
	padding: .2em;
	text-align: right;
	text-decoration: none;
}
.ui-datepicker .ui-datepicker-buttonpane {
	background-image: none;
	margin: .7em 0 0 0;
	padding: 0 .2em;
	border-left: 0;
	border-right: 0;
	border-bottom: 0;
}
.ui-datepicker .ui-datepicker-buttonpane button {
	float: right;
	margin: .5em .2em .4em;
	cursor: pointer;
	padding: .2em .6em .3em .6em;
	width: auto;
	overflow: visible;
}
.ui-datepicker .ui-datepicker-buttonpane button.ui-datepicker-current {
```

```
	float: left;
}

/* with multiple calendars */
.ui-datepicker.ui-datepicker-multi {
	width: auto;
}
.ui-datepicker-multi .ui-datepicker-group {
	float: left;
}
.ui-datepicker-multi .ui-datepicker-group table {
	width: 95%;
	margin: 0 auto .4em;
}
.ui-datepicker-multi-2 .ui-datepicker-group {
	width: 50%;
}
.ui-datepicker-multi-3 .ui-datepicker-group {
	width: 33.3%;
}
.ui-datepicker-multi-4 .ui-datepicker-group {
	width: 25%;
}
.ui-datepicker-multi .ui-datepicker-group-last .ui-datepicker-header,
.ui-datepicker-multi .ui-datepicker-group-middle .ui-datepicker-header {
	border-left-width: 0;
}
.ui-datepicker-multi .ui-datepicker-buttonpane {
	clear: left;
}
.ui-datepicker-row-break {
	clear: both;
	width: 100%;
	font-size: 0;
}

/* RTL support */
.ui-datepicker-rtl {
```

```
	direction: rtl;
}
.ui-datepicker-rtl .ui-datepicker-prev {
	right: 2px;
	left: auto;
}
.ui-datepicker-rtl .ui-datepicker-next {
	left: 2px;
	right: auto;
}
.ui-datepicker-rtl .ui-datepicker-prev:hover {
	right: 1px;
	left: auto;
}
.ui-datepicker-rtl .ui-datepicker-next:hover {
	left: 1px;
	right: auto;
}
.ui-datepicker-rtl .ui-datepicker-buttonpane {
	clear: right;
}
.ui-datepicker-rtl .ui-datepicker-buttonpane button {
	float: left;
}
.ui-datepicker-rtl .ui-datepicker-buttonpane button.ui-datepicker-current,
.ui-datepicker-rtl .ui-datepicker-group {
	float: right;
}
.ui-datepicker-rtl .ui-datepicker-group-last .ui-datepicker-header,
.ui-datepicker-rtl .ui-datepicker-group-middle .ui-datepicker-header {
	border-right-width: 0;
	border-left-width: 1px;
}
.ui-dialog {
	overflow: hidden;
	position: absolute;
	top: 0;
	left: 0;
```

```
        padding: .2em;
        outline: 0;
}
.ui-dialog .ui-dialog-titlebar {
        padding: .4em 1em;
        position: relative;
}
.ui-dialog .ui-dialog-title {
        float: left;
        margin: .1em 0;
        white-space: nowrap;
        width: 90%;
        overflow: hidden;
        text-overflow: ellipsis;
}
.ui-dialog .ui-dialog-titlebar-close {
        position: absolute;
        right: .3em;
        top: 50%;
        width: 20px;
        margin: -10px 0 0 0;
        padding: 1px;
        height: 20px;
}
.ui-dialog .ui-dialog-content {
        position: relative;
        border: 0;
        padding: .5em 1em;
        background: none;
        overflow: auto;
}
.ui-dialog .ui-dialog-buttonpane {
        text-align: left;
        border-width: 1px 0 0 0;
        background-image: none;
        margin-top: .5em;
        padding: .3em 1em .5em .4em;
}
```

```
.ui-dialog .ui-dialog-buttonpane .ui-dialog-buttonset {
	float: right;
}
.ui-dialog .ui-dialog-buttonpane button {
	margin: .5em .4em .5em 0;
	cursor: pointer;
}
.ui-dialog .ui-resizable-se {
	width: 12px;
	height: 12px;
	right: -5px;
	bottom: -5px;
	background-position: 16px 16px;
}
.ui-draggable .ui-dialog-titlebar {
	cursor: move;
}
.ui-draggable-handle {
	-ms-touch-action: none;
	touch-action: none;
}
.ui-menu {
	list-style: none;
	padding: 0;
	margin: 0;
	display: block;
	outline: none;
}
.ui-menu .ui-menu {
	position: absolute;
}
.ui-menu .ui-menu-item {
	position: relative;
	margin: 0;
	padding: 3px 1em 3px .4em;
	cursor: pointer;
	min-height: 0; /* support: IE7 */
	/* support: IE10, see #8844 */
```

```
	list-style-image:
url("
AAAABAAEAAAIBRAA7");
}
.ui-menu .ui-menu-divider {
	margin: 5px 0;
	height: 0;
	font-size: 0;
	line-height: 0;
	border-width: 1px 0 0 0;
}
.ui-menu .ui-state-focus,
.ui-menu .ui-state-active {
	margin: -1px;
}

/* icon support */
.ui-menu-icons {
	position: relative;
}
.ui-menu-icons .ui-menu-item {
	padding-left: 2em;
}

/* left-aligned */
.ui-menu .ui-icon {
	position: absolute;
	top: 0;
	bottom: 0;
	left: .2em;
	margin: auto 0;
}

/* right-aligned */
.ui-menu .ui-menu-icon {
	left: auto;
	right: 0;
}
```

```
.ui-progressbar {
	height: 2em;
	text-align: left;
	overflow: hidden;
}
.ui-progressbar .ui-progressbar-value {
	margin: -1px;
	height: 100%;
}
.ui-progressbar .ui-progressbar-overlay {
	background:
url("
FMi4wAwEAAAAh+QQJAQABACwAAAAAKAAoAAACkYwNqXrdC52DS06a7MFZI
+4FHBCKoDeWKXqymPqGqxvJrXZbMx7Ttc+w9XgU2FB3lOyQRWET2IFGiU9m
1frDVpxZZc6bfHwv4c1YXP6k1Vdy292Fb6UkuvFtXpvWSzA+HycXJHUXiGYIiM
g2R6W459gnWGfHNdjIqDWVqemH2ekpObkpOlppWUqZiqr6edqqWQAAIfkEC
QEAAQAsAAAAACgAKAAAApSMgZnGfaqcg1E2uuzDmmHUBR8Qil95hiPKqWn3
aqtLsS18y7G1SzNeowWBENtQd+T1JktP05nzPTdJZlR6vUxNWWjV+vUWhWN
kWFwxl9VpZRedYcflIOLafaa28XdsH/ynlcc1uPVDZxQIR0K25+cICCmoqCe5m
GhZOfeYSUh5yJcJyrkZWWpaR8doJ2o4NYq62lAAACH5BAkBAAEALAAAAAAoA
CgAAAKVDI4Yy22ZnINRNqosw0Bv7i1gyHUkFj7oSaWlu3ovC8GxNso5fluz3qLV
hBVeT/Lz7ZTHyxL5dDalQWPVOsQWtRnuwXaFTj9jVVh8pma9JjZ4zYSj5ZOym
a7uuolffh+IR5aW97cHuBUXKGKXlKjn+DiHWMcYJah4N0lYCMlJOXipGRr5qdgo
STrqWSq6WFl2ypoaUAAAIfkECQEAAQAsAAAAACgAKAAAApaEb6HLgd/iO7FN
WtcFWe+ufODGjRfoiJ2akShbueb0wtI50zm02pbvwfWEMWBQ1zKGlLIhskiEPm
9R6vRXxV4ZzWT2yHOGpWMyorblKlNp8HmHEb/lCXjcW7bmtXP8Xt229OVWR
1fod2eWqNfHuMjXCPkIGNileOiImVmCOEmoSfn3yXlJWmoHGhqp6ilYuWYpmT
qKUgAAIfkECQEAAQAsAAAAACgAKAAAApiEH6kb58biQ3FNWtMFWW3eNVcoju
FGfqnZqSebuS06w5V80/X02pKe8zFwP6EFWOT1lDFk8rGERh1TTNOocQ61Hm
4Xm2VexUHpzjymViHrFbiELsefVrn6XKfnt2Q9G/+Xdie499XHd2g4h7ioOGhXG
JboGAnXSBnoBwKYyfioubZJ2Hn0RuRZaflZOil56Zp6iioKSXpUAAAh+QQJAQAB
ACwAAAAKAAoAAACkoQRqRvnxuI7kU1a1UU5bd5tnSeOZXhmn5lWK3qNTW
vRdQxP8qvaC+/yaYQzXO7BMvaUEmJRd3TsiMAgswmNYrSgZdYrTX6tSHGZO7
3ezuAw2uxuQ+BbeZfMxsexY35+/Qe4J1inV0g4x3WHuMhIl2jXOKT2Q+VU5fg
oSUI52VfZyfkJGkha6jmY+aaYdirq+lQAACH5BAkBAAEALAAAAAAoACgAAAKW
BIKpYe0L3YNKToqswUlvznigd4wiR4KhZrKt9Upqip61i9E3vMvxRdHlbEFiEXfk9
YARYxOZZD6VQ2pUunBmtRXo1Lf8hMVVcNl8JafV38aM2/Fu5V16Bn63r6xt97j
09+MXSFi4BniGFae3hzbH9+hYBzkpuUh5aZmHuanZOZgIuvbGiNeomCnaxxa
```

p2upaCZsq+1kAACH5BAkBAAEALAAAAAAoACgAAAKXjI8By5zf4kOxTVrXNVlv
1X0d8IGZGKLnNpYtm8Lr9cqVeuOSvfOW79D9aDHizNhDJidFZhNydEahOaDH6
nomtJjp1tutKoNWkvA6JqfRVLHU/QUfau9l2x7G54d1fl995xcIGAdXqMfBNadoY
rhH+Mg2KBlpVpbluCiXmMnZ2Sh4GBqJ+ckIOqqJ6LmKSllZmsoq6wpQAAAh+
QQJAQABACwAAAAKAAoAAAClYx/oLvoxuJDkU1a1YUZbJ59nSd2ZXhWqbRa2
/gF8Gu2DY3iqs7yrq+xBYEkYvFSM8aSSObE+ZgRl1BHFZNr7pRCavZ5BW2142
hY3AN/zWtsmf12p9XxxFl2lpLn1rseztfXZjdIWIf2s5dItwjYKBgo9yg5pHgzJXTE
eGlZuenpyPmpGQoKOWkYmSpaSnqKileI2FAAACH5BAkBAAEALAAAAAAoACg
AAAKVjB+gu+jG4kORTVrVhRlsnn2dJ3ZleFaptFrb+CXmO9OozeL5VfP99HvAW
hpiUdcwkpBH3825AwYdU8xTqlLGhtCosArKMpvfa1mMRae9VvWZfeB2XfPkeL
mm18lUcBj+p5dnN8jXZ3YIGEhYuOUn45aoCDkp16hl5IjYJvjWKcnoGQpqyPlp
Ohr3aElaqrq56Bq7VAAAOw==");
```
        height: 100%;
        filter: alpha(opacity=25); /* support: IE8 */
        opacity: 0.25;
}
.ui-progressbar-indeterminate .ui-progressbar-value {
        background-image: none;
}
.ui-resizable {
        position: relative;
}
.ui-resizable-handle {
        position: absolute;
        font-size: 0.1px;
        display: block;
        -ms-touch-action: none;
        touch-action: none;
}
.ui-resizable-disabled .ui-resizable-handle,
.ui-resizable-autohide .ui-resizable-handle {
        display: none;
}
.ui-resizable-n {
        cursor: n-resize;
        height: 7px;
        width: 100%;
        top: -5px;
```

```
        left: 0;
}
.ui-resizable-s {
        cursor: s-resize;
        height: 7px;
        width: 100%;
        bottom: -5px;
        left: 0;
}
.ui-resizable-e {
        cursor: e-resize;
        width: 7px;
        right: -5px;
        top: 0;
        height: 100%;
}
.ui-resizable-w {
        cursor: w-resize;
        width: 7px;
        left: -5px;
        top: 0;
        height: 100%;
}
.ui-resizable-se {
        cursor: se-resize;
        width: 12px;
        height: 12px;
        right: 1px;
        bottom: 1px;
}
.ui-resizable-sw {
        cursor: sw-resize;
        width: 9px;
        height: 9px;
        left: -5px;
        bottom: -5px;
}
.ui-resizable-nw {
```

```
	cursor: nw-resize;
	width: 9px;
	height: 9px;
	left: -5px;
	top: -5px;
}
.ui-resizable-ne {
	cursor: ne-resize;
	width: 9px;
	height: 9px;
	right: -5px;
	top: -5px;
}
.ui-selectable {
	-ms-touch-action: none;
	touch-action: none;
}
.ui-selectable-helper {
	position: absolute;
	z-index: 100;
	border: 1px dotted black;
}
.ui-selectmenu-menu {
	padding: 0;
	margin: 0;
	position: absolute;
	top: 0;
	left: 0;
	display: none;
}
.ui-selectmenu-menu .ui-menu {
	overflow: auto;
	/* Support: IE7 */
	overflow-x: hidden;
	padding-bottom: 1px;
}
.ui-selectmenu-menu .ui-menu .ui-selectmenu-optgroup {
	font-size: 1em;
```

```
	font-weight: bold;
	line-height: 1.5;
	padding: 2px 0.4em;
	margin: 0.5em 0 0 0;
	height: auto;
	border: 0;
}
.ui-selectmenu-open {
	display: block;
}
.ui-selectmenu-button {
	display: inline-block;
	overflow: hidden;
	position: relative;
	text-decoration: none;
	cursor: pointer;
}
.ui-selectmenu-button span.ui-icon {
	right: 0.5em;
	left: auto;
	margin-top: -8px;
	position: absolute;
	top: 50%;
}
.ui-selectmenu-button span.ui-selectmenu-text {
	text-align: left;
	padding: 0.4em 2.1em 0.4em 1em;
	display: block;
	line-height: 1.4;
	overflow: hidden;
	text-overflow: ellipsis;
	white-space: nowrap;
}
.ui-slider {
	position: relative;
	text-align: left;
}
.ui-slider .ui-slider-handle {
```

```
	position: absolute;
	z-index: 2;
	width: 1.2em;
	height: 1.2em;
	cursor: default;
	-ms-touch-action: none;
	touch-action: none;
}
.ui-slider .ui-slider-range {
	position: absolute;
	z-index: 1;
	font-size: .7em;
	display: block;
	border: 0;
	background-position: 0 0;
}

/* support: IE8 - See #6727 */
.ui-slider.ui-state-disabled .ui-slider-handle,
.ui-slider.ui-state-disabled .ui-slider-range {
	filter: inherit;
}

.ui-slider-horizontal {
	height: .8em;
}
.ui-slider-horizontal .ui-slider-handle {
	top: -.3em;
	margin-left: -.6em;
}
.ui-slider-horizontal .ui-slider-range {
	top: 0;
	height: 100%;
}
.ui-slider-horizontal .ui-slider-range-min {
	left: 0;
}
.ui-slider-horizontal .ui-slider-range-max {
```

```
	right: 0;
}

.ui-slider-vertical {
	width: .8em;
	height: 100px;
}
.ui-slider-vertical .ui-slider-handle {
	left: -.3em;
	margin-left: 0;
	margin-bottom: -.6em;
}
.ui-slider-vertical .ui-slider-range {
	left: 0;
	width: 100%;
}
.ui-slider-vertical .ui-slider-range-min {
	bottom: 0;
}
.ui-slider-vertical .ui-slider-range-max {
	top: 0;
}
.ui-sortable-handle {
	-ms-touch-action: none;
	touch-action: none;
}
.ui-spinner {
	position: relative;
	display: inline-block;
	overflow: hidden;
	padding: 0;
	vertical-align: middle;
}
.ui-spinner-input {
	border: none;
	background: none;
	color: inherit;
	padding: 0;
```

```
        margin: .2em 0;
        vertical-align: middle;
        margin-left: .4em;
        margin-right: 22px;
}
.ui-spinner-button {
        width: 16px;
        height: 50%;
        font-size: .5em;
        padding: 0;
        margin: 0;
        text-align: center;
        position: absolute;
        cursor: default;
        display: block;
        overflow: hidden;
        right: 0;
}
/* more specificity required here to override default borders */
.ui-spinner a.ui-spinner-button {
        border-top: none;
        border-bottom: none;
        border-right: none;
}
/* vertically center icon */
.ui-spinner .ui-icon {
        position: absolute;
        margin-top: -8px;
        top: 50%;
        left: 0;
}
.ui-spinner-up {
        top: 0;
}
.ui-spinner-down {
        bottom: 0;
}
```

```
/* TR overrides */
.ui-spinner .ui-icon-triangle-1-s {
	/* need to fix icons sprite */
	background-position: -65px -16px;
}
.ui-tabs {
	position: relative;/* position: relative prevents IE scroll bug (element
with position: relative inside container with overflow: auto appear as "fixed")
*/
	padding: .2em;
}
.ui-tabs .ui-tabs-nav {
	margin: 0;
	padding: .2em .2em 0;
}
.ui-tabs .ui-tabs-nav li {
	list-style: none;
	float: left;
	position: relative;
	top: 0;
	margin: 1px .2em 0 0;
	border-bottom-width: 0;
	padding: 0;
	white-space: nowrap;
}
.ui-tabs .ui-tabs-nav .ui-tabs-anchor {
	float: left;
	padding: .5em 1em;
	text-decoration: none;
}
.ui-tabs .ui-tabs-nav li.ui-tabs-active {
	margin-bottom: -1px;
	padding-bottom: 1px;
}
.ui-tabs .ui-tabs-nav li.ui-tabs-active .ui-tabs-anchor,
.ui-tabs .ui-tabs-nav li.ui-state-disabled .ui-tabs-anchor,
.ui-tabs .ui-tabs-nav li.ui-tabs-loading .ui-tabs-anchor {
	cursor: text;
```

```
}
.ui-tabs-collapsible .ui-tabs-nav li.ui-tabs-active .ui-tabs-anchor {
	cursor: pointer;
}
.ui-tabs .ui-tabs-panel {
	display: block;
	border-width: 0;
	padding: 1em 1.4em;
	background: none;
}
.ui-tooltip {
	padding: 8px;
	position: absolute;
	z-index: 9999;
	max-width: 300px;
	-webkit-box-shadow: 0 0 5px #aaa;
	box-shadow: 0 0 5px #aaa;
}
body .ui-tooltip {
	border-width: 2px;
}

/* Component containers
----------------------------------*/
.ui-widget {
	font-family: Verdana,Arial,sans-serif;
	font-size: 1.1em;
}
.ui-widget .ui-widget {
	font-size: 1em;
}
.ui-widget input,
.ui-widget select,
.ui-widget textarea,
.ui-widget button {
	font-family: Verdana,Arial,sans-serif;
	font-size: 1em;
}
```

```
.ui-widget-content {
	border: 1px solid #aaaaaa;
	background: #ffffff url("images/ui-bg_flat_75_ffffff_40x100.png")
50% 50% repeat-x;
	color: #222222;
}
.ui-widget-content a {
	color: #222222;
}
.ui-widget-header {
	border: 1px solid #aaaaaa;
	background: #cccccc url("images/ui-bg_highlight-
soft_75_cccccc_1x100.png") 50% 50% repeat-x;
	color: #222222;
	font-weight: bold;
}
.ui-widget-header a {
	color: #222222;
}

/* Interaction states
----------------------------------*/
.ui-state-default,
.ui-widget-content .ui-state-default,
.ui-widget-header .ui-state-default {
	border: 1px solid #d3d3d3;
	background: #e6e6e6 url("images/ui-
bg_glass_75_e6e6e6_1x400.png") 50% 50% repeat-x;
	font-weight: normal;
	color: #555555;
}
.ui-state-default a,
.ui-state-default a:link,
.ui-state-default a:visited {
	color: #555555;
	text-decoration: none;
}
.ui-state-hover,
```

```
.ui-widget-content .ui-state-hover,
.ui-widget-header .ui-state-hover,
.ui-state-focus,
.ui-widget-content .ui-state-focus,
.ui-widget-header .ui-state-focus {
	border: 1px solid #999999;
	background: #dadada url("images/ui-
bg_glass_75_dadada_1x400.png") 50% 50% repeat-x;
	font-weight: normal;
	color: #212121;
}
.ui-state-hover a,
.ui-state-hover a:hover,
.ui-state-hover a:link,
.ui-state-hover a:visited,
.ui-state-focus a,
.ui-state-focus a:hover,
.ui-state-focus a:link,
.ui-state-focus a:visited {
	color: #212121;
	text-decoration: none;
}
.ui-state-active,
.ui-widget-content .ui-state-active,
.ui-widget-header .ui-state-active {
	border: 1px solid #aaaaaa;
	background: #ffffff url("images/ui-bg_glass_65_ffffff_1x400.png")
50% 50% repeat-x;
	font-weight: normal;
	color: #212121;
}
.ui-state-active a,
.ui-state-active a:link,
.ui-state-active a:visited {
	color: #212121;
	text-decoration: none;
}
```

```
/* Interaction Cues
----------------------------------*/
.ui-state-highlight,
.ui-widget-content .ui-state-highlight,
.ui-widget-header .ui-state-highlight {
	border: 1px solid #fcefa1;
	background: #fbf9ee url("images/ui-bg_glass_55_fbf9ee_1x400.png")
50% 50% repeat-x;
	color: #363636;
}
.ui-state-highlight a,
.ui-widget-content .ui-state-highlight a,
.ui-widget-header .ui-state-highlight a {
	color: #363636;
}
.ui-state-error,
.ui-widget-content .ui-state-error,
.ui-widget-header .ui-state-error {
	border: 1px solid #cd0a0a;
	background: #fef1ec url("images/ui-bg_glass_95_fef1ec_1x400.png")
50% 50% repeat-x;
	color: #cd0a0a;
}
.ui-state-error a,
.ui-widget-content .ui-state-error a,
.ui-widget-header .ui-state-error a {
	color: #cd0a0a;
}
.ui-state-error-text,
.ui-widget-content .ui-state-error-text,
.ui-widget-header .ui-state-error-text {
	color: #cd0a0a;
}
.ui-priority-primary,
.ui-widget-content .ui-priority-primary,
.ui-widget-header .ui-priority-primary {
	font-weight: bold;
}
```

```
.ui-priority-secondary,
.ui-widget-content .ui-priority-secondary,
.ui-widget-header .ui-priority-secondary {
	opacity: .7;
	filter:Alpha(Opacity=70); /* support: IE8 */
	font-weight: normal;
}
.ui-state-disabled,
.ui-widget-content .ui-state-disabled,
.ui-widget-header .ui-state-disabled {
	opacity: .35;
	filter:Alpha(Opacity=35); /* support: IE8 */
	background-image: none;
}
.ui-state-disabled .ui-icon {
	filter:Alpha(Opacity=35); /* support: IE8 - See #6059 */
}

/* Icons
----------------------------------*/

/* states and images */
.ui-icon {
	width: 16px;
	height: 16px;
}
.ui-icon,
.ui-widget-content .ui-icon {
	background-image: url("images/ui-icons_222222_256x240.png");
}
.ui-widget-header .ui-icon {
	background-image: url("images/ui-icons_222222_256x240.png");
}
.ui-state-default .ui-icon {
	background-image: url("images/ui-icons_888888_256x240.png");
}
.ui-state-hover .ui-icon,
.ui-state-focus .ui-icon {
```

```
	background-image: url("images/ui-icons_454545_256x240.png");
}
.ui-state-active .ui-icon {
	background-image: url("images/ui-icons_454545_256x240.png");
}
.ui-state-highlight .ui-icon {
	background-image: url("images/ui-icons_2e83ff_256x240.png");
}
.ui-state-error .ui-icon,
.ui-state-error-text .ui-icon {
	background-image: url("images/ui-icons_cd0a0a_256x240.png");
}

/* positioning */
.ui-icon-blank { background-position: 16px 16px; }
.ui-icon-carat-1-n { background-position: 0 0; }
.ui-icon-carat-1-ne { background-position: -16px 0; }
.ui-icon-carat-1-e { background-position: -32px 0; }
.ui-icon-carat-1-se { background-position: -48px 0; }
.ui-icon-carat-1-s { background-position: -64px 0; }
.ui-icon-carat-1-sw { background-position: -80px 0; }
.ui-icon-carat-1-w { background-position: -96px 0; }
.ui-icon-carat-1-nw { background-position: -112px 0; }
.ui-icon-carat-2-n-s { background-position: -128px 0; }
.ui-icon-carat-2-e-w { background-position: -144px 0; }
.ui-icon-triangle-1-n { background-position: 0 -16px; }
.ui-icon-triangle-1-ne { background-position: -16px -16px; }
.ui-icon-triangle-1-e { background-position: -32px -16px; }
.ui-icon-triangle-1-se { background-position: -48px -16px; }
.ui-icon-triangle-1-s { background-position: -64px -16px; }
.ui-icon-triangle-1-sw { background-position: -80px -16px; }
.ui-icon-triangle-1-w { background-position: -96px -16px; }
.ui-icon-triangle-1-nw { background-position: -112px -16px; }
.ui-icon-triangle-2-n-s { background-position: -128px -16px; }
.ui-icon-triangle-2-e-w { background-position: -144px -16px; }
.ui-icon-arrow-1-n { background-position: 0 -32px; }
.ui-icon-arrow-1-ne { background-position: -16px -32px; }
.ui-icon-arrow-1-e { background-position: -32px -32px; }
```

```
.ui-icon-arrow-1-se { background-position: -48px -32px; }
.ui-icon-arrow-1-s { background-position: -64px -32px; }
.ui-icon-arrow-1-sw { background-position: -80px -32px; }
.ui-icon-arrow-1-w { background-position: -96px -32px; }
.ui-icon-arrow-1-nw { background-position: -112px -32px; }
.ui-icon-arrow-2-n-s { background-position: -128px -32px; }
.ui-icon-arrow-2-ne-sw { background-position: -144px -32px; }
.ui-icon-arrow-2-e-w { background-position: -160px -32px; }
.ui-icon-arrow-2-se-nw { background-position: -176px -32px; }
.ui-icon-arrowstop-1-n { background-position: -192px -32px; }
.ui-icon-arrowstop-1-e { background-position: -208px -32px; }
.ui-icon-arrowstop-1-s { background-position: -224px -32px; }
.ui-icon-arrowstop-1-w { background-position: -240px -32px; }
.ui-icon-arrowthick-1-n { background-position: 0 -48px; }
.ui-icon-arrowthick-1-ne { background-position: -16px -48px; }
.ui-icon-arrowthick-1-e { background-position: -32px -48px; }
.ui-icon-arrowthick-1-se { background-position: -48px -48px; }
.ui-icon-arrowthick-1-s { background-position: -64px -48px; }
.ui-icon-arrowthick-1-sw { background-position: -80px -48px; }
.ui-icon-arrowthick-1-w { background-position: -96px -48px; }
.ui-icon-arrowthick-1-nw { background-position: -112px -48px; }
.ui-icon-arrowthick-2-n-s { background-position: -128px -48px; }
.ui-icon-arrowthick-2-ne-sw { background-position: -144px -48px; }
.ui-icon-arrowthick-2-e-w { background-position: -160px -48px; }
.ui-icon-arrowthick-2-se-nw { background-position: -176px -48px; }
.ui-icon-arrowthickstop-1-n { background-position: -192px -48px; }
.ui-icon-arrowthickstop-1-e { background-position: -208px -48px; }
.ui-icon-arrowthickstop-1-s { background-position: -224px -48px; }
.ui-icon-arrowthickstop-1-w { background-position: -240px -48px; }
.ui-icon-arrowreturnthick-1-w { background-position: 0 -64px; }
.ui-icon-arrowreturnthick-1-n { background-position: -16px -64px; }
.ui-icon-arrowreturnthick-1-e { background-position: -32px -64px; }
.ui-icon-arrowreturnthick-1-s { background-position: -48px -64px; }
.ui-icon-arrowreturn-1-w { background-position: -64px -64px; }
.ui-icon-arrowreturn-1-n { background-position: -80px -64px; }
.ui-icon-arrowreturn-1-e { background-position: -96px -64px; }
.ui-icon-arrowreturn-1-s { background-position: -112px -64px; }
.ui-icon-arrowrefresh-1-w { background-position: -128px -64px; }
```

```
.ui-icon-arrowrefresh-1-n { background-position: -144px -64px; }
.ui-icon-arrowrefresh-1-e { background-position: -160px -64px; }
.ui-icon-arrowrefresh-1-s { background-position: -176px -64px; }
.ui-icon-arrow-4 { background-position: 0 -80px; }
.ui-icon-arrow-4-diag { background-position: -16px -80px; }
.ui-icon-extlink { background-position: -32px -80px; }
.ui-icon-newwin { background-position: -48px -80px; }
.ui-icon-refresh { background-position: -64px -80px; }
.ui-icon-shuffle { background-position: -80px -80px; }
.ui-icon-transfer-e-w { background-position: -96px -80px; }
.ui-icon-transferthick-e-w { background-position: -112px -80px; }
.ui-icon-folder-collapsed { background-position: 0 -96px; }
.ui-icon-folder-open { background-position: -16px -96px; }
.ui-icon-document { background-position: -32px -96px; }
.ui-icon-document-b { background-position: -48px -96px; }
.ui-icon-note { background-position: -64px -96px; }
.ui-icon-mail-closed { background-position: -80px -96px; }
.ui-icon-mail-open { background-position: -96px -96px; }
.ui-icon-suitcase { background-position: -112px -96px; }
.ui-icon-comment { background-position: -128px -96px; }
.ui-icon-person { background-position: -144px -96px; }
.ui-icon-print { background-position: -160px -96px; }
.ui-icon-trash { background-position: -176px -96px; }
.ui-icon-locked { background-position: -192px -96px; }
.ui-icon-unlocked { background-position: -208px -96px; }
.ui-icon-bookmark { background-position: -224px -96px; }
.ui-icon-tag { background-position: -240px -96px; }
.ui-icon-home { background-position: 0 -112px; }
.ui-icon-flag { background-position: -16px -112px; }
.ui-icon-calendar { background-position: -32px -112px; }
.ui-icon-cart { background-position: -48px -112px; }
.ui-icon-pencil { background-position: -64px -112px; }
.ui-icon-clock { background-position: -80px -112px; }
.ui-icon-disk { background-position: -96px -112px; }
.ui-icon-calculator { background-position: -112px -112px; }
.ui-icon-zoomin { background-position: -128px -112px; }
.ui-icon-zoomout { background-position: -144px -112px; }
.ui-icon-search { background-position: -160px -112px; }
```

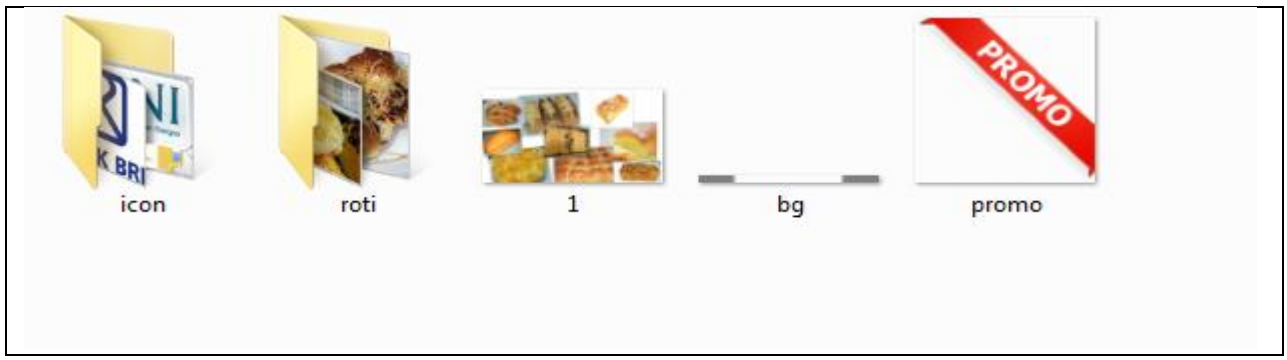
```
.ui-icon-wrench { background-position: -176px -112px; }
.ui-icon-gear { background-position: -192px -112px; }
.ui-icon-heart { background-position: -208px -112px; }
.ui-icon-star { background-position: -224px -112px; }
.ui-icon-link { background-position: -240px -112px; }
.ui-icon-cancel { background-position: 0 -128px; }
.ui-icon-plus { background-position: -16px -128px; }
.ui-icon-plusthick { background-position: -32px -128px; }
.ui-icon-minus { background-position: -48px -128px; }
.ui-icon-minusthick { background-position: -64px -128px; }
.ui-icon-close { background-position: -80px -128px; }
.ui-icon-closethick { background-position: -96px -128px; }
.ui-icon-key { background-position: -112px -128px; }
.ui-icon-lightbulb { background-position: -128px -128px; }
.ui-icon-scissors { background-position: -144px -128px; }
.ui-icon-clipboard { background-position: -160px -128px; }
.ui-icon-copy { background-position: -176px -128px; }
.ui-icon-contact { background-position: -192px -128px; }
.ui-icon-image { background-position: -208px -128px; }
.ui-icon-video { background-position: -224px -128px; }
.ui-icon-script { background-position: -240px -128px; }
.ui-icon-alert { background-position: 0 -144px; }
.ui-icon-info { background-position: -16px -144px; }
.ui-icon-notice { background-position: -32px -144px; }
.ui-icon-help { background-position: -48px -144px; }
.ui-icon-check { background-position: -64px -144px; }
.ui-icon-bullet { background-position: -80px -144px; }
.ui-icon-radio-on { background-position: -96px -144px; }
.ui-icon-radio-off { background-position: -112px -144px; }
.ui-icon-pin-w { background-position: -128px -144px; }
.ui-icon-pin-s { background-position: -144px -144px; }
.ui-icon-play { background-position: 0 -160px; }
.ui-icon-pause { background-position: -16px -160px; }
.ui-icon-seek-next { background-position: -32px -160px; }
.ui-icon-seek-prev { background-position: -48px -160px; }
.ui-icon-seek-end { background-position: -64px -160px; }
.ui-icon-seek-start { background-position: -80px -160px; }
/* ui-icon-seek-first is deprecated, use ui-icon-seek-start instead */
```

```
.ui-icon-seek-first { background-position: -80px -160px; }
.ui-icon-stop { background-position: -96px -160px; }
.ui-icon-eject { background-position: -112px -160px; }
.ui-icon-volume-off { background-position: -128px -160px; }
.ui-icon-volume-on { background-position: -144px -160px; }
.ui-icon-power { background-position: 0 -176px; }
.ui-icon-signal-diag { background-position: -16px -176px; }
.ui-icon-signal { background-position: -32px -176px; }
.ui-icon-battery-0 { background-position: -48px -176px; }
.ui-icon-battery-1 { background-position: -64px -176px; }
.ui-icon-battery-2 { background-position: -80px -176px; }
.ui-icon-battery-3 { background-position: -96px -176px; }
.ui-icon-circle-plus { background-position: 0 -192px; }
.ui-icon-circle-minus { background-position: -16px -192px; }
.ui-icon-circle-close { background-position: -32px -192px; }
.ui-icon-circle-triangle-e { background-position: -48px -192px; }
.ui-icon-circle-triangle-s { background-position: -64px -192px; }
.ui-icon-circle-triangle-w { background-position: -80px -192px; }
.ui-icon-circle-triangle-n { background-position: -96px -192px; }
.ui-icon-circle-arrow-e { background-position: -112px -192px; }
.ui-icon-circle-arrow-s { background-position: -128px -192px; }
.ui-icon-circle-arrow-w { background-position: -144px -192px; }
.ui-icon-circle-arrow-n { background-position: -160px -192px; }
.ui-icon-circle-zoomin { background-position: -176px -192px; }
.ui-icon-circle-zoomout { background-position: -192px -192px; }
.ui-icon-circle-check { background-position: -208px -192px; }
.ui-icon-circlesmall-plus { background-position: 0 -208px; }
.ui-icon-circlesmall-minus { background-position: -16px -208px; }
.ui-icon-circlesmall-close { background-position: -32px -208px; }
.ui-icon-squaresmall-plus { background-position: -48px -208px; }
.ui-icon-squaresmall-minus { background-position: -64px -208px; }
.ui-icon-squaresmall-close { background-position: -80px -208px; }
.ui-icon-grip-dotted-vertical { background-position: 0 -224px; }
.ui-icon-grip-dotted-horizontal { background-position: -16px -224px; }
.ui-icon-grip-solid-vertical { background-position: -32px -224px; }
.ui-icon-grip-solid-horizontal { background-position: -48px -224px; }
.ui-icon-gripsmall-diagonal-se { background-position: -64px -224px; }
.ui-icon-grip-diagonal-se { background-position: -80px -224px; }
```

```
/* Misc visuals
----------------------------------*/

/* Corner radius */
.ui-corner-all,
.ui-corner-top,
.ui-corner-left,
.ui-corner-tl {
	border-top-left-radius: 4px;
}
.ui-corner-all,
.ui-corner-top,
.ui-corner-right,
.ui-corner-tr {
	border-top-right-radius: 4px;
}
.ui-corner-all,
.ui-corner-bottom,
.ui-corner-left,
.ui-corner-bl {
	border-bottom-left-radius: 4px;
}
.ui-corner-all,
.ui-corner-bottom,
.ui-corner-right,
.ui-corner-br {
	border-bottom-right-radius: 4px;
}

/* Overlays */
.ui-widget-overlay {
	background: #aaaaaa url("images/ui-bg_flat_0_aaaaaa_40x100.png")
50% 50% repeat-x;
	opacity: .3;
	filter: Alpha(Opacity=30); /* support: IE8 */
}
```

```
.ui-widget-shadow {
	margin: -8px 0 0 -8px;
	padding: 8px;
	background: #aaaaaa url("images/ui-bg_flat_0_aaaaaa_40x100.png")
50% 50% repeat-x;
	opacity: .3;
	filter: Alpha(Opacity=30); /* support: IE8 */
	border-radius: 8px;
}
```

Dibawah ini gambar isinya didalam folder images untuk gambar produknya.



Gambar 1.5

Pembuatan jquerynya sebgai berikut ini

```
/*!
 * jQuery JavaScript Library v1.10.2
 * http://jquery.com/
 *
 * Includes Sizzle.js
 * http://sizzlejs.com/
 *
 * Copyright 2005, 2013 jQuery Foundation, Inc. and other contributors
 * Released under the MIT license
 * http://jquery.org/license
 *
 * Date: 2013-07-03T13:48Z
 */
(function( window, undefined ) {
```

```
// Can't do this because several apps including ASP.NET trace
// the stack via arguments.caller.callee and Firefox dies if
// you try to trace through "use strict" call chains. (#13335)
// Support: Firefox 18+
//"use strict";
var
	// The deferred used on DOM ready
	readyList,

	// A central reference to the root jQuery(document)
	rootjQuery,

	// Support: IE<10
	// For `typeof xmlNode.method` instead of `xmlNode.method !==
undefined`
	core_strundefined = typeof undefined,

	// Use the correct document accordingly with window argument
(sandbox)
	location = window.location,
	document = window.document,
	docElem = document.documentElement,

	// Map over jQuery in case of overwrite
	_jQuery = window.jQuery,

	// Map over the $ in case of overwrite
	_$ = window.$,

	// [[Class]] -> type pairs
	class2type = {},

	// List of deleted data cache ids, so we can reuse them
	core_deletedIds = [],

	core_version = "1.10.2",

	// Save a reference to some core methods
```

```
	core_concat = core_deletedIds.concat,
	core_push = core_deletedIds.push,
	core_slice = core_deletedIds.slice,
	core_indexOf = core_deletedIds.indexOf,
	core_toString = class2type.toString,
	core_hasOwn = class2type.hasOwnProperty,
	core_trim = core_version.trim,

	// Define a local copy of jQuery
	jQuery = function( selector, context ) {
		// The jQuery object is actually just the init constructor
'enhanced'
		return new jQuery.fn.init( selector, context, rootjQuery );
	},

	// Used for matching numbers
	core_pnum = /[+-]?(?:\d*\.|)\d+(?:[eE][+-]?\d+|)/.source,

	// Used for splitting on whitespace
	core_rnotwhite = /\S+/g,

	// Make sure we trim BOM and NBSP (here's looking at you, Safari 5.0
and IE)
	rtrim = /^[\s\uFEFF\xA0]+|[\s\uFEFF\xA0]+$/g,

	// A simple way to check for HTML strings
	// Prioritize #id over <tag> to avoid XSS via location.hash (#9521)
	// Strict HTML recognition (#11290: must start with <)
	rquickExpr = /^(?:\s*(<[\w\W]+>)[^>]*|#([\w-]*))$/,

	// Match a standalone tag
	rsingleTag = /^<(\w+)\s*\/?>(?:<\/\1>|)$/,

	// JSON RegExp
	rvalidchars = /^[\],:{}\s]*$/,
	rvalidbraces = /(?:^|:|,)(?:\s*\[)+/g,
	rvalidescape = /\\(?:["\\\/bfnrt]|u[\da-fA-F]{4})/g,
	rvalidtokens = /"[^"\\\r\n]*"|true|false|null|-
```

```
?(?:\d+\.|)\d+(?:[eE][+-]?\d+|)/g,

	// Matches dashed string for camelizing
	rmsPrefix = /^-ms-/,
	rdashAlpha = /-([\da-z])/gi,

	// Used by jQuery.camelCase as callback to replace()
	fcamelCase = function( all, letter ) {
		return letter.toUpperCase();
	},

	// The ready event handler
	completed = function( event ) {

		// readyState === "complete" is good enough for us to call the
dom ready in oldIE
		if ( document.addEventListener || event.type === "load" ||
document.readyState === "complete" ) {
			detach();
			jQuery.ready();
		}
	},
	// Clean-up method for dom ready events
	detach = function() {
		if ( document.addEventListener ) {
			document.removeEventListener( "DOMContentLoaded",
completed, false );
			window.removeEventListener( "load", completed, false );

		} else {
			document.detachEvent( "onreadystatechange", completed
);
			window.detachEvent( "onload", completed );
		}
	};

jQuery.fn = jQuery.prototype = {
	// The current version of jQuery being used
```

```
	jquery: core_version,

	constructor: jQuery,
	init: function( selector, context, rootjQuery ) {
		var match, elem;

		// HANDLE: $(""), $(null), $(undefined), $(false)
		if ( !selector ) {
			return this;
		}

		// Handle HTML strings
		if ( typeof selector === "string" ) {
			if ( selector.charAt(0) === "<" && selector.charAt(
selector.length - 1 ) === ">" && selector.length >= 3 ) {
				// Assume that strings that start and end with <>
are HTML and skip the regex check
				match = [ null, selector, null ];

			} else {
				match = rquickExpr.exec( selector );
			}

			// Match html or make sure no context is specified for #id
			if ( match && (match[1] || !context) ) {

				// HANDLE: $(html) -> $(array)
				if ( match[1] ) {
					context = context instanceof jQuery ?
context[0] : context;

					// scripts is true for back-compat
					jQuery.merge( this, jQuery.parseHTML(
						match[1],
						context && context.nodeType ?
context.ownerDocument || context : document,
						true
					) );
```

```
                                // HANDLE: $(html, props)
                                if ( rsingleTag.test( match[1] ) &&
jQuery.isPlainObject( context ) ) {
                                        for ( match in context ) {
                                                // Properties of context are called
as methods if possible
                                                if ( jQuery.isFunction( this[ match ]
) ) {
                                                        this[ match ]( context[
match ] );

                                                // ...and otherwise set as attributes
                                                } else {
                                                        this.attr( match, context[
match ] );
                                                }
                                        }
                                }

                                return this;

                        // HANDLE: $(#id)
                        } else {
                                elem = document.getElementById( match[2]
);

                                // Check parentNode to catch when Blackberry
4.6 returns
                                // nodes that are no longer in the document
#6963
                                if ( elem && elem.parentNode ) {
                                        // Handle the case where IE and Opera
return items
                                        // by name instead of ID
                                        if ( elem.id !== match[2] ) {
                                                return rootjQuery.find( selector );
                                        }
```

```
                                        // Otherwise, we inject the element
directly into the jQuery object
                                        this.length = 1;
                                        this[0] = elem;
                                }

                                this.context = document;
                                this.selector = selector;
                                return this;
                        }

                // HANDLE: $(expr, $(...))
                } else if ( !context || context.jquery ) {
                        return ( context || rootjQuery ).find( selector );

                // HANDLE: $(expr, context)
                // (which is just equivalent to: $(context).find(expr)
                } else {
                        return this.constructor( context ).find( selector );
                }

        // HANDLE: $(DOMElement)
        } else if ( selector.nodeType ) {
                this.context = this[0] = selector;
                this.length = 1;
                return this;

        // HANDLE: $(function)
        // Shortcut for document ready
        } else if ( jQuery.isFunction( selector ) ) {
                return rootjQuery.ready( selector );
        }

        if ( selector.selector !== undefined ) {
                this.selector = selector.selector;
                this.context = selector.context;
        }
```

```
        return jQuery.makeArray( selector, this );
    },

    // Start with an empty selector
    selector: "",

    // The default length of a jQuery object is 0
    length: 0,

    toArray: function() {
        return core_slice.call( this );
    },

    // Get the Nth element in the matched element set OR
    // Get the whole matched element set as a clean array
    get: function( num ) {
        return num == null ?

            // Return a 'clean' array
            this.toArray() :

            // Return just the object
            ( num < 0 ? this[ this.length + num ] : this[ num ] );
    },

    // Take an array of elements and push it onto the stack
    // (returning the new matched element set)
    pushStack: function( elems ) {

        // Build a new jQuery matched element set
        var ret = jQuery.merge( this.constructor(), elems );

        // Add the old object onto the stack (as a reference)
        ret.prevObject = this;
        ret.context = this.context;

        // Return the newly-formed element set
```

```
		return ret;
	},

	// Execute a callback for every element in the matched set.
	// (You can seed the arguments with an array of args, but this is
	// only used internally.)
	each: function( callback, args ) {
		return jQuery.each( this, callback, args );
	},

	ready: function( fn ) {
		// Add the callback
		jQuery.ready.promise().done( fn );

		return this;
	},

	slice: function() {
		return this.pushStack( core_slice.apply( this, arguments ) );
	},

	first: function() {
		return this.eq( 0 );
	},

	last: function() {
		return this.eq( -1 );
	},

	eq: function( i ) {
		var len = this.length,
			j = +i + ( i < 0 ? len : 0 );
		return this.pushStack( j >= 0 && j < len ? [ this[j] ] : [] );
	},

	map: function( callback ) {
		return this.pushStack( jQuery.map(this, function( elem, i ) {
			return callback.call( elem, i, elem );
```

```
            }));
        },

        end: function() {
            return this.prevObject || this.constructor(null);
        },

        // For internal use only.
        // Behaves like an Array's method, not like a jQuery method.
        push: core_push,
        sort: [].sort,
        splice: [].splice
};

// Give the init function the jQuery prototype for later instantiation
jQuery.fn.init.prototype = jQuery.fn;

jQuery.extend = jQuery.fn.extend = function() {
        var src, copyIsArray, copy, name, options, clone,
            target = arguments[0] || {},
            i = 1,
            length = arguments.length,
            deep = false;

        // Handle a deep copy situation
        if ( typeof target === "boolean" ) {
            deep = target;
            target = arguments[1] || {};
            // skip the boolean and the target
            i = 2;
        }

        // Handle case when target is a string or something (possible in deep
copy)
        if ( typeof target !== "object" && !jQuery.isFunction(target) ) {
            target = {};
        }
```

```
	// extend jQuery itself if only one argument is passed
	if ( length === i ) {
		target = this;
		--i;
	}

	for ( ; i < length; i++ ) {
		// Only deal with non-null/undefined values
		if ( (options = arguments[ i ]) != null ) {
			// Extend the base object
			for ( name in options ) {
				src = target[ name ];
				copy = options[ name ];

				// Prevent never-ending loop
				if ( target === copy ) {
					continue;
				}

				// Recurse if we're merging plain objects or arrays
				if ( deep && copy && ( jQuery.isPlainObject(copy) ||
(copyIsArray = jQuery.isArray(copy)) ) ) {
					if ( copyIsArray ) {
						copyIsArray = false;
						clone = src && jQuery.isArray(src) ? src :
[];

					} else {
						clone = src && jQuery.isPlainObject(src)
? src : {};
					}

					// Never move original objects, clone them
					target[ name ] = jQuery.extend( deep, clone,
copy );

				// Don't bring in undefined values
				} else if ( copy !== undefined ) {
```

```
                                target[ name ] = copy;
                        }
                }
        }
    }

    // Return the modified object
    return target;
};

jQuery.extend({
    // Unique for each copy of jQuery on the page
    // Non-digits removed to match rinlinejQuery
    expando: "jQuery" + ( core_version + Math.random() ).replace( /\D/g,
"" ),

    noConflict: function( deep ) {
        if ( window.$ === jQuery ) {
            window.$ = _$;
        }

        if ( deep && window.jQuery === jQuery ) {
            window.jQuery = _jQuery;
        }

        return jQuery;
    },

    // Is the DOM ready to be used? Set to true once it occurs.
    isReady: false,

    // A counter to track how many items to wait for before
    // the ready event fires. See #6781
    readyWait: 1,

    // Hold (or release) the ready event
    holdReady: function( hold ) {
        if ( hold ) {
```

```
                jQuery.readyWait++;
            } else {
                jQuery.ready( true );
            }
        },

        // Handle when the DOM is ready
        ready: function( wait ) {

            // Abort if there are pending holds or we're already ready
            if ( wait === true ? --jQuery.readyWait : jQuery.isReady ) {
                return;
            }

            // Make sure body exists, at least, in case IE gets a little
overzealous (ticket #5443).
            if ( !document.body ) {
                return setTimeout( jQuery.ready );
            }

            // Remember that the DOM is ready
            jQuery.isReady = true;

            // If a normal DOM Ready event fired, decrement, and wait if
need be
            if ( wait !== true && --jQuery.readyWait > 0 ) {
                return;
            }

            // If there are functions bound, to execute
            readyList.resolveWith( document, [ jQuery ] );

            // Trigger any bound ready events
            if ( jQuery.fn.trigger ) {
                jQuery( document ).trigger("ready").off("ready");
            }
        },
```

```
	// See test/unit/core.js for details concerning isFunction.
	// Since version 1.3, DOM methods and functions like alert
	// aren't supported. They return false on IE (#2968).
	isFunction: function( obj ) {
		return jQuery.type(obj) === "function";
	},

	isArray: Array.isArray || function( obj ) {
		return jQuery.type(obj) === "array";
	},

	isWindow: function( obj ) {
		/* jshint eqeqeq: false */
		return obj != null && obj == obj.window;
	},

	isNumeric: function( obj ) {
		return !isNaN( parseFloat(obj) ) && isFinite( obj );
	},

	type: function( obj ) {
		if ( obj == null ) {
			return String( obj );
		}
		return typeof obj === "object" || typeof obj === "function" ?
			class2type[ core_toString.call(obj) ] || "object" :
			typeof obj;
	},

	isPlainObject: function( obj ) {
		var key;

		// Must be an Object.
		// Because of IE, we also have to check the presence of the
constructor property.
		// Make sure that DOM nodes and window objects don't pass
through, as well
		if ( !obj || jQuery.type(obj) !== "object" || obj.nodeType ||
```

```
jQuery.isWindow( obj ) ) {
                return false;
        }

        try {
                // Not own constructor property must be Object
                if ( obj.constructor &&
                        !core_hasOwn.call(obj, "constructor") &&
                        !core_hasOwn.call(obj.constructor.prototype,
"isPrototypeOf") ) {
                        return false;
                }
        } catch ( e ) {
                // IE8,9 Will throw exceptions on certain host objects
#9897
                return false;
        }

        // Support: IE<9
        // Handle iteration over inherited properties before own
properties.
        if ( jQuery.support.ownLast ) {
                for ( key in obj ) {
                        return core_hasOwn.call( obj, key );
                }
        }

        // Own properties are enumerated firstly, so to speed up,
        // if last one is own, then all properties are own.
        for ( key in obj ) {}

        return key === undefined || core_hasOwn.call( obj, key );
    },

    isEmptyObject: function( obj ) {
        var name;
        for ( name in obj ) {
                return false;
```

```
		}
		return true;
	},

	error: function( msg ) {
		throw new Error( msg );
	},

	// data: string of html
	// context (optional): If specified, the fragment will be created in this
context, defaults to document
	// keepScripts (optional): If true, will include scripts passed in the html
string
	parseHTML: function( data, context, keepScripts ) {
		if ( !data || typeof data !== "string" ) {
			return null;
		}
		if ( typeof context === "boolean" ) {
			keepScripts = context;
			context = false;
		}
		context = context || document;

		var parsed = rsingleTag.exec( data ),
			scripts = !keepScripts && [];

		// Single tag
		if ( parsed ) {
			return [ context.createElement( parsed[1] ) ];
		}

		parsed = jQuery.buildFragment( [ data ], context, scripts );
		if ( scripts ) {
			jQuery( scripts ).remove();
		}
		return jQuery.merge( [], parsed.childNodes );
	},
```

```
	parseJSON: function( data ) {
		// Attempt to parse using the native JSON parser first
		if ( window.JSON && window.JSON.parse ) {
			return window.JSON.parse( data );
		}

		if ( data === null ) {
			return data;
		}

		if ( typeof data === "string" ) {

			// Make sure leading/trailing whitespace is removed (IE
can't handle it)
			data = jQuery.trim( data );

			if ( data ) {
				// Make sure the incoming data is actual JSON
				// Logic borrowed from http://json.org/json2.js
				if ( rvalidchars.test( data.replace( rvalidescape, "@"
)
					.replace( rvalidtokens, "]" )
					.replace( rvalidbraces, "")) ) {

					return ( new Function( "return " + data ) )();
				}
			}
		}

		jQuery.error( "Invalid JSON: " + data );
	},

	// Cross-browser xml parsing
	parseXML: function( data ) {
		var xml, tmp;
		if ( !data || typeof data !== "string" ) {
			return null;
		}
```

```
		try {
			if ( window.DOMParser ) { // Standard
				tmp = new DOMParser();
				xml = tmp.parseFromString( data , "text/xml" );
			} else { // IE
				xml = new ActiveXObject( "Microsoft.XMLDOM" );
				xml.async = "false";
				xml.loadXML( data );
			}
		} catch( e ) {
			xml = undefined;
		}
		if ( !xml || !xml.documentElement ||
xml.getElementsByTagName( "parsererror" ).length ) {
			jQuery.error( "Invalid XML: " + data );
		}
		return xml;
	},

	noop: function() {},

	// Evaluates a script in a global context
	// Workarounds based on findings by Jim Driscoll
	// http://weblogs.java.net/blog/driscoll/archive/2009/09/08/eval-
javascript-global-context
	globalEval: function( data ) {
		if ( data && jQuery.trim( data ) ) {
			// We use execScript on Internet Explorer
			// We use an anonymous function so that context is
window
			// rather than jQuery in Firefox
			( window.execScript || function( data ) {
				window[ "eval" ].call( window, data );
			} )( data );
		}
	},

	// Convert dashed to camelCase; used by the css and data modules
```

```
	// Microsoft forgot to hump their vendor prefix (#9572)
	camelCase: function( string ) {
		return string.replace( rmsPrefix, "ms-" ).replace( rdashAlpha,
fcamelCase );
	},

	nodeName: function( elem, name ) {
		return elem.nodeName && elem.nodeName.toLowerCase() ===
name.toLowerCase();
	},

	// args is for internal usage only
	each: function( obj, callback, args ) {
		var value,
			i = 0,
			length = obj.length,
			isArray = isArraylike( obj );

		if ( args ) {
			if ( isArray ) {
				for ( ; i < length; i++ ) {
					value = callback.apply( obj[ i ], args );

					if ( value === false ) {
						break;
					}
				}
			} else {
				for ( i in obj ) {
					value = callback.apply( obj[ i ], args );

					if ( value === false ) {
						break;
					}
				}
			}

		// A special, fast, case for the most common use of each
```

```
		} else {
			if ( isArray ) {
				for ( ; i < length; i++ ) {
					value = callback.call( obj[ i ], i, obj[ i ] );

					if ( value === false ) {
						break;
					}
				}
			} else {
				for ( i in obj ) {
					value = callback.call( obj[ i ], i, obj[ i ] );

					if ( value === false ) {
						break;
					}
				}
			}
		}

		return obj;
	},

	// Use native String.trim function wherever possible
	trim: core_trim && !core_trim.call("\uFEFF\xA0") ?
		function( text ) {
			return text == null ?
				"" :
				core_trim.call( text );
		} :

		// Otherwise use our own trimming functionality
		function( text ) {
			return text == null ?
				"" :
				( text + "" ).replace( rtrim, "" );
		},
```

```
	// results is for internal usage only
	makeArray: function( arr, results ) {
		var ret = results || [];

		if ( arr != null ) {
			if ( isArraylike( Object(arr) ) ) {
				jQuery.merge( ret,
					typeof arr === "string" ?
					[ arr ] : arr
				);
			} else {
				core_push.call( ret, arr );
			}
		}

		return ret;
	},

	inArray: function( elem, arr, i ) {
		var len;

		if ( arr ) {
			if ( core_indexOf ) {
				return core_indexOf.call( arr, elem, i );
			}

			len = arr.length;
			i = i ? i < 0 ? Math.max( 0, len + i ) : i : 0;

			for ( ; i < len; i++ ) {
				// Skip accessing in sparse arrays
				if ( i in arr && arr[ i ] === elem ) {
					return i;
				}
			}
		}

		return -1;
```

```
	},

	merge: function( first, second ) {
		var l = second.length,
			i = first.length,
			j = 0;

		if ( typeof l === "number" ) {
			for ( ; j < l; j++ ) {
				first[ i++ ] = second[ j ];
			}
		} else {
			while ( second[j] !== undefined ) {
				first[ i++ ] = second[ j++ ];
			}
		}

		first.length = i;

		return first;
	},

	grep: function( elems, callback, inv ) {
		var retVal,
			ret = [],
			i = 0,
			length = elems.length;
		inv = !!inv;

		// Go through the array, only saving the items
		// that pass the validator function
		for ( ; i < length; i++ ) {
			retVal = !!callback( elems[ i ], i );
			if ( inv !== retVal ) {
				ret.push( elems[ i ] );
			}
		}
```

```
            return ret;
        },

        // arg is for internal usage only
        map: function( elems, callback, arg ) {
            var value,
                i = 0,
                length = elems.length,
                isArray = isArraylike( elems ),
                ret = [];

            // Go through the array, translating each of the items to their
            if ( isArray ) {
                for ( ; i < length; i++ ) {
                    value = callback( elems[ i ], i, arg );

                    if ( value != null ) {
                        ret[ ret.length ] = value;
                    }
                }

            // Go through every key on the object,
            } else {
                for ( i in elems ) {
                    value = callback( elems[ i ], i, arg );

                    if ( value != null ) {
                        ret[ ret.length ] = value;
                    }
                }
            }

            // Flatten any nested arrays
            return core_concat.apply( [], ret );
        },

        // A global GUID counter for objects
        guid: 1,
```

```
	// Bind a function to a context, optionally partially applying any
	// arguments.
	proxy: function( fn, context ) {
		var args, proxy, tmp;

		if ( typeof context === "string" ) {
			tmp = fn[ context ];
			context = fn;
			fn = tmp;
		}

		// Quick check to determine if target is callable, in the spec
		// this throws a TypeError, but we will just return undefined.
		if ( !jQuery.isFunction( fn ) ) {
			return undefined;
		}

		// Simulated bind
		args = core_slice.call( arguments, 2 );
		proxy = function() {
			return fn.apply( context || this, args.concat(
core_slice.call( arguments ) ) );
		};

		// Set the guid of unique handler to the same of original handler,
so it can be removed
		proxy.guid = fn.guid = fn.guid || jQuery.guid++;

		return proxy;
	},

	// Multifunctional method to get and set values of a collection
	// The value/s can optionally be executed if it's a function
	access: function( elems, fn, key, value, chainable, emptyGet, raw ) {
		var i = 0,
			length = elems.length,
			bulk = key == null;
```

```
		// Sets many values
		if ( jQuery.type( key ) === "object" ) {
			chainable = true;
			for ( i in key ) {
				jQuery.access( elems, fn, i, key[i], true, emptyGet,
raw );
			}

		// Sets one value
		} else if ( value !== undefined ) {
			chainable = true;

			if ( !jQuery.isFunction( value ) ) {
				raw = true;
			}

			if ( bulk ) {
				// Bulk operations run against the entire set
				if ( raw ) {
					fn.call( elems, value );
					fn = null;

				// ...except when executing function values
				} else {
					bulk = fn;
					fn = function( elem, key, value ) {
						return bulk.call( jQuery( elem ), value );
					};
				}
			}

			if ( fn ) {
				for ( ; i < length; i++ ) {
					fn( elems[i], key, raw ? value : value.call(
elems[i], i, fn( elems[i], key ) ) );
				}
			}
```

```
		}

		return chainable ?
			elems :

			// Gets
			bulk ?
				fn.call( elems ) :
				length ? fn( elems[0], key ) : emptyGet;
	},

	now: function() {
		return ( new Date() ).getTime();
	},

	// A method for quickly swapping in/out CSS properties to get correct
calculations.
	// Note: this method belongs to the css module but it's needed here
for the support module.
	// If support gets modularized, this method should be moved back to
the css module.
	swap: function( elem, options, callback, args ) {
		var ret, name,
			old = {};

		// Remember the old values, and insert the new ones
		for ( name in options ) {
			old[ name ] = elem.style[ name ];
			elem.style[ name ] = options[ name ];
		}

		ret = callback.apply( elem, args || [] );

		// Revert the old values
		for ( name in options ) {
			elem.style[ name ] = old[ name ];
		}
```

```
		return ret;
	}
});

jQuery.ready.promise = function( obj ) {
	if ( !readyList ) {

		readyList = jQuery.Deferred();

		// Catch cases where $(document).ready() is called after the
browser event has already occurred.
		// we once tried to use readyState "interactive" here, but it
caused issues like the one
		// discovered by ChrisS here:
http://bugs.jquery.com/ticket/12282#comment:15
		if ( document.readyState === "complete" ) {
			// Handle it asynchronously to allow scripts the opportunity
to delay ready
			setTimeout( jQuery.ready );

		// Standards-based browsers support DOMContentLoaded
		} else if ( document.addEventListener ) {
			// Use the handy event callback
			document.addEventListener( "DOMContentLoaded",
completed, false );

			// A fallback to window.onload, that will always work
			window.addEventListener( "load", completed, false );

		// If IE event model is used
		} else {
			// Ensure firing before onload, maybe late but safe also for
iframes
			document.attachEvent( "onreadystatechange", completed
);

			// A fallback to window.onload, that will always work
			window.attachEvent( "onload", completed );
```

```
                    // If IE and not a frame
                    // continually check to see if the document is ready
                    var top = false;

                    try {
                            top = window.frameElement == null &&
document.documentElement;
                    } catch(e) {}

                    if ( top && top.doScroll ) {
                            (function doScrollCheck() {
                                    if ( !jQuery.isReady ) {

                                            try {
                                                    // Use the trick by Diego Perini
                                                    //
http://javascript.nwbox.com/IEContentLoaded/
                                                    top.doScroll("left");
                                            } catch(e) {
                                                    return setTimeout( doScrollCheck,
50 );
                                            }

                                            // detach all dom ready events
                                            detach();

                                            // and execute any waiting functions
                                            jQuery.ready();
                                    }
                            })();
                    }
            }
    }
    return readyList.promise( obj );
};

// Populate the class2type map
```

```
jQuery.each("Boolean Number String Function Array Date RegExp Object
Error".split(" "), function(i, name) {
	class2type[ "[object " + name + "]" ] = name.toLowerCase();
});

function isArraylike( obj ) {
	var length = obj.length,
		type = jQuery.type( obj );

	if ( jQuery.isWindow( obj ) ) {
		return false;
	}

	if ( obj.nodeType === 1 && length ) {
		return true;
	}

	return type === "array" || type !== "function" &&
		( length === 0 ||
		typeof length === "number" && length > 0 && ( length - 1 ) in
obj );
}

// All jQuery objects should point back to these
rootjQuery = jQuery(document);
/*!
 * Sizzle CSS Selector Engine v1.10.2
 * http://sizzlejs.com/
 *
 * Copyright 2013 jQuery Foundation, Inc. and other contributors
 * Released under the MIT license
 * http://jquery.org/license
 *
 * Date: 2013-07-03
 */
(function( window, undefined ) {

var i,
```

```
support,
cachedruns,
Expr,
getText,
isXML,
compile,
outermostContext,
sortInput,

// Local document vars
setDocument,
document,
docElem,
documentIsHTML,
rbuggyQSA,
rbuggyMatches,
matches,
contains,

// Instance-specific data
expando = "sizzle" + -(new Date()),
preferredDoc = window.document,
dirruns = 0,
done = 0,
classCache = createCache(),
tokenCache = createCache(),
compilerCache = createCache(),
hasDuplicate = false,
sortOrder = function( a, b ) {
	if ( a === b ) {
		hasDuplicate = true;
		return 0;
	}
	return 0;
},

// General-purpose constants
strundefined = typeof undefined,
```

```
	MAX_NEGATIVE = 1 << 31,

	// Instance methods
	hasOwn = ({}).hasOwnProperty,
	arr = [],
	pop = arr.pop,
	push_native = arr.push,
	push = arr.push,
	slice = arr.slice,
	// Use a stripped-down indexOf if we can't use a native one
	indexOf = arr.indexOf || function( elem ) {
		var i = 0,
			len = this.length;
		for ( ; i < len; i++ ) {
			if ( this[i] === elem ) {
				return i;
			}
		}
		return -1;
	},

	booleans =
"checked|selected|async|autofocus|autoplay|controls|defer|disabled|hidden|
ismap|loop|multiple|open|readonly|required|scoped",

	// Regular expressions

	// Whitespace characters http://www.w3.org/TR/css3-
selectors/#whitespace
	whitespace = "[\\x20\\t\\r\\n\\f]",
	// http://www.w3.org/TR/css3-syntax/#characters
	characterEncoding = "(?:\\\\.|[\\w-]|[^\\x00-\\xa0])+",

	// Loosely modeled on CSS identifier characters
	// An unquoted value should be a CSS identifier
http://www.w3.org/TR/css3-selectors/#attribute-selectors
	// Proper syntax: http://www.w3.org/TR/CSS21/syndata.html#value-
def-identifier
```

```
	identifier = characterEncoding.replace( "w", "w#" ),

	// Acceptable operators http://www.w3.org/TR/selectors/#attribute-
selectors
	attributes = "\\[" + whitespace + "*(" + characterEncoding + ")" +
whitespace +
		"*(?:([*^$|!~]?=)" + whitespace +
"*(?:(['\"])((?:\\\\.|[^\\\\])*?)\\3|(" + identifier + ")|)|)" + whitespace +
"*\\]",

	// Prefer arguments quoted,
	//   then not containing pseudos/brackets,
	//   then attribute selectors/non-parenthetical expressions,
	//   then anything else
	// These preferences are here to reduce the number of selectors
	//   needing tokenize in the PSEUDO preFilter
	pseudos = ":(" + characterEncoding +
")(?:\\(((['\"])((?:\\\\.|[^\\\\])*?)\\3|((?:\\\\.|[^\\\\()[\\]]|" +
attributes.replace( 3, 8 ) + ")*)|.*)\\)|)",

	// Leading and non-escaped trailing whitespace, capturing some non-
whitespace characters preceding the latter
	rtrim = new RegExp( "^" + whitespace + "+|((?:^|[^\\\\])(?:\\\\.)*)"
+ whitespace + "+$", "g" ),

	rcomma = new RegExp( "^" + whitespace + "*," + whitespace + "*"
),
	rcombinators = new RegExp( "^" + whitespace + "*([>+~]|" +
whitespace + ")" + whitespace + "*" ),

	rsibling = new RegExp( whitespace + "*[+~]" ),
	rattributeQuotes = new RegExp( "=" + whitespace + "*([^\\]'\"]*)" +
whitespace + "*\\]", "g" ),

	rpseudo = new RegExp( pseudos ),
	ridentifier = new RegExp( "^" + identifier + "$" ),

	matchExpr = {
```

```
		"ID": new RegExp( "^#(" + characterEncoding + ")" ),
		"CLASS": new RegExp( "^\\.(" + characterEncoding + ")" ),
		"TAG": new RegExp( "^(" + characterEncoding.replace( "w",
"w*" ) + ")" ),
		"ATTR": new RegExp( "^" + attributes ),
		"PSEUDO": new RegExp( "^" + pseudos ),
		"CHILD": new RegExp( "^:(only|first|last|nth|nth-last)-(child|of-
type)(?:\\(" + whitespace +
			"*(even|odd|(([+-]|)(\\d*)n|)" + whitespace + "*(?:([+-
]|)" + whitespace +
			"*(\\d+)|))" + whitespace + "*\\)|)", "i" ),
		"bool": new RegExp( "^(?:" + booleans + ")$", "i" ),
		// For use in libraries implementing .is()
		// We use this for POS matching in `select`
		"needsContext": new RegExp( "^" + whitespace +
"*[>+~]|:(even|odd|eq|gt|lt|nth|first|last)(?:\\(" +
			whitespace + "*((?:-\\d)?\\d*)" + whitespace +
"*\\)|)(?=[^-]|$)", "i" )
	},

	rnative = /^[^{]+\{\s*\[native \w/,

	// Easily-parseable/retrievable ID or TAG or CLASS selectors
	rquickExpr = /^(?:#([\w-]+)|(\w+)|\.([\w-]+))$/,

	rinputs = /^(?:input|select|textarea|button)$/i,
	rheader = /^h\d$/i,

	rescape = /'|\\/g,

	// CSS escapes http://www.w3.org/TR/CSS21/syndata.html#escaped-
characters
	runescape = new RegExp( "\\\\([\\da-f]{1,6}" + whitespace + "?|(" +
whitespace + ")|.)", "ig" ),
	funescape = function( _, escaped, escapedWhitespace ) {
		var high = "0x" + escaped - 0x10000;
		// NaN means non-codepoint
		// Support: Firefox
```

```
		// Workaround erroneous numeric interpretation of +"0x"
		return high !== high || escapedWhitespace ?
			escaped :
			// BMP codepoint
			high < 0 ?
				String.fromCharCode( high + 0x10000 ) :
				// Supplemental Plane codepoint (surrogate pair)
				String.fromCharCode( high >> 10 | 0xD800, high &
0x3FF | 0xDC00 );
	};

// Optimize for push.apply( _, NodeList )
try {
	push.apply(
		(arr = slice.call( preferredDoc.childNodes )),
		preferredDoc.childNodes
	);
	// Support: Android<4.0
	// Detect silently failing push.apply
	arr[ preferredDoc.childNodes.length ].nodeType;
} catch ( e ) {
	push = { apply: arr.length ?

		// Leverage slice if possible
		function( target, els ) {
			push_native.apply( target, slice.call(els) );
		} :

		// Support: IE<9
		// Otherwise append directly
		function( target, els ) {
			var j = target.length,
				i = 0;
			// Can't trust NodeList.length
			while ( (target[j++] = els[i++]) ) {}
			target.length = j - 1;
		}
	};
```

```
}

function Sizzle( selector, context, results, seed ) {
	var match, elem, m, nodeType,
		// QSA vars
		i, groups, old, nid, newContext, newSelector;

	if ( ( context ? context.ownerDocument || context : preferredDoc )
!== document ) {
		setDocument( context );
	}

	context = context || document;
	results = results || [];

	if ( !selector || typeof selector !== "string" ) {
		return results;
	}

	if ( (nodeType = context.nodeType) !== 1 && nodeType !== 9 ) {
		return [];
	}

	if ( documentIsHTML && !seed ) {

		// Shortcuts
		if ( (match = rquickExpr.exec( selector )) ) {
			// Speed-up: Sizzle("#ID")
			if ( (m = match[1]) ) {
				if ( nodeType === 9 ) {
					elem = context.getElementById( m );
					// Check parentNode to catch when Blackberry
4.6 returns
					// nodes that are no longer in the document
#6963
					if ( elem && elem.parentNode ) {
						// Handle the case where IE, Opera, and
Webkit return items
```

```
                                        // by name instead of ID
                                        if ( elem.id === m ) {
                                                results.push( elem );
                                                return results;
                                        }
                                } else {
                                        return results;
                                }
                        } else {
                                // Context is not a document
                                if ( context.ownerDocument && (elem =
context.ownerDocument.getElementById( m )) &&
                                        contains( context, elem ) && elem.id
=== m ) {
                                        results.push( elem );
                                        return results;
                                }
                        }

                // Speed-up: Sizzle("TAG")
                } else if ( match[2] ) {
                        push.apply( results,
context.getElementsByTagName( selector ) );
                        return results;

                // Speed-up: Sizzle(".CLASS")
                } else if ( (m = match[3]) &&
support.getElementsByClassName && context.getElementsByClassName ) {
                        push.apply( results,
context.getElementsByClassName( m ) );
                        return results;
                }
        }

        // QSA path
        if ( support.qsa && (!rbuggyQSA || !rbuggyQSA.test( selector ))
) {
                nid = old = expando;
```

```
                    newContext = context;
                    newSelector = nodeType === 9 && selector;

                    // qSA works strangely on Element-rooted queries
                    // We can work around this by specifying an extra ID on
the root
                    // and working up from there (Thanks to Andrew Dupont
for the technique)
                    // IE 8 doesn't work on object elements
                    if ( nodeType === 1 && context.nodeName.toLowerCase()
!== "object" ) {
                        groups = tokenize( selector );

                        if ( (old = context.getAttribute("id")) ) {
                            nid = old.replace( rescape, "\\$&" );
                        } else {
                            context.setAttribute( "id", nid );
                        }
                        nid = "[id='" + nid + "'] ";

                        i = groups.length;
                        while ( i-- ) {
                            groups[i] = nid + toSelector( groups[i] );
                        }
                        newContext = rsibling.test( selector ) &&
context.parentNode || context;
                        newSelector = groups.join(",");
                    }

                    if ( newSelector ) {
                        try {
                            push.apply( results,
                                    newContext.querySelectorAll(
newSelector )
                            );
                            return results;
                        } catch(qsaError) {
                        } finally {
```

```
                                        if ( !old ) {
                                                context.removeAttribute("id");
                                        }
                                }
                        }
                }
        }

        // All others
        return select( selector.replace( rtrim, "$1" ), context, results, seed );
}

/**
 * Create key-value caches of limited size
 * @returns {Function(string, Object)} Returns the Object data after storing
it on itself with
 *      property name the (space-suffixed) string and (if the cache is larger
than Expr.cacheLength)
 *      deleting the oldest entry
 */
function createCache() {
        var keys = [];

        function cache( key, value ) {
                // Use (key + " ") to avoid collision with native prototype
properties (see Issue #157)
                if ( keys.push( key += " " ) > Expr.cacheLength ) {
                        // Only keep the most recent entries
                        delete cache[ keys.shift() ];
                }
                return (cache[ key ] = value);
        }
        return cache;
}

/**
 * Mark a function for special use by Sizzle
 * @param {Function} fn The function to mark
```

```
 */
function markFunction( fn ) {
	fn[ expando ] = true;
	return fn;
}

/**
 * Support testing using an element
 * @param {Function} fn Passed the created div and expects a boolean
result
 */
function assert( fn ) {
	var div = document.createElement("div");

	try {
		return !!fn( div );
	} catch (e) {
		return false;
	} finally {
		// Remove from its parent by default
		if ( div.parentNode ) {
			div.parentNode.removeChild( div );
		}
		// release memory in IE
		div = null;
	}
}

/**
 * Adds the same handler for all of the specified attrs
 * @param {String} attrs Pipe-separated list of attributes
 * @param {Function} handler The method that will be applied
 */
function addHandle( attrs, handler ) {
	var arr = attrs.split("|"),
		i = attrs.length;

	while ( i-- ) {
```

```
		Expr.attrHandle[ arr[i] ] = handler;
	}
}

/**
 * Checks document order of two siblings
 * @param {Element} a
 * @param {Element} b
 * @returns {Number} Returns less than 0 if a precedes b, greater than 0 if
a follows b
 */
function siblingCheck( a, b ) {
	var cur = b && a,
		diff = cur && a.nodeType === 1 && b.nodeType === 1 &&
			( ~b.sourceIndex || MAX_NEGATIVE ) -
			( ~a.sourceIndex || MAX_NEGATIVE );

	// Use IE sourceIndex if available on both nodes
	if ( diff ) {
		return diff;
	}

	// Check if b follows a
	if ( cur ) {
		while ( (cur = cur.nextSibling) ) {
			if ( cur === b ) {
				return -1;
			}
		}
	}

	return a ? 1 : -1;
}

/**
 * Returns a function to use in pseudos for input types
 * @param {String} type
 */
```

```
function createInputPseudo( type ) {
	return function( elem ) {
		var name = elem.nodeName.toLowerCase();
		return name === "input" && elem.type === type;
	};
}

/**
 * Returns a function to use in pseudos for buttons
 * @param {String} type
 */
function createButtonPseudo( type ) {
	return function( elem ) {
		var name = elem.nodeName.toLowerCase();
		return (name === "input" || name === "button") && elem.type
=== type;
	};
}

/**
 * Returns a function to use in pseudos for positionals
 * @param {Function} fn
 */
function createPositionalPseudo( fn ) {
	return markFunction(function( argument ) {
		argument = +argument;
		return markFunction(function( seed, matches ) {
			var j,
				matchIndexes = fn( [], seed.length, argument ),
				i = matchIndexes.length;

			// Match elements found at the specified indexes
			while ( i-- ) {
				if ( seed[ (j = matchIndexes[i]) ] ) {
					seed[j] = !(matches[j] = seed[j]);
				}
			}
		});
```

```
	});
}

/**
 * Detect xml
 * @param {Element|Object} elem An element or a document
 */
isXML = Sizzle.isXML = function( elem ) {
	// documentElement is verified for cases where it doesn't yet exist
	// (such as loading iframes in IE - #4833)
	var documentElement = elem && (elem.ownerDocument ||
elem).documentElement;
	return documentElement ? documentElement.nodeName !== "HTML" :
false;
};

// Expose support vars for convenience
support = Sizzle.support = {};

/**
 * Sets document-related variables once based on the current document
 * @param {Element|Object} [doc] An element or document object to use
to set the document
 * @returns {Object} Returns the current document
 */
setDocument = Sizzle.setDocument = function( node ) {
	var doc = node ? node.ownerDocument || node : preferredDoc,
		parent = doc.defaultView;

	// If no document and documentElement is available, return
	if ( doc === document || doc.nodeType !== 9 ||
!doc.documentElement ) {
		return document;
	}

	// Set our document
	document = doc;
	docElem = doc.documentElement;
```

```
	// Support tests
	documentIsHTML = !isXML( doc );

	// Support: IE>8
	// If iframe document is assigned to "document" variable and if iframe
has been reloaded,
	// IE will throw "permission denied" error when accessing "document"
variable, see jQuery #13936
	// IE6-8 do not support the defaultView property so parent will be
undefined
	if ( parent && parent.attachEvent && parent !== parent.top ) {
		parent.attachEvent( "onbeforeunload", function() {
			setDocument();
		});
	}

	/* Attributes
	---------------------------------------------------------------------- */

	// Support: IE<8
	// Verify that getAttribute really returns attributes and not properties
(excepting IE8 booleans)
	support.attributes = assert(function( div ) {
		div.className = "i";
		return !div.getAttribute("className");
	});

	/* getElement(s)By*
	---------------------------------------------------------------------- */

	// Check if getElementsByTagName("*") returns only elements
	support.getElementsByTagName = assert(function( div ) {
		div.appendChild( doc.createComment("") );
		return !div.getElementsByTagName("*").length;
	});

	// Check if getElementsByClassName can be trusted
```

```
	support.getElementsByClassName = assert(function( div ) {
		div.innerHTML = "<div class='a'></div><div class='a
i'></div>";

		// Support: Safari<4
		// Catch class over-caching
		div.firstChild.className = "i";
		// Support: Opera<10
		// Catch gEBCN failure to find non-leading classes
		return div.getElementsByClassName("i").length === 2;
	});

	// Support: IE<10
	// Check if getElementById returns elements by name
	// The broken getElementById methods don't pick up programatically-
set names,
	// so use a roundabout getElementsByName test
	support.getById = assert(function( div ) {
		docElem.appendChild( div ).id = expando;
		return !doc.getElementsByName || !doc.getElementsByName(
expando ).length;
	});

	// ID find and filter
	if ( support.getById ) {
		Expr.find["ID"] = function( id, context ) {
			if ( typeof context.getElementById !== strundefined &&
documentIsHTML ) {
				var m = context.getElementById( id );
				// Check parentNode to catch when Blackberry 4.6
returns
				// nodes that are no longer in the document #6963
				return m && m.parentNode ? [m] : [];
			}
		};
		Expr.filter["ID"] = function( id ) {
			var attrId = id.replace( runescape, funescape );
			return function( elem ) {
```

```
                        return elem.getAttribute("id") === attrId;
                    };
                };
        } else {
                // Support: IE6/7
                // getElementById is not reliable as a find shortcut
                delete Expr.find["ID"];

                Expr.filter["ID"] =  function( id ) {
                        var attrId = id.replace( runescape, funescape );
                        return function( elem ) {
                                var node = typeof elem.getAttributeNode !==
strundefined && elem.getAttributeNode("id");
                                return node && node.value === attrId;
                        };
                };
        }

        // Tag
        Expr.find["TAG"] = support.getElementsByTagName ?
                function( tag, context ) {
                        if ( typeof context.getElementsByTagName !==
strundefined ) {
                                return context.getElementsByTagName( tag );
                        }
                } :
                function( tag, context ) {
                        var elem,
                                tmp = [],
                                i = 0,
                                results = context.getElementsByTagName( tag );

                        // Filter out possible comments
                        if ( tag === "*" ) {
                                while ( (elem = results[i++]) ) {
                                        if ( elem.nodeType === 1 ) {
                                                tmp.push( elem );
                                        }
```

```
                        }

                        return tmp;
                }
                return results;
        };

    // Class
    Expr.find["CLASS"] = support.getElementsByClassName && function(
className, context ) {
        if ( typeof context.getElementsByClassName !== strundefined
&& documentIsHTML ) {
                return context.getElementsByClassName( className );
        }
    };

    /* QSA/matchesSelector
    ---------------------------------------------------------------------- */

    // QSA and matchesSelector support

    // matchesSelector(:active) reports false when true (IE9/Opera 11.5)
    rbuggyMatches = [];

    // qSa(:focus) reports false when true (Chrome 21)
    // We allow this because of a bug in IE8/9 that throws an error
    // whenever `document.activeElement` is accessed on an iframe
    // So, we allow :focus to pass through QSA all the time to avoid the IE
error
    // See http://bugs.jquery.com/ticket/13378
    rbuggyQSA = [];

    if ( (support.qsa = rnative.test( doc.querySelectorAll )) ) {
        // Build QSA regex
        // Regex strategy adopted from Diego Perini
        assert(function( div ) {
                // Select is set to empty string on purpose
                // This is to test IE's treatment of not explicitly
```

```
                    // setting a boolean content attribute,
                    // since its presence should be enough
                    // http://bugs.jquery.com/ticket/12359
                    div.innerHTML = "<select><option
selected=''></option></select>";

                    // Support: IE8
                    // Boolean attributes and "value" are not treated correctly
                    if ( !div.querySelectorAll("[selected]").length ) {
                          rbuggyQSA.push( "\\[" + whitespace + "*(?:value|"
+ booleans + ")" );
                    }

                    // Webkit/Opera - :checked should return selected option
elements
                    // http://www.w3.org/TR/2011/REC-css3-selectors-
20110929/#checked
                    // IE8 throws error here and will not see later tests
                    if ( !div.querySelectorAll(":checked").length ) {
                          rbuggyQSA.push(":checked");
                    }
              });

              assert(function( div ) {

                    // Support: Opera 10-12/IE8
                    // ^= $= *= and empty values
                    // Should not select anything
                    // Support: Windows 8 Native Apps
                    // The type attribute is restricted during .innerHTML
assignment
                    var input = doc.createElement("input");
                    input.setAttribute( "type", "hidden" );
                    div.appendChild( input ).setAttribute( "t", "" );

                    if ( div.querySelectorAll("[t^='']").length ) {
                          rbuggyQSA.push( "[*^$]=" + whitespace +
"*(?:''|\"\")" );
```

```
                }

                // FF 3.5 - :enabled/:disabled and hidden elements
(hidden elements are still enabled)
                // IE8 throws error here and will not see later tests
                if ( !div.querySelectorAll(":enabled").length ) {
                        rbuggyQSA.push( ":enabled", ":disabled" );
                }

                // Opera 10-11 does not throw on post-comma invalid
pseudos
                div.querySelectorAll("*,:x");
                rbuggyQSA.push(",.*:");
        });
    }

    if ( (support.matchesSelector = rnative.test( (matches =
docElem.webkitMatchesSelector ||
        docElem.mozMatchesSelector ||
        docElem.oMatchesSelector ||
        docElem.msMatchesSelector) )) ) {

        assert(function( div ) {
                // Check to see if it's possible to do matchesSelector
                // on a disconnected node (IE 9)
                support.disconnectedMatch = matches.call( div, "div" );

                // This should fail with an exception
                // Gecko does not error, returns false instead
                matches.call( div, "[s!='']:x" );
                rbuggyMatches.push( "!=", pseudos );
        });
    }

    rbuggyQSA = rbuggyQSA.length && new RegExp( rbuggyQSA.join("|")
);
    rbuggyMatches = rbuggyMatches.length && new RegExp(
rbuggyMatches.join("|") );
```

```
	/* Contains
	---------------------------------------------------------------------- */

	// Element contains another
	// Purposefully does not implement inclusive descendent
	// As in, an element does not contain itself
	contains = rnative.test( docElem.contains ) ||
docElem.compareDocumentPosition ?
		function( a, b ) {
			var adown = a.nodeType === 9 ? a.documentElement : a,
				bup = b && b.parentNode;
			return a === bup || !!( bup && bup.nodeType === 1 && (
				adown.contains ?
					adown.contains( bup ) :
					a.compareDocumentPosition &&
a.compareDocumentPosition( bup ) & 16
			));
		} :
		function( a, b ) {
			if ( b ) {
				while ( (b = b.parentNode) ) {
					if ( b === a ) {
						return true;
					}
				}
			}
			return false;
		};

	/* Sorting
	---------------------------------------------------------------------- */

	// Document order sorting
	sortOrder = docElem.compareDocumentPosition ?
	function( a, b ) {

		// Flag for duplicate removal
```

```
		if ( a === b ) {
			hasDuplicate = true;
			return 0;
		}

		var compare = b.compareDocumentPosition &&
a.compareDocumentPosition && a.compareDocumentPosition( b );

		if ( compare ) {
			// Disconnected nodes
			if ( compare & 1 ||
				(!support.sortDetached &&
b.compareDocumentPosition( a ) === compare) ) {

				// Choose the first element that is related to our
preferred document
				if ( a === doc || contains(preferredDoc, a) ) {
					return -1;
				}
				if ( b === doc || contains(preferredDoc, b) ) {
					return 1;
				}

				// Maintain original order
				return sortInput ?
					( indexOf.call( sortInput, a ) - indexOf.call(
sortInput, b ) ) :
					0;
			}

			return compare & 4 ? -1 : 1;
		}

		// Not directly comparable, sort on existence of method
		return a.compareDocumentPosition ? -1 : 1;
	} :
	function( a, b ) {
		var cur,
```

```
                i = 0,
                aup = a.parentNode,
                bup = b.parentNode,
                ap = [ a ],
                bp = [ b ];

        // Exit early if the nodes are identical
        if ( a === b ) {
                hasDuplicate = true;
                return 0;

        // Parentless nodes are either documents or disconnected
        } else if ( !aup || !bup ) {
                return a === doc ? -1 :
                        b === doc ? 1 :
                        aup ? -1 :
                        bup ? 1 :
                        sortInput ?
                        ( indexOf.call( sortInput, a ) - indexOf.call(
sortInput, b ) ) :
                        0;

        // If the nodes are siblings, we can do a quick check
        } else if ( aup === bup ) {
                return siblingCheck( a, b );
        }

        // Otherwise we need full lists of their ancestors for comparison
        cur = a;
        while ( (cur = cur.parentNode) ) {
                ap.unshift( cur );
        }
        cur = b;
        while ( (cur = cur.parentNode) ) {
                bp.unshift( cur );
        }

        // Walk down the tree looking for a discrepancy
```

```
		while ( ap[i] === bp[i] ) {
			i++;
		}

		return i ?
			// Do a sibling check if the nodes have a common ancestor
			siblingCheck( ap[i], bp[i] ) :

			// Otherwise nodes in our document sort first
			ap[i] === preferredDoc ? -1 :
			bp[i] === preferredDoc ? 1 :
			0;
	};

	return doc;
};

Sizzle.matches = function( expr, elements ) {
	return Sizzle( expr, null, null, elements );
};

Sizzle.matchesSelector = function( elem, expr ) {
	// Set document vars if needed
	if ( ( elem.ownerDocument || elem ) !== document ) {
		setDocument( elem );
	}

	// Make sure that attribute selectors are quoted
	expr = expr.replace( rattributeQuotes, "='$1']" );

	if ( support.matchesSelector && documentIsHTML &&
		( !rbuggyMatches || !rbuggyMatches.test( expr ) ) &&
		( !rbuggyQSA     || !rbuggyQSA.test( expr ) ) ) {

		try {
			var ret = matches.call( elem, expr );

			// IE 9's matchesSelector returns false on disconnected
```

```
nodes
                    if ( ret || support.disconnectedMatch ||
                                // As well, disconnected nodes are said to be in
a document
                                // fragment in IE 9
                                elem.document && elem.document.nodeType
!== 11 ) {
                        return ret;
                    }
                } catch(e) {}
        }

        return Sizzle( expr, document, null, [elem] ).length > 0;
};

Sizzle.contains = function( context, elem ) {
        // Set document vars if needed
        if ( ( context.ownerDocument || context ) !== document ) {
                setDocument( context );
        }
        return contains( context, elem );
};

Sizzle.attr = function( elem, name ) {
        // Set document vars if needed
        if ( ( elem.ownerDocument || elem ) !== document ) {
                setDocument( elem );
        }

        var fn = Expr.attrHandle[ name.toLowerCase() ],
                // Don't get fooled by Object.prototype properties (jQuery
#13807)
                val = fn && hasOwn.call( Expr.attrHandle, name.toLowerCase() )
?
                        fn( elem, name, !documentIsHTML ) :
                        undefined;

        return val === undefined ?
```

```
		support.attributes || !documentIsHTML ?
			elem.getAttribute( name ) :
			(val = elem.getAttributeNode(name)) && val.specified ?
				val.value :
				null :
		val;
};

Sizzle.error = function( msg ) {
	throw new Error( "Syntax error, unrecognized expression: " + msg );
};

/**
 * Document sorting and removing duplicates
 * @param {ArrayLike} results
 */
Sizzle.uniqueSort = function( results ) {
	var elem,
		duplicates = [],
		j = 0,
		i = 0;

	// Unless we *know* we can detect duplicates, assume their presence
	hasDuplicate = !support.detectDuplicates;
	sortInput = !support.sortStable && results.slice( 0 );
	results.sort( sortOrder );

	if ( hasDuplicate ) {
		while ( (elem = results[i++]) ) {
			if ( elem === results[ i ] ) {
				j = duplicates.push( i );
			}
		}
		while ( j-- ) {
			results.splice( duplicates[ j ], 1 );
		}
	}
```

```
	return results;
};

/**
 * Utility function for retrieving the text value of an array of DOM nodes
 * @param {Array|Element} elem
 */
getText = Sizzle.getText = function( elem ) {
	var node,
		ret = "",
		i = 0,
		nodeType = elem.nodeType;

	if ( !nodeType ) {
		// If no nodeType, this is expected to be an array
		for ( ; (node = elem[i]); i++ ) {
			// Do not traverse comment nodes
			ret += getText( node );
		}
	} else if ( nodeType === 1 || nodeType === 9 || nodeType === 11 )
{
		// Use textContent for elements
		// innerText usage removed for consistency of new lines (see
#11153)
		if ( typeof elem.textContent === "string" ) {
			return elem.textContent;
		} else {
			// Traverse its children
			for ( elem = elem.firstChild; elem; elem =
elem.nextSibling ) {
				ret += getText( elem );
			}
		}
	} else if ( nodeType === 3 || nodeType === 4 ) {
		return elem.nodeValue;
	}
	// Do not include comment or processing instruction nodes
```

```
	return ret;
};

Expr = Sizzle.selectors = {

	// Can be adjusted by the user
	cacheLength: 50,

	createPseudo: markFunction,

	match: matchExpr,

	attrHandle: {},

	find: {},

	relative: {
		">": { dir: "parentNode", first: true },
		" ": { dir: "parentNode" },
		"+": { dir: "previousSibling", first: true },
		"~": { dir: "previousSibling" }
	},

	preFilter: {
		"ATTR": function( match ) {
			match[1] = match[1].replace( runescape, funescape );

			// Move the given value to match[3] whether quoted or
unquoted
			match[3] = ( match[4] || match[5] || "" ).replace(
runescape, funescape );

			if ( match[2] === "~=" ) {
				match[3] = " " + match[3] + " ";
			}

			return match.slice( 0, 4 );
		},
```

```
		"CHILD": function( match ) {
			/* matches from matchExpr["CHILD"]
				1 type (only|nth|...)
				2 what (child|of-type)
				3 argument (even|odd|\d*|\d*n([+-]\d+)?|...)
				4 xn-component of xn+y argument ([+-]?\d*n|)
				5 sign of xn-component
				6 x of xn-component
				7 sign of y-component
				8 y of y-component
			*/
			match[1] = match[1].toLowerCase();

			if ( match[1].slice( 0, 3 ) === "nth" ) {
				// nth-* requires argument
				if ( !match[3] ) {
					Sizzle.error( match[0] );
				}

				// numeric x and y parameters for Expr.filter.CHILD
				// remember that false/true cast respectively to 0/1
				match[4] = +( match[4] ? match[5] + (match[6] ||
1) : 2 * ( match[3] === "even" || match[3] === "odd" ) );
				match[5] = +( ( match[7] + match[8] ) || match[3]
=== "odd" );

			// other types prohibit arguments
			} else if ( match[3] ) {
				Sizzle.error( match[0] );
			}

			return match;
		},

		"PSEUDO": function( match ) {
			var excess,
				unquoted = !match[5] && match[2];
```

```
                    if ( matchExpr["CHILD"].test( match[0] ) ) {
                            return null;
                    }

                    // Accept quoted arguments as-is
                    if ( match[3] && match[4] !== undefined ) {
                            match[2] = match[4];

                    // Strip excess characters from unquoted arguments
                    } else if ( unquoted && rpseudo.test( unquoted ) &&
                            // Get excess from tokenize (recursively)
                            (excess = tokenize( unquoted, true )) &&
                            // advance to the next closing parenthesis
                            (excess = unquoted.indexOf( ")", unquoted.length -
excess ) - unquoted.length) ) {

                            // excess is a negative index
                            match[0] = match[0].slice( 0, excess );
                            match[2] = unquoted.slice( 0, excess );
                    }

                    // Return only captures needed by the pseudo filter
method (type and argument)
                    return match.slice( 0, 3 );
            }
    },

    filter: {

            "TAG": function( nodeNameSelector ) {
                    var nodeName = nodeNameSelector.replace( runescape,
funescape ).toLowerCase();
                    return nodeNameSelector === "*" ?
                            function() { return true; } :
                            function( elem ) {
                                    return elem.nodeName &&
elem.nodeName.toLowerCase() === nodeName;
```

```
                        };
            },

            "CLASS": function( className ) {
                    var pattern = classCache[ className + " " ];

                    return pattern ||
                            (pattern = new RegExp( "(^|" + whitespace + ")" +
className + "(" + whitespace + "|$)" )) &&
                            classCache( className, function( elem ) {
                                    return pattern.test( typeof elem.className
=== "string" && elem.className || typeof elem.getAttribute !==
strundefined && elem.getAttribute("class") || "" );
                            });
            },

            "ATTR": function( name, operator, check ) {
                    return function( elem ) {
                            var result = Sizzle.attr( elem, name );

                            if ( result == null ) {
                                    return operator === "!=";
                            }
                            if ( !operator ) {
                                    return true;
                            }

                            result += "";

                            return operator === "=" ? result === check :
                                    operator === "!=" ? result !== check :
                                    operator === "^=" ? check && result.indexOf(
check ) === 0 :
                                    operator === "*=" ? check && result.indexOf(
check ) > -1 :
                                    operator === "$=" ? check && result.slice( -
check.length ) === check :
                                    operator === "~=" ? ( " " + result + " "
```

```
).indexOf( check ) > -1 :
                                operator === "|=" ? result === check ||
result.slice( 0, check.length + 1 ) === check + "-" :
                                false;
                        };
                },

                "CHILD": function( type, what, argument, first, last ) {
                        var simple = type.slice( 0, 3 ) !== "nth",
                                forward = type.slice( -4 ) !== "last",
                                ofType = what === "of-type";

                        return first === 1 && last === 0 ?

                                // Shortcut for :nth-*(n)
                                function( elem ) {
                                        return !!elem.parentNode;
                                } :

                                function( elem, context, xml ) {
                                        var cache, outerCache, node, diff, nodeIndex,
start,
                                                dir = simple !== forward ? "nextSibling"
: "previousSibling",
                                                parent = elem.parentNode,
                                                name = ofType &&
elem.nodeName.toLowerCase(),
                                                useCache = !xml && !ofType;

                                        if ( parent ) {

                                                // :(first|last|only)-(child|of-type)
                                                if ( simple ) {
                                                        while ( dir ) {
                                                                node = elem;
                                                                while ( (node = node[ dir ]) )
{
                                                                        if ( ofType ?
```

```
node.nodeName.toLowerCase() === name : node.nodeType === 1 ) {
                                                                return false;
                                                            }
                                                        }
                                                        // Reverse direction for
:only-* (if we haven't yet done so)
                                                        start = dir = type ===
"only" && !start && "nextSibling";
                                                    }
                                                    return true;
                                                }

                                                start = [ forward ? parent.firstChild :
parent.lastChild ];

                                                // non-xml :nth-child(...) stores cache
data on `parent`
                                                if ( forward && useCache ) {
                                                    // Seek `elem` from a previously-
cached index
                                                    outerCache = parent[ expando ] ||
(parent[ expando ] = {});
                                                    cache = outerCache[ type ] || [];
                                                    nodeIndex = cache[0] === dirruns
&& cache[1];
                                                    diff = cache[0] === dirruns &&
cache[2];
                                                    node = nodeIndex &&
parent.childNodes[ nodeIndex ];

                                                    while ( (node = ++nodeIndex &&
node && node[ dir ] ||

                                                        // Fallback to seeking `elem`
from the start
                                                        (diff = nodeIndex = 0) ||
start.pop()) ) {
```

```
                                                        // When found, cache
indexes on `parent` and break
                                                        if ( node.nodeType === 1
&& ++diff && node === elem ) {
                                                                outerCache[ type ] = [
dirruns, nodeIndex, diff ];
                                                                break;
                                                        }
                                                }

                                        // Use previously-cached element index
if available
                                        } else if ( useCache && (cache = (elem[
expando ] || (elem[ expando ] = {}))[ type ]) && cache[0] === dirruns ) {
                                                diff = cache[1];

                                        // xml :nth-child(...) or :nth-last-
child(...) or :nth(-last)?-of-type(...)
                                        } else {
                                                // Use the same loop as above to
seek `elem` from the start
                                                while ( (node = ++nodeIndex &&
node && node[ dir ] ||
                                                        (diff = nodeIndex = 0) ||
start.pop()) ) {

                                                        if ( ( ofType ?
node.nodeName.toLowerCase() === name : node.nodeType === 1 ) &&
++diff ) {
                                                                // Cache the index of
each encountered element
                                                                if ( useCache ) {
                                                                        (node[ expando ]
|| (node[ expando ] = {}))[ type ] = [ dirruns, diff ];
                                                                }

                                                                if ( node === elem ) {
                                                                        break;
```

```
                                                        }
                                                    }
                                                }
                                            }

                                            // Incorporate the offset, then check
against cycle size
                                            diff -= last;
                                            return diff === first || ( diff % first ===
0 && diff / first >= 0 );
                                        }
                                    };
                        },

                        "PSEUDO": function( pseudo, argument ) {
                                // pseudo-class names are case-insensitive
                                // http://www.w3.org/TR/selectors/#pseudo-classes
                                // Prioritize by case sensitivity in case custom pseudos are
added with uppercase letters
                                // Remember that setFilters inherits from pseudos
                                var args,
                                        fn = Expr.pseudos[ pseudo ] || Expr.setFilters[
pseudo.toLowerCase() ] ||
                                                Sizzle.error( "unsupported pseudo: " + pseudo
);

                                // The user may use createPseudo to indicate that
                                // arguments are needed to create the filter function
                                // just as Sizzle does
                                if ( fn[ expando ] ) {
                                        return fn( argument );
                                }

                                // But maintain support for old signatures
                                if ( fn.length > 1 ) {
                                        args = [ pseudo, pseudo, "", argument ];
                                        return Expr.setFilters.hasOwnProperty(
pseudo.toLowerCase() ) ?
```

```
                                markFunction(function( seed, matches ) {
                                        var idx,
                                                matched = fn( seed, argument ),
                                                i = matched.length;
                                        while ( i-- ) {
                                                idx = indexOf.call( seed,
matched[i] );
                                                seed[ idx ] = !( matches[ idx ] =
matched[i] );
                                        }
                                }) :
                                function( elem ) {
                                        return fn( elem, 0, args );
                                };
                        }

                        return fn;
                }
        },

        pseudos: {
                // Potentially complex pseudos
                "not": markFunction(function( selector ) {
                        // Trim the selector passed to compile
                        // to avoid treating leading and trailing
                        // spaces as combinators
                        var input = [],
                                results = [],
                                matcher = compile( selector.replace( rtrim, "$1" ) );

                        return matcher[ expando ] ?
                                markFunction(function( seed, matches, context, xml
) {
                                        var elem,
                                                unmatched = matcher( seed, null, xml,
[] ),
                                                i = seed.length;
```

```
                                // Match elements unmatched by `matcher`
                                while ( i-- ) {
                                        if ( (elem = unmatched[i]) ) {
                                                seed[i] = !(matches[i] = elem);
                                        }
                                }
                        }) :
                        function( elem, context, xml ) {
                                input[0] = elem;
                                matcher( input, null, xml, results );
                                return !results.pop();
                        };
                }),

                "has": markFunction(function( selector ) {
                        return function( elem ) {
                                return Sizzle( selector, elem ).length > 0;
                        };
                }),

                "contains": markFunction(function( text ) {
                        return function( elem ) {
                                return ( elem.textContent || elem.innerText ||
getText( elem ) ).indexOf( text ) > -1;
                        };
                }),

                // "Whether an element is represented by a :lang() selector
                // is based solely on the element's language value
                // being equal to the identifier C,
                // or beginning with the identifier C immediately followed by "-".
                // The matching of C against the element's language value is
performed case-insensitively.
                // The identifier C does not have to be a valid language name."
                // http://www.w3.org/TR/selectors/#lang-pseudo
                "lang": markFunction( function( lang ) {
                        // lang value must be a valid identifier
                        if ( !ridentifier.test(lang || "") ) {
```

```
                        Sizzle.error( "unsupported lang: " + lang );
                }
                lang = lang.replace( runescape, funescape
).toLowerCase();
                return function( elem ) {
                        var elemLang;
                        do {
                                if ( (elemLang = documentIsHTML ?
                                        elem.lang :
                                        elem.getAttribute("xml:lang") ||
elem.getAttribute("lang")) ) {

                                        elemLang = elemLang.toLowerCase();
                                        return elemLang === lang ||
elemLang.indexOf( lang + "-" ) === 0;
                                }
                        } while ( (elem = elem.parentNode) &&
elem.nodeType === 1 );
                        return false;
                };
        }),

        // Miscellaneous
        "target": function( elem ) {
                var hash = window.location && window.location.hash;
                return hash && hash.slice( 1 ) === elem.id;
        },

        "root": function( elem ) {
                return elem === docElem;
        },

        "focus": function( elem ) {
                return elem === document.activeElement &&
(!document.hasFocus || document.hasFocus()) && !!(elem.type || elem.href
|| ~elem.tabIndex);
        },
```

```
		// Boolean properties
		"enabled": function( elem ) {
			return elem.disabled === false;
		},

		"disabled": function( elem ) {
			return elem.disabled === true;
		},

		"checked": function( elem ) {
			// In CSS3, :checked should return both checked and
selected elements
			// http://www.w3.org/TR/2011/REC-css3-selectors-
20110929/#checked
			var nodeName = elem.nodeName.toLowerCase();
			return (nodeName === "input" && !!elem.checked) ||
(nodeName === "option" && !!elem.selected);
		},

		"selected": function( elem ) {
			// Accessing this property makes selected-by-default
			// options in Safari work properly
			if ( elem.parentNode ) {
				elem.parentNode.selectedIndex;
			}

			return elem.selected === true;
		},

		// Contents
		"empty": function( elem ) {
			// http://www.w3.org/TR/selectors/#empty-pseudo
			// :empty is only affected by element nodes and content
nodes(including text(3), cdata(4)),
			//   not comment, processing instructions, or others
			// Thanks to Diego Perini for the nodeName shortcut
			//   Greater than "@" means alpha characters (specifically
not starting with "#" or "?")
```

```
                for ( elem = elem.firstChild; elem; elem =
elem.nextSibling ) {
                        if ( elem.nodeName > "@" || elem.nodeType === 3
|| elem.nodeType === 4 ) {
                                return false;
                        }
                }
                return true;
        },

        "parent": function( elem ) {
                return !Expr.pseudos["empty"]( elem );
        },

        // Element/input types
        "header": function( elem ) {
                return rheader.test( elem.nodeName );
        },

        "input": function( elem ) {
                return rinputs.test( elem.nodeName );
        },

        "button": function( elem ) {
                var name = elem.nodeName.toLowerCase();
                return name === "input" && elem.type === "button" ||
name === "button";
        },

        "text": function( elem ) {
                var attr;
                // IE6 and 7 will map elem.type to 'text' for new HTML5
types (search, etc)
                // use getAttribute instead to test this case
                return elem.nodeName.toLowerCase() === "input" &&
                        elem.type === "text" &&
                        ( (attr = elem.getAttribute("type")) == null ||
attr.toLowerCase() === elem.type );
```

```
		},

		// Position-in-collection
		"first": createPositionalPseudo(function() {
			return [ 0 ];
		}),

		"last": createPositionalPseudo(function( matchIndexes, length )
{
			return [ length - 1 ];
		}),

		"eq": createPositionalPseudo(function( matchIndexes, length,
argument ) {
			return [ argument < 0 ? argument + length : argument ];
		}),

		"even": createPositionalPseudo(function( matchIndexes, length )
{
			var i = 0;
			for ( ; i < length; i += 2 ) {
				matchIndexes.push( i );
			}
			return matchIndexes;
		}),

		"odd": createPositionalPseudo(function( matchIndexes, length )
{
			var i = 1;
			for ( ; i < length; i += 2 ) {
				matchIndexes.push( i );
			}
			return matchIndexes;
		}),

		"lt": createPositionalPseudo(function( matchIndexes, length,
argument ) {
			var i = argument < 0 ? argument + length : argument;
```

```
                    for ( ; --i >= 0; ) {
                            matchIndexes.push( i );
                    }
                    return matchIndexes;
            }),

            "gt": createPositionalPseudo(function( matchIndexes, length,
argument ) {
                    var i = argument < 0 ? argument + length : argument;
                    for ( ; ++i < length; ) {
                            matchIndexes.push( i );
                    }
                    return matchIndexes;
            })
    }
};

Expr.pseudos["nth"] = Expr.pseudos["eq"];

// Add button/input type pseudos
for ( i in { radio: true, checkbox: true, file: true, password: true, image:
true } ) {
    Expr.pseudos[ i ] = createInputPseudo( i );
}
for ( i in { submit: true, reset: true } ) {
    Expr.pseudos[ i ] = createButtonPseudo( i );
}

// Easy API for creating new setFilters
function setFilters() {}
setFilters.prototype = Expr.filters = Expr.pseudos;
Expr.setFilters = new setFilters();

function tokenize( selector, parseOnly ) {
    var matched, match, tokens, type,
            soFar, groups, preFilters,
            cached = tokenCache[ selector + " " ];
```

```
        if ( cached ) {
                return parseOnly ? 0 : cached.slice( 0 );
        }

        soFar = selector;
        groups = [];
        preFilters = Expr.preFilter;

        while ( soFar ) {

                // Comma and first run
                if ( !matched || (match = rcomma.exec( soFar )) ) {
                        if ( match ) {
                                // Don't consume trailing commas as valid
                                soFar = soFar.slice( match[0].length ) || soFar;
                        }
                        groups.push( tokens = [] );
                }

                matched = false;

                // Combinators
                if ( (match = rcombinators.exec( soFar )) ) {
                        matched = match.shift();
                        tokens.push({
                                value: matched,
                                // Cast descendant combinators to space
                                type: match[0].replace( rtrim, " " )
                        });
                        soFar = soFar.slice( matched.length );
                }

                // Filters
                for ( type in Expr.filter ) {
                        if ( (match = matchExpr[ type ].exec( soFar )) &&
(!preFilters[ type ] ||
                                (match = preFilters[ type ]( match ))) ) {
                                matched = match.shift();
```

```
                        tokens.push({
                                value: matched,
                                type: type,
                                matches: match
                        });
                        soFar = soFar.slice( matched.length );
                }
        }

        if ( !matched ) {
                break;
        }
    }

    // Return the length of the invalid excess
    // if we're just parsing
    // Otherwise, throw an error or return tokens
    return parseOnly ?
        soFar.length :
        soFar ?
            Sizzle.error( selector ) :
            // Cache the tokens
            tokenCache( selector, groups ).slice( 0 );
}

function toSelector( tokens ) {
    var i = 0,
        len = tokens.length,
        selector = "";
    for ( ; i < len; i++ ) {
        selector += tokens[i].value;
    }
    return selector;
}

function addCombinator( matcher, combinator, base ) {
    var dir = combinator.dir,
        checkNonElements = base && dir === "parentNode",
```

```
		doneName = done++;

	return combinator.first ?
		// Check against closest ancestor/preceding element
		function( elem, context, xml ) {
			while ( (elem = elem[ dir ]) ) {
				if ( elem.nodeType === 1 || checkNonElements ) {
					return matcher( elem, context, xml );
				}
			}
		} :

		// Check against all ancestor/preceding elements
		function( elem, context, xml ) {
			var data, cache, outerCache,
				dirkey = dirruns + " " + doneName;

			// We can't set arbitrary data on XML nodes, so they don't
benefit from dir caching
			if ( xml ) {
				while ( (elem = elem[ dir ]) ) {
					if ( elem.nodeType === 1 ||
checkNonElements ) {
						if ( matcher( elem, context, xml ) ) {
							return true;
						}
					}
				}
			} else {
				while ( (elem = elem[ dir ]) ) {
					if ( elem.nodeType === 1 ||
checkNonElements ) {
						outerCache = elem[ expando ] || (elem[
expando ] = {});
						if ( (cache = outerCache[ dir ]) &&
cache[0] === dirkey ) {
							if ( (data = cache[1]) === true ||
data === cachedruns ) {
```

```
                                                return data === true;
                                        }
                                } else {
                                        cache = outerCache[ dir ] = [
dirkey ];
                                        cache[1] = matcher( elem,
context, xml ) || cachedruns;
                                        if ( cache[1] === true ) {
                                                return true;
                                        }
                                }
                        }
                }
        }
        };
}

function elementMatcher( matchers ) {
        return matchers.length > 1 ?
                function( elem, context, xml ) {
                        var i = matchers.length;
                        while ( i-- ) {
                                if ( !matchers[i]( elem, context, xml ) ) {
                                        return false;
                                }
                        }
                        return true;
                } :
                matchers[0];
}

function condense( unmatched, map, filter, context, xml ) {
        var elem,
                newUnmatched = [],
                i = 0,
                len = unmatched.length,
                mapped = map != null;
```

```
	for ( ; i < len; i++ ) {
		if ( (elem = unmatched[i]) ) {
			if ( !filter || filter( elem, context, xml ) ) {
				newUnmatched.push( elem );
				if ( mapped ) {
					map.push( i );
				}
			}
		}
	}

	return newUnmatched;
}

function setMatcher( preFilter, selector, matcher, postFilter, postFinder,
postSelector ) {
	if ( postFilter && !postFilter[ expando ] ) {
		postFilter = setMatcher( postFilter );
	}
	if ( postFinder && !postFinder[ expando ] ) {
		postFinder = setMatcher( postFinder, postSelector );
	}
	return markFunction(function( seed, results, context, xml ) {
		var temp, i, elem,
			preMap = [],
			postMap = [],
			preexisting = results.length,

			// Get initial elements from seed or context
			elems = seed || multipleContexts( selector || "*",
context.nodeType ? [ context ] : context, [] ),

			// Prefilter to get matcher input, preserving a map for
seed-results synchronization
			matcherIn = preFilter && ( seed || !selector ) ?
				condense( elems, preMap, preFilter, context, xml ) :
				elems,
```

```
                    matcherOut = matcher ?
                          // If we have a postFinder, or filtered seed, or non-
seed postFilter or preexisting results,
                          postFinder || ( seed ? preFilter : preexisting ||
postFilter ) ?

                                // ...intermediate processing is necessary
                                [] :

                                // ...otherwise use results directly
                                results :
                          matcherIn;

              // Find primary matches
              if ( matcher ) {
                    matcher( matcherIn, matcherOut, context, xml );
              }

              // Apply postFilter
              if ( postFilter ) {
                    temp = condense( matcherOut, postMap );
                    postFilter( temp, [], context, xml );

                    // Un-match failing elements by moving them back to
matcherIn
                    i = temp.length;
                    while ( i-- ) {
                          if ( (elem = temp[i]) ) {
                                matcherOut[ postMap[i] ] = !(matcherIn[
postMap[i] ] = elem);
                          }
                    }
              }

              if ( seed ) {
                    if ( postFinder || preFilter ) {
                          if ( postFinder ) {
                                // Get the final matcherOut by condensing this
```

```
intermediate into postFinder contexts
                                temp = [];
                                i = matcherOut.length;
                                while ( i-- ) {
                                        if ( (elem = matcherOut[i]) ) {
                                                // Restore matcherIn since elem is
not yet a final match
                                                temp.push( (matcherIn[i] = elem)
);
                                        }
                                }
                                postFinder( null, (matcherOut = []), temp, xml
);
                        }

                        // Move matched elements from seed to results to
keep them synchronized
                        i = matcherOut.length;
                        while ( i-- ) {
                                if ( (elem = matcherOut[i]) &&
                                        (temp = postFinder ? indexOf.call( seed,
elem ) : preMap[i]) > -1 ) {

                                        seed[temp] = !(results[temp] = elem);
                                }
                        }
                }

            // Add elements to results, through postFinder if defined
            } else {
                matcherOut = condense(
                        matcherOut === results ?
                                matcherOut.splice( preexisting,
matcherOut.length ) :
                                matcherOut
                );
                if ( postFinder ) {
                        postFinder( null, results, matcherOut, xml );
```

```
                        } else {
                                push.apply( results, matcherOut );
                        }
                }
        });
}

function matcherFromTokens( tokens ) {
        var checkContext, matcher, j,
                len = tokens.length,
                leadingRelative = Expr.relative[ tokens[0].type ],
                implicitRelative = leadingRelative || Expr.relative[" "],
                i = leadingRelative ? 1 : 0,

                // The foundational matcher ensures that elements are reachable
from top-level context(s)
                matchContext = addCombinator( function( elem ) {
                        return elem === checkContext;
                }, implicitRelative, true ),
                matchAnyContext = addCombinator( function( elem ) {
                        return indexOf.call( checkContext, elem ) > -1;
                }, implicitRelative, true ),
                matchers = [ function( elem, context, xml ) {
                        return ( !leadingRelative && ( xml || context !==
outermostContext ) ) || (
                                (checkContext = context).nodeType ?
                                        matchContext( elem, context, xml ) :
                                        matchAnyContext( elem, context, xml ) );
                } ];

        for ( ; i < len; i++ ) {
                if ( (matcher = Expr.relative[ tokens[i].type ]) ) {
                        matchers = [ addCombinator(elementMatcher( matchers ),
matcher) ];
                } else {
                        matcher = Expr.filter[ tokens[i].type ].apply( null,
tokens[i].matches );
```

```
                        // Return special upon seeing a positional matcher
                        if ( matcher[ expando ] ) {
                                // Find the next relative operator (if any) for proper
handling
                                j = ++i;
                                for ( ; j < len; j++ ) {
                                        if ( Expr.relative[ tokens[j].type ] ) {
                                                break;
                                        }
                                }
                                return setMatcher(
                                        i > 1 && elementMatcher( matchers ),
                                        i > 1 && toSelector(
                                                // If the preceding token was a
descendant combinator, insert an implicit any-element `*`
                                                tokens.slice( 0, i - 1 ).concat({ value:
tokens[ i - 2 ].type === " " ? "*" : "" })
                                        ).replace( rtrim, "$1" ),
                                        matcher,
                                        i < j && matcherFromTokens( tokens.slice( i, j
) ),
                                        j < len && matcherFromTokens( (tokens =
tokens.slice( j )) ),
                                        j < len && toSelector( tokens )
                                );
                        }
                        matchers.push( matcher );
                }
        }

        return elementMatcher( matchers );
}

function matcherFromGroupMatchers( elementMatchers, setMatchers ) {
        // A counter to specify which element is currently being matched
        var matcherCachedRuns = 0,
                bySet = setMatchers.length > 0,
                byElement = elementMatchers.length > 0,
```

```
            superMatcher = function( seed, context, xml, results,
expandContext ) {
                var elem, j, matcher,
                        setMatched = [],
                        matchedCount = 0,
                        i = "0",
                        unmatched = seed && [],
                        outermost = expandContext != null,
                        contextBackup = outermostContext,
                        // We must always have either seed elements or
context
                        elems = seed || byElement && Expr.find["TAG"](
"*", expandContext && context.parentNode || context ),
                        // Use integer dirruns iff this is the outermost
matcher
                        dirrunsUnique = (dirruns += contextBackup == null
? 1 : Math.random() || 0.1);

                if ( outermost ) {
                        outermostContext = context !== document &&
context;
                        cachedruns = matcherCachedRuns;
                }

                // Add elements passing elementMatchers directly to
results
                // Keep `i` a string if there are no elements so
`matchedCount` will be "00" below
                for ( ; (elem = elems[i]) != null; i++ ) {
                        if ( byElement && elem ) {
                                j = 0;
                                while ( (matcher = elementMatchers[j++]) ) {
                                        if ( matcher( elem, context, xml ) ) {
                                                results.push( elem );
                                                break;
                                        }
                                }
                                if ( outermost ) {
```

```
                                        dirruns = dirrunsUnique;
                                        cachedruns = ++matcherCachedRuns;
                                }
                        }

                        // Track unmatched elements for set filters
                        if ( bySet ) {
                                // They will have gone through all possible
matchers
                                if ( (elem = !matcher && elem) ) {
                                        matchedCount--;
                                }

                                // Lengthen the array for every element,
matched or not
                                if ( seed ) {
                                        unmatched.push( elem );
                                }
                        }
                }

                // Apply set filters to unmatched elements
                matchedCount += i;
                if ( bySet && i !== matchedCount ) {
                        j = 0;
                        while ( (matcher = setMatchers[j++]) ) {
                                matcher( unmatched, setMatched, context, xml
);
                        }

                        if ( seed ) {
                                // Reintegrate element matches to eliminate
the need for sorting
                                if ( matchedCount > 0 ) {
                                        while ( i-- ) {
                                                if ( !(unmatched[i] ||
setMatched[i]) ) {
                                                        setMatched[i] = pop.call(
```

```
results );
                                        }
                                    }
                                }

                                // Discard index placeholder values to get only
actual matches
                                setMatched = condense( setMatched );
                            }

                            // Add matches to results
                            push.apply( results, setMatched );

                            // Seedless set matches succeeding multiple
successful matchers stipulate sorting
                            if ( outermost && !seed && setMatched.length > 0
&&
                                ( matchedCount + setMatchers.length ) > 1 ) {

                                Sizzle.uniqueSort( results );
                            }
                        }

                        // Override manipulation of globals by nested matchers
                        if ( outermost ) {
                            dirruns = dirrunsUnique;
                            outermostContext = contextBackup;
                        }

                        return unmatched;
                    };

        return bySet ?
                markFunction( superMatcher ) :
                superMatcher;
}

compile = Sizzle.compile = function( selector, group /* Internal Use Only */
```

```
) {
	var i,
		setMatchers = [],
		elementMatchers = [],
		cached = compilerCache[ selector + " " ];

	if ( !cached ) {
		// Generate a function of recursive functions that can be used to
check each element
		if ( !group ) {
			group = tokenize( selector );
		}
		i = group.length;
		while ( i-- ) {
			cached = matcherFromTokens( group[i] );
			if ( cached[ expando ] ) {
				setMatchers.push( cached );
			} else {
				elementMatchers.push( cached );
			}
		}

		// Cache the compiled function
		cached = compilerCache( selector, matcherFromGroupMatchers(
elementMatchers, setMatchers ) );
	}
	return cached;
};

function multipleContexts( selector, contexts, results ) {
	var i = 0,
		len = contexts.length;
	for ( ; i < len; i++ ) {
		Sizzle( selector, contexts[i], results );
	}
	return results;
}
```

```
function select( selector, context, results, seed ) {
	var i, tokens, token, type, find,
		match = tokenize( selector );

	if ( !seed ) {
		// Try to minimize operations if there is only one group
		if ( match.length === 1 ) {

			// Take a shortcut and set the context if the root selector
is an ID
			tokens = match[0] = match[0].slice( 0 );
			if ( tokens.length > 2 && (token = tokens[0]).type ===
"ID" &&
				support.getById && context.nodeType === 9
&& documentIsHTML &&
				Expr.relative[ tokens[1].type ] ) {

				context = ( Expr.find["ID"](
token.matches[0].replace(runescape, funescape), context ) || [] )[0];
				if ( !context ) {
					return results;
				}
				selector = selector.slice( tokens.shift().value.length
);
			}

			// Fetch a seed set for right-to-left matching
			i = matchExpr["needsContext"].test( selector ) ? 0 :
tokens.length;
			while ( i-- ) {
				token = tokens[i];

				// Abort if we hit a combinator
				if ( Expr.relative[ (type = token.type) ] ) {
					break;
				}
				if ( (find = Expr.find[ type ]) ) {
					// Search, expanding context for leading
```

```
sibling combinators
                                if ( (seed = find(
                                        token.matches[0].replace( runescape,
funescape ),
                                        rsibling.test( tokens[0].type ) &&
context.parentNode || context
                                )) ) {

                                        // If seed is empty or no tokens remain,
we can return early
                                        tokens.splice( i, 1 );
                                        selector = seed.length && toSelector(
tokens );
                                        if ( !selector ) {
                                                push.apply( results, seed );
                                                return results;
                                        }

                                        break;
                                }
                        }
                }
        }
    }

    // Compile and execute a filtering function
    // Provide `match` to avoid retokenization if we modified the selector
above
    compile( selector, match )(
        seed,
        context,
        !documentIsHTML,
        results,
        rsibling.test( selector )
    );
    return results;
}
```

```
// One-time assignments

// Sort stability
support.sortStable = expando.split("").sort( sortOrder ).join("") ===
expando;

// Support: Chrome<14
// Always assume duplicates if they aren't passed to the comparison function
support.detectDuplicates = hasDuplicate;

// Initialize against the default document
setDocument();

// Support: Webkit<537.32 - Safari 6.0.3/Chrome 25 (fixed in Chrome 27)
// Detached nodes confoundingly follow *each other*
support.sortDetached = assert(function( div1 ) {
	// Should return 1, but returns 4 (following)
	return div1.compareDocumentPosition(
document.createElement("div") ) & 1;
});

// Support: IE<8
// Prevent attribute/property "interpolation"
// http://msdn.microsoft.com/en-us/library/ms536429%28VS.85%29.aspx
if ( !assert(function( div ) {
	div.innerHTML = "<a href='#'></a>";
	return div.firstChild.getAttribute("href") === "#" ;
}) ) {
	addHandle( "type|href|height|width", function( elem, name, isXML ) {
		if ( !isXML ) {
			return elem.getAttribute( name, name.toLowerCase() ===
"type" ? 1 : 2 );
		}
	});
}

// Support: IE<9
// Use defaultValue in place of getAttribute("value")
```

```
if ( !support.attributes || !assert(function( div ) {
	div.innerHTML = "<input/>";
	div.firstChild.setAttribute( "value", "" );
	return div.firstChild.getAttribute( "value" ) === "";
}) ) {
	addHandle( "value", function( elem, name, isXML ) {
		if ( !isXML && elem.nodeName.toLowerCase() === "input" ) {
			return elem.defaultValue;
		}
	});
}

// Support: IE<9
// Use getAttributeNode to fetch booleans when getAttribute lies
if ( !assert(function( div ) {
	return div.getAttribute("disabled") == null;
}) ) {
	addHandle( booleans, function( elem, name, isXML ) {
		var val;
		if ( !isXML ) {
			return (val = elem.getAttributeNode( name )) &&
val.specified ?
				val.value :
				elem[ name ] === true ? name.toLowerCase() :
null;
		}
	});
}

jQuery.find = Sizzle;
jQuery.expr = Sizzle.selectors;
jQuery.expr[":"] = jQuery.expr.pseudos;
jQuery.unique = Sizzle.uniqueSort;
jQuery.text = Sizzle.getText;
jQuery.isXMLDoc = Sizzle.isXML;
jQuery.contains = Sizzle.contains;
```

```
})( window );
// String to Object options format cache
var optionsCache = {};

// Convert String-formatted options into Object-formatted ones and store in
cache
function createOptions( options ) {
	var object = optionsCache[ options ] = {};
	jQuery.each( options.match( core_rnotwhite ) || [], function( _, flag )
{
		object[ flag ] = true;
	});
	return object;
}

/*
 * Create a callback list using the following parameters:
 *
 *	options: an optional list of space-separated options that will change
how
 *				the callback list behaves or a more traditional option object
 *
 * By default a callback list will act like an event callback list and can be
 * "fired" multiple times.
 *
 * Possible options:
 *
 *	once:		will ensure the callback list can only be fired once
(like a Deferred)
 *
 *	memory:			will keep track of previous values and will call
any callback added
 *				after the list has been fired right away with the
latest "memorized"
 *				values (like a Deferred)
 *
 *	unique:			will ensure a callback can only be added once
(no duplicate in the list)
```

```
 *
 *     stopOnFalse:      interrupt callings when a callback returns false
 *
 */
jQuery.Callbacks = function( options ) {

        // Convert options from String-formatted to Object-formatted if
needed
        // (we check in cache first)
        options = typeof options === "string" ?
                ( optionsCache[ options ] || createOptions( options ) ) :
                jQuery.extend( {}, options );

        var // Flag to know if list is currently firing
                firing,
                // Last fire value (for non-forgettable lists)
                memory,
                // Flag to know if list was already fired
                fired,
                // End of the loop when firing
                firingLength,
                // Index of currently firing callback (modified by remove if
needed)
                firingIndex,
                // First callback to fire (used internally by add and fireWith)
                firingStart,
                // Actual callback list
                list = [],
                // Stack of fire calls for repeatable lists
                stack = !options.once && [],
                // Fire callbacks
                fire = function( data ) {
                        memory = options.memory && data;
                        fired = true;
                        firingIndex = firingStart || 0;
                        firingStart = 0;
                        firingLength = list.length;
                        firing = true;
```

```
                    for ( ; list && firingIndex < firingLength; firingIndex++ ) {
                            if ( list[ firingIndex ].apply( data[ 0 ], data[ 1 ] )
=== false && options.stopOnFalse ) {
                                    memory = false; // To prevent further calls
using add
                                    break;
                            }
                    }
                    firing = false;
                    if ( list ) {
                            if ( stack ) {
                                    if ( stack.length ) {
                                            fire( stack.shift() );
                                    }
                            } else if ( memory ) {
                                    list = [];
                            } else {
                                    self.disable();
                            }
                    }
            },
            // Actual Callbacks object
            self = {
                    // Add a callback or a collection of callbacks to the list
                    add: function() {
                            if ( list ) {
                                    // First, we save the current length
                                    var start = list.length;
                                    (function add( args ) {
                                            jQuery.each( args, function( _, arg ) {
                                                    var type = jQuery.type( arg );
                                                    if ( type === "function" ) {
                                                            if ( !options.unique ||
!self.has( arg ) ) {
                                                                    list.push( arg );
                                                            }
                                                    } else if ( arg && arg.length &&
type !== "string" ) {
```

```
                                                // Inspect recursively
                                                add( arg );
                                        }
                                }});
                        })( arguments );
                        // Do we need to add the callbacks to the
                        // current firing batch?
                        if ( firing ) {
                                firingLength = list.length;
                        // With memory, if we're not firing then
                        // we should call right away
                        } else if ( memory ) {
                                firingStart = start;
                                fire( memory );
                        }
                }
                return this;
        },
        // Remove a callback from the list
        remove: function() {
                if ( list ) {
                        jQuery.each( arguments, function( _, arg ) {
                                var index;
                                while( ( index = jQuery.inArray( arg, list,
index ) ) > -1 ) {
                                        list.splice( index, 1 );
                                        // Handle firing indexes
                                        if ( firing ) {
                                                if ( index <= firingLength ) {
                                                        firingLength--;
                                                }
                                                if ( index <= firingIndex ) {
                                                        firingIndex--;
                                                }
                                        }
                                }
                        });
                }
```

```
                        return this;
                },
                // Check if a given callback is in the list.
                // If no argument is given, return whether or not list has
callbacks attached.
                has: function( fn ) {
                        return fn ? jQuery.inArray( fn, list ) > -1 : !!( list &&
list.length );
                },
                // Remove all callbacks from the list
                empty: function() {
                        list = [];
                        firingLength = 0;
                        return this;
                },
                // Have the list do nothing anymore
                disable: function() {
                        list = stack = memory = undefined;
                        return this;
                },
                // Is it disabled?
                disabled: function() {
                        return !list;
                },
                // Lock the list in its current state
                lock: function() {
                        stack = undefined;
                        if ( !memory ) {
                                self.disable();
                        }
                        return this;
                },
                // Is it locked?
                locked: function() {
                        return !stack;
                },
                // Call all callbacks with the given context and arguments
                fireWith: function( context, args ) {
```

```
                        if ( list && ( !fired || stack ) ) {
                                args = args || [];
                                args = [ context, args.slice ? args.slice() : args
];
                                if ( firing ) {
                                        stack.push( args );
                                } else {
                                        fire( args );
                                }
                        }
                        return this;
                },
                // Call all the callbacks with the given arguments
                fire: function() {
                        self.fireWith( this, arguments );
                        return this;
                },
                // To know if the callbacks have already been called at
least once
                fired: function() {
                        return !!fired;
                }
        };

    return self;
};
jQuery.extend({

    Deferred: function( func ) {
        var tuples = [
                        // action, add listener, listener list, final state
                        [ "resolve", "done", jQuery.Callbacks("once
memory"), "resolved" ],
                        [ "reject", "fail", jQuery.Callbacks("once memory"),
"rejected" ],
                        [ "notify", "progress", jQuery.Callbacks("memory") ]
                ],
                state = "pending",
```

```
promise = {
	state: function() {
		return state;
	},
	always: function() {
		deferred.done( arguments ).fail( arguments );
		return this;
	},
	then: function( /* fnDone, fnFail, fnProgress */ ) {
		var fns = arguments;
		return jQuery.Deferred(function( newDefer ) {
			jQuery.each( tuples, function( i, tuple ) {
				var action = tuple[ 0 ],
					fn = jQuery.isFunction( fns[ i
] ) && fns[ i ];
				// deferred[ done | fail | progress ]
for forwarding actions to newDefer
				deferred[ tuple[1] ](function() {
					var returned = fn &&
fn.apply( this, arguments );
					if ( returned &&
jQuery.isFunction( returned.promise ) ) {
						returned.promise()
							.done(
newDefer.resolve )
							.fail(
newDefer.reject )
							.progress(
newDefer.notify );
					} else {
						newDefer[ action +
"With" ]( this === promise ? newDefer.promise() : this, fn ? [ returned ] :
arguments );
					}
				});
			});
			fns = null;
		}).promise();
```

```
                        },
                        // Get a promise for this deferred
                        // If obj is provided, the promise aspect is added to
the object
                        promise: function( obj ) {
                                return obj != null ? jQuery.extend( obj,
promise ) : promise;
                        }
                },
                deferred = {};

        // Keep pipe for back-compat
        promise.pipe = promise.then;

        // Add list-specific methods
        jQuery.each( tuples, function( i, tuple ) {
                var list = tuple[ 2 ],
                        stateString = tuple[ 3 ];

                // promise[ done | fail | progress ] = list.add
                promise[ tuple[1] ] = list.add;

                // Handle state
                if ( stateString ) {
                        list.add(function() {
                                // state = [ resolved | rejected ]
                                state = stateString;

                        // [ reject_list | resolve_list ].disable;
progress_list.lock
                        }, tuples[ i ^ 1 ][ 2 ].disable, tuples[ 2 ][ 2 ].lock );
                }

                // deferred[ resolve | reject | notify ]
                deferred[ tuple[0] ] = function() {
                        deferred[ tuple[0] + "With" ]( this === deferred ?
promise : this, arguments );
                        return this;
```

```
                    };
                    deferred[ tuple[0] + "With" ] = list.fireWith;
            });

            // Make the deferred a promise
            promise.promise( deferred );

            // Call given func if any
            if ( func ) {
                    func.call( deferred, deferred );
            }

            // All done!
            return deferred;
    },

    // Deferred helper
    when: function( subordinate /* , ..., subordinateN */ ) {
            var i = 0,
                    resolveValues = core_slice.call( arguments ),
                    length = resolveValues.length,

                    // the count of uncompleted subordinates
                    remaining = length !== 1 || ( subordinate &&
jQuery.isFunction( subordinate.promise ) ) ? length : 0,

                    // the master Deferred. If resolveValues consist of only a
single Deferred, just use that.
                    deferred = remaining === 1 ? subordinate :
jQuery.Deferred(),

                    // Update function for both resolve and progress values
                    updateFunc = function( i, contexts, values ) {
                            return function( value ) {
                                    contexts[ i ] = this;
                                    values[ i ] = arguments.length > 1 ?
core_slice.call( arguments ) : value;
                                    if( values === progressValues ) {
```

```
                                        deferred.notifyWith( contexts, values );
                                } else if ( !( --remaining ) ) {
                                        deferred.resolveWith( contexts, values );
                                }
                        };
                },

                progressValues, progressContexts, resolveContexts;

        // add listeners to Deferred subordinates; treat others as
resolved
        if ( length > 1 ) {
                progressValues = new Array( length );
                progressContexts = new Array( length );
                resolveContexts = new Array( length );
                for ( ; i < length; i++ ) {
                        if ( resolveValues[ i ] && jQuery.isFunction(
resolveValues[ i ].promise ) ) {
                                resolveValues[ i ].promise()
                                        .done( updateFunc( i, resolveContexts,
resolveValues ) )
                                        .fail( deferred.reject )
                                        .progress( updateFunc( i,
progressContexts, progressValues ) );
                        } else {
                                --remaining;
                        }
                }
        }

        // if we're not waiting on anything, resolve the master
        if ( !remaining ) {
                deferred.resolveWith( resolveContexts, resolveValues );
        }

        return deferred.promise();
    }
});
```

```
jQuery.support = (function( support ) {

	var all, a, input, select, fragment, opt, eventName, isSupported, i,
		div = document.createElement("div");

	// Setup
	div.setAttribute( "className", "t" );
	div.innerHTML = "  <link/><table></table><a
href='/a'>a</a><input type='checkbox'/>";

	// Finish early in limited (non-browser) environments
	all = div.getElementsByTagName("*") || [];
	a = div.getElementsByTagName("a")[ 0 ];
	if ( !a || !a.style || !all.length ) {
		return support;
	}

	// First batch of tests
	select = document.createElement("select");
	opt = select.appendChild( document.createElement("option") );
	input = div.getElementsByTagName("input")[ 0 ];

	a.style.cssText = "top:1px;float:left;opacity:.5";

	// Test setAttribute on camelCase class. If it works, we need attrFixes
when doing get/setAttribute (ie6/7)
	support.getSetAttribute = div.className !== "t";

	// IE strips leading whitespace when .innerHTML is used
	support.leadingWhitespace = div.firstChild.nodeType === 3;

	// Make sure that tbody elements aren't automatically inserted
	// IE will insert them into empty tables
	support.tbody = !div.getElementsByTagName("tbody").length;

	// Make sure that link elements get serialized correctly by innerHTML
	// This requires a wrapper element in IE
	support.htmlSerialize = !!div.getElementsByTagName("link").length;
```

```
	// Get the style information from getAttribute
	// (IE uses .cssText instead)
	support.style = /top/.test( a.getAttribute("style") );

	// Make sure that URLs aren't manipulated
	// (IE normalizes it by default)
	support.hrefNormalized = a.getAttribute("href") === "/a";

	// Make sure that element opacity exists
	// (IE uses filter instead)
	// Use a regex to work around a WebKit issue. See #5145
	support.opacity = /^0.5/.test( a.style.opacity );

	// Verify style float existence
	// (IE uses styleFloat instead of cssFloat)
	support.cssFloat = !!a.style.cssFloat;

	// Check the default checkbox/radio value ("" on WebKit; "on"
elsewhere)
	support.checkOn = !!input.value;

	// Make sure that a selected-by-default option has a working selected
property.
	// (WebKit defaults to false instead of true, IE too, if it's in an
optgroup)
	support.optSelected = opt.selected;

	// Tests for enctype support on a form (#6743)
	support.enctype = !!document.createElement("form").enctype;

	// Makes sure cloning an html5 element does not cause problems
	// Where outerHTML is undefined, this still works
	support.html5Clone = document.createElement("nav").cloneNode(
true ).outerHTML !== "<:nav></:nav>";

	// Will be defined later
	support.inlineBlockNeedsLayout = false;
```

```
	support.shrinkWrapBlocks = false;
	support.pixelPosition = false;
	support.deleteExpando = true;
	support.noCloneEvent = true;
	support.reliableMarginRight = true;
	support.boxSizingReliable = true;

	// Make sure checked status is properly cloned
	input.checked = true;
	support.noCloneChecked = input.cloneNode( true ).checked;

	// Make sure that the options inside disabled selects aren't marked as
disabled
	// (WebKit marks them as disabled)
	select.disabled = true;
	support.optDisabled = !opt.disabled;

	// Support: IE<9
	try {
		delete div.test;
	} catch( e ) {
		support.deleteExpando = false;
	}

	// Check if we can trust getAttribute("value")
	input = document.createElement("input");
	input.setAttribute( "value", "" );
	support.input = input.getAttribute( "value" ) === "";

	// Check if an input maintains its value after becoming a radio
	input.value = "t";
	input.setAttribute( "type", "radio" );
	support.radioValue = input.value === "t";

	// #11217 - WebKit loses check when the name is after the checked
attribute
	input.setAttribute( "checked", "t" );
	input.setAttribute( "name", "t" );
```

```
	fragment = document.createDocumentFragment();
	fragment.appendChild( input );

	// Check if a disconnected checkbox will retain its checked
	// value of true after appended to the DOM (IE6/7)
	support.appendChecked = input.checked;

	// WebKit doesn't clone checked state correctly in fragments
	support.checkClone = fragment.cloneNode( true ).cloneNode( true
).lastChild.checked;

	// Support: IE<9
	// Opera does not clone events (and typeof div.attachEvent ===
undefined).
	// IE9-10 clones events bound via attachEvent, but they don't trigger
with .click()
	if ( div.attachEvent ) {
		div.attachEvent( "onclick", function() {
			support.noCloneEvent = false;
		});

		div.cloneNode( true ).click();
	}

	// Support: IE<9 (lack submit/change bubble), Firefox 17+ (lack
focusin event)
	// Beware of CSP restrictions
(https://developer.mozilla.org/en/Security/CSP)
	for ( i in { submit: true, change: true, focusin: true }) {
		div.setAttribute( eventName = "on" + i, "t" );

		support[ i + "Bubbles" ] = eventName in window ||
div.attributes[ eventName ].expando === false;
	}

	div.style.backgroundClip = "content-box";
	div.cloneNode( true ).style.backgroundClip = "";
```

```
	support.clearCloneStyle = div.style.backgroundClip === "content-
box";

	// Support: IE<9
	// Iteration over object's inherited properties before its own.
	for ( i in jQuery( support ) ) {
		break;
	}
	support.ownLast = i !== "0";

	// Run tests that need a body at doc ready
	jQuery(function() {
		var container, marginDiv, tds,
			divReset =
"padding:0;margin:0;border:0;display:block;box-sizing:content-box;-moz-
box-sizing:content-box;-webkit-box-sizing:content-box;",
			body = document.getElementsByTagName("body")[0];

		if ( !body ) {
			// Return for frameset docs that don't have a body
			return;
		}

		container = document.createElement("div");
		container.style.cssText =
"border:0;width:0;height:0;position:absolute;top:0;left:-9999px;margin-
top:1px";

		body.appendChild( container ).appendChild( div );

		// Support: IE8
		// Check if table cells still have offsetWidth/Height when they are
set
		// to display:none and there are still other visible table cells in a
		// table row; if so, offsetWidth/Height are not reliable for use
when
		// determining if an element has been hidden directly using
		// display:none (it is still safe to use offsets if a parent element
```

```
is
		// hidden; don safety goggles and see bug #4512 for more
information).
		div.innerHTML =
"<table><tr><td></td><td>t</td></tr></table>";
		tds = div.getElementsByTagName("td");
		tds[ 0 ].style.cssText =
"padding:0;margin:0;border:0;display:none";
		isSupported = ( tds[ 0 ].offsetHeight === 0 );

		tds[ 0 ].style.display = "";
		tds[ 1 ].style.display = "none";

		// Support: IE8
		// Check if empty table cells still have offsetWidth/Height
		support.reliableHiddenOffsets = isSupported && ( tds[ 0
].offsetHeight === 0 );

		// Check box-sizing and margin behavior.
		div.innerHTML = "";
		div.style.cssText = "box-sizing:border-box;-moz-box-
sizing:border-box;-webkit-box-sizing:border-
box;padding:1px;border:1px;display:block;width:4px;margin-
top:1%;position:absolute;top:1%;";

		// Workaround failing boxSizing test due to offsetWidth returning
wrong value
		// with some non-1 values of body zoom, ticket #13543
		jQuery.swap( body, body.style.zoom != null ? { zoom: 1 } : {},
function() {
			support.boxSizing = div.offsetWidth === 4;
		});

		// Use window.getComputedStyle because jsdom on node.js will
break without it.
		if ( window.getComputedStyle ) {
			support.pixelPosition = ( window.getComputedStyle( div,
null ) || {} ).top !== "1%";
```

```
                    support.boxSizingReliable = ( window.getComputedStyle(
div, null ) || { width: "4px" } ).width === "4px";

                    // Check if div with explicit width and no margin-right
incorrectly
                    // gets computed margin-right based on width of
container. (#3333)
                    // Fails in WebKit before Feb 2011 nightlies
                    // WebKit Bug 13343 - getComputedStyle returns wrong
value for margin-right
                    marginDiv = div.appendChild(
document.createElement("div") );
                    marginDiv.style.cssText = div.style.cssText = divReset;
                    marginDiv.style.marginRight = marginDiv.style.width =
"0";
                    div.style.width = "1px";

                    support.reliableMarginRight =
                          !parseFloat( ( window.getComputedStyle( marginDiv,
null ) || {} ).marginRight );
              }

              if ( typeof div.style.zoom !== core_strundefined ) {
                    // Support: IE<8
                    // Check if natively block-level elements act like inline-
block
                    // elements when setting their display to 'inline' and giving
                    // them layout
                    div.innerHTML = "";
                    div.style.cssText = divReset +
"width:1px;padding:1px;display:inline;zoom:1";
                    support.inlineBlockNeedsLayout = ( div.offsetWidth === 3
);

                    // Support: IE6
                    // Check if elements with layout shrink-wrap their children
                    div.style.display = "block";
                    div.innerHTML = "<div></div>";
```

```
                div.firstChild.style.width = "5px";
                support.shrinkWrapBlocks = ( div.offsetWidth !== 3 );

                if ( support.inlineBlockNeedsLayout ) {
                        // Prevent IE 6 from affecting layout for positioned
elements #11048
                        // Prevent IE from shrinking the body in IE 7 mode
#12869
                        // Support: IE<8
                        body.style.zoom = 1;
                }
        }

        body.removeChild( container );

        // Null elements to avoid leaks in IE
        container = div = tds = marginDiv = null;
    });

    // Null elements to avoid leaks in IE
    all = select = fragment = opt = a = input = null;

    return support;
})({});

var rbrace = /(?:\{[\s\S]*\}|\[[\s\S]*\])$/,
    rmultiDash = /([A-Z])/g;

function internalData( elem, name, data, pvt /* Internal Use Only */ ){
    if ( !jQuery.acceptData( elem ) ) {
        return;
    }

    var ret, thisCache,
        internalKey = jQuery.expando,

        // We have to handle DOM nodes and JS objects differently
because IE6-7
```

```
		// can't GC object references properly across the DOM-JS
boundary
		isNode = elem.nodeType,

		// Only DOM nodes need the global jQuery cache; JS object data
is
		// attached directly to the object so GC can occur automatically
		cache = isNode ? jQuery.cache : elem,

		// Only defining an ID for JS objects if its cache already exists
allows
		// the code to shortcut on the same path as a DOM node with no
cache
		id = isNode ? elem[ internalKey ] : elem[ internalKey ] &&
internalKey;

	// Avoid doing any more work than we need to when trying to get data
on an
	// object that has no data at all
	if ( (!id || !cache[id] || (!pvt && !cache[id].data)) && data ===
undefined && typeof name === "string" ) {
		return;
	}

	if ( !id ) {
		// Only DOM nodes need a new unique ID for each element since
their data
		// ends up in the global cache
		if ( isNode ) {
			id = elem[ internalKey ] = core_deletedIds.pop() ||
jQuery.guid++;
		} else {
			id = internalKey;
		}
	}

	if ( !cache[ id ] ) {
		// Avoid exposing jQuery metadata on plain JS objects when the
```

```
object
		// is serialized using JSON.stringify
		cache[ id ] = isNode ? {} : { toJSON: jQuery.noop };
	}

	// An object can be passed to jQuery.data instead of a key/value pair;
this gets
	// shallow copied over onto the existing cache
	if ( typeof name === "object" || typeof name === "function" ) {
		if ( pvt ) {
			cache[ id ] = jQuery.extend( cache[ id ], name );
		} else {
			cache[ id ].data = jQuery.extend( cache[ id ].data, name
);
		}
	}

	thisCache = cache[ id ];

	// jQuery data() is stored in a separate object inside the object's
internal data
	// cache in order to avoid key collisions between internal data and
user-defined
	// data.
	if ( !pvt ) {
		if ( !thisCache.data ) {
			thisCache.data = {};
		}

		thisCache = thisCache.data;
	}

	if ( data !== undefined ) {
		thisCache[ jQuery.camelCase( name ) ] = data;
	}

	// Check for both converted-to-camel and non-converted data property
names
```

```
	// If a data property was specified
	if ( typeof name === "string" ) {

		// First Try to find as-is property data
		ret = thisCache[ name ];

		// Test for null|undefined property data
		if ( ret == null ) {

			// Try to find the camelCased property
			ret = thisCache[ jQuery.camelCase( name ) ];
		}
	} else {
		ret = thisCache;
	}

	return ret;
}

function internalRemoveData( elem, name, pvt ) {
	if ( !jQuery.acceptData( elem ) ) {
		return;
	}

	var thisCache, i,
		isNode = elem.nodeType,

		// See jQuery.data for more information
		cache = isNode ? jQuery.cache : elem,
		id = isNode ? elem[ jQuery.expando ] : jQuery.expando;

	// If there is already no cache entry for this object, there is no
	// purpose in continuing
	if ( !cache[ id ] ) {
		return;
	}

	if ( name ) {
```

```
		thisCache = pvt ? cache[ id ] : cache[ id ].data;

		if ( thisCache ) {

			// Support array or space separated string names for data
keys
			if ( !jQuery.isArray( name ) ) {

				// try the string as a key before any manipulation
				if ( name in thisCache ) {
					name = [ name ];
				} else {

					// split the camel cased version by spaces
unless a key with the spaces exists
					name = jQuery.camelCase( name );
					if ( name in thisCache ) {
						name = [ name ];
					} else {
						name = name.split(" ");
					}
				}
			} else {
				// If "name" is an array of keys...
				// When data is initially created, via ("key", "val")
signature,
				// keys will be converted to camelCase.
				// Since there is no way to tell _how_ a key was
added, remove
				// both plain key and camelCase key. #12786
				// This will only penalize the array argument path.
				name = name.concat( jQuery.map( name,
jQuery.camelCase ) );
			}

			i = name.length;
			while ( i-- ) {
```

```
                        delete thisCache[ name[i] ];
                    }

                    // If there is no data left in the cache, we want to continue
                    // and let the cache object itself get destroyed
                    if ( pvt ? !isEmptyDataObject(thisCache) :
!jQuery.isEmptyObject(thisCache) ) {
                        return;
                    }
                }
            }

            // See jQuery.data for more information
            if ( !pvt ) {
                delete cache[ id ].data;

                // Don't destroy the parent cache unless the internal data object
                // had been the only thing left in it
                if ( !isEmptyDataObject( cache[ id ] ) ) {
                    return;
                }
            }

            // Destroy the cache
            if ( isNode ) {
                jQuery.cleanData( [ elem ], true );

            // Use delete when supported for expandos or `cache` is not a window
per isWindow (#10080)
            /* jshint eqeqeq: false */
            } else if ( jQuery.support.deleteExpando || cache != cache.window ) {
                /* jshint eqeqeq: true */
                delete cache[ id ];

            // When all else fails, null
            } else {
                cache[ id ] = null;
            }
```

```
}

jQuery.extend({
	cache: {},

	// The following elements throw uncatchable exceptions if you
	// attempt to add expando properties to them.
	noData: {
		"applet": true,
		"embed": true,
		// Ban all objects except for Flash (which handle expandos)
		"object": "clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
	},

	hasData: function( elem ) {
		elem = elem.nodeType ? jQuery.cache[ elem[jQuery.expando] ]
: elem[ jQuery.expando ];
		return !!elem && !isEmptyDataObject( elem );
	},

	data: function( elem, name, data ) {
		return internalData( elem, name, data );
	},

	removeData: function( elem, name ) {
		return internalRemoveData( elem, name );
	},

	// For internal use only.
	_data: function( elem, name, data ) {
		return internalData( elem, name, data, true );
	},

	_removeData: function( elem, name ) {
		return internalRemoveData( elem, name, true );
	},

	// A method for determining if a DOM node can handle the data
```

```
expando
	acceptData: function( elem ) {
		// Do not set data on non-element because it will not be cleared
(#8335).
		if ( elem.nodeType && elem.nodeType !== 1 && elem.nodeType
!== 9 ) {
			return false;
		}

		var noData = elem.nodeName && jQuery.noData[
elem.nodeName.toLowerCase() ];

		// nodes accept data unless otherwise specified; rejection can be
conditional
		return !noData || noData !== true &&
elem.getAttribute("classid") === noData;
	}
});

jQuery.fn.extend({
	data: function( key, value ) {
		var attrs, name,
			data = null,
			i = 0,
			elem = this[0];

		// Special expections of .data basically thwart jQuery.access,
		// so implement the relevant behavior ourselves

		// Gets all values
		if ( key === undefined ) {
			if ( this.length ) {
				data = jQuery.data( elem );

				if ( elem.nodeType === 1 && !jQuery._data( elem,
"parsedAttrs" ) ) {
					attrs = elem.attributes;
					for ( ; i < attrs.length; i++ ) {
```

```
                                        name = attrs[i].name;

                                        if ( name.indexOf("data-") === 0 ) {
                                                name = jQuery.camelCase(
name.slice(5) );

                                                dataAttr( elem, name, data[ name
] );
                                        }
                                }
                                jQuery._data( elem, "parsedAttrs", true );
                        }
                }

                return data;
        }

        // Sets multiple values
        if ( typeof key === "object" ) {
                return this.each(function() {
                        jQuery.data( this, key );
                });
        }

        return arguments.length > 1 ?

                // Sets one value
                this.each(function() {
                        jQuery.data( this, key, value );
                }) :

                // Gets one value
                // Try to fetch any internally stored data first
                elem ? dataAttr( elem, key, jQuery.data( elem, key ) ) :
null;
    },

    removeData: function( key ) {
```

```
		return this.each(function() {
			jQuery.removeData( this, key );
		});
	}
});

function dataAttr( elem, key, data ) {
	// If nothing was found internally, try to fetch any
	// data from the HTML5 data-* attribute
	if ( data === undefined && elem.nodeType === 1 ) {

		var name = "data-" + key.replace( rmultiDash, "-$1"
).toLowerCase();

		data = elem.getAttribute( name );

		if ( typeof data === "string" ) {
			try {
				data = data === "true" ? true :
					data === "false" ? false :
					data === "null" ? null :
					// Only convert to a number if it doesn't
change the string
					+data + "" === data ? +data :
					rbrace.test( data ) ? jQuery.parseJSON( data )
:
						data;
			} catch( e ) {}

			// Make sure we set the data so it isn't changed later
			jQuery.data( elem, key, data );

		} else {
			data = undefined;
		}
	}

	return data;
```

```
}

// checks a cache object for emptiness
function isEmptyDataObject( obj ) {
	var name;
	for ( name in obj ) {

		// if the public data object is empty, the private is still empty
		if ( name === "data" && jQuery.isEmptyObject( obj[name] ) ) {
			continue;
		}
		if ( name !== "toJSON" ) {
			return false;
		}
	}

	return true;
}
jQuery.extend({
	queue: function( elem, type, data ) {
		var queue;

		if ( elem ) {
			type = ( type || "fx" ) + "queue";
			queue = jQuery._data( elem, type );

			// Speed up dequeue by getting out quickly if this is just a
lookup
			if ( data ) {
				if ( !queue || jQuery.isArray(data) ) {
					queue = jQuery._data( elem, type,
jQuery.makeArray(data) );
				} else {
					queue.push( data );
				}
			}
			return queue || [];
		}
```

```
	},

	dequeue: function( elem, type ) {
		type = type || "fx";

		var queue = jQuery.queue( elem, type ),
			startLength = queue.length,
			fn = queue.shift(),
			hooks = jQuery._queueHooks( elem, type ),
			next = function() {
				jQuery.dequeue( elem, type );
			};

		// If the fx queue is dequeued, always remove the progress
sentinel
		if ( fn === "inprogress" ) {
			fn = queue.shift();
			startLength--;
		}

		if ( fn ) {

			// Add a progress sentinel to prevent the fx queue from
being
			// automatically dequeued
			if ( type === "fx" ) {
				queue.unshift( "inprogress" );
			}

			// clear up the last queue stop function
			delete hooks.stop;
			fn.call( elem, next, hooks );
		}

		if ( !startLength && hooks ) {
			hooks.empty.fire();
		}
	},
```

```
	// not intended for public consumption - generates a queueHooks
object, or returns the current one
	_queueHooks: function( elem, type ) {
		var key = type + "queueHooks";
		return jQuery._data( elem, key ) || jQuery._data( elem, key, {
			empty: jQuery.Callbacks("once memory").add(function() {
				jQuery._removeData( elem, type + "queue" );
				jQuery._removeData( elem, key );
			})
		});
	}
});

jQuery.fn.extend({
	queue: function( type, data ) {
		var setter = 2;

		if ( typeof type !== "string" ) {
			data = type;
			type = "fx";
			setter--;
		}

		if ( arguments.length < setter ) {
			return jQuery.queue( this[0], type );
		}

		return data === undefined ?
			this :
			this.each(function() {
				var queue = jQuery.queue( this, type, data );

				// ensure a hooks for this queue
				jQuery._queueHooks( this, type );

				if ( type === "fx" && queue[0] !== "inprogress" ) {
					jQuery.dequeue( this, type );
```

```
                        }
                });
        },
        dequeue: function( type ) {
                return this.each(function() {
                        jQuery.dequeue( this, type );
                });
        },
        // Based off of the plugin by Clint Helfers, with permission.
        // http://blindsignals.com/index.php/2009/07/jquery-delay/
        delay: function( time, type ) {
                time = jQuery.fx ? jQuery.fx.speeds[ time ] || time : time;
                type = type || "fx";

                return this.queue( type, function( next, hooks ) {
                        var timeout = setTimeout( next, time );
                        hooks.stop = function() {
                                clearTimeout( timeout );
                        };
                });
        },
        clearQueue: function( type ) {
                return this.queue( type || "fx", [] );
        },
        // Get a promise resolved when queues of a certain type
        // are emptied (fx is the type by default)
        promise: function( type, obj ) {
                var tmp,
                        count = 1,
                        defer = jQuery.Deferred(),
                        elements = this,
                        i = this.length,
                        resolve = function() {
                                if ( !( --count ) ) {
                                        defer.resolveWith( elements, [ elements ] );
                                }
                        };
```

```
		if ( typeof type !== "string" ) {
			obj = type;
			type = undefined;
		}
		type = type || "fx";

		while( i-- ) {
			tmp = jQuery._data( elements[ i ], type + "queueHooks"
);
			if ( tmp && tmp.empty ) {
				count++;
				tmp.empty.add( resolve );
			}
		}
		resolve();
		return defer.promise( obj );
	}
});
var nodeHook, boolHook,
	rclass = /[\t\r\n\f]/g,
	rreturn = /\r/g,
	rfocusable = /^(?:input|select|textarea|button|object)$/i,
	rclickable = /^(?:a|area)$/i,
	ruseDefault = /^(?:checked|selected)$/i,
	getSetAttribute = jQuery.support.getSetAttribute,
	getSetInput = jQuery.support.input;

jQuery.fn.extend({
	attr: function( name, value ) {
		return jQuery.access( this, jQuery.attr, name, value,
arguments.length > 1 );
	},

	removeAttr: function( name ) {
		return this.each(function() {
			jQuery.removeAttr( this, name );
		});
	},
```

```
	prop: function( name, value ) {
		return jQuery.access( this, jQuery.prop, name, value,
arguments.length > 1 );
	},

	removeProp: function( name ) {
		name = jQuery.propFix[ name ] || name;
		return this.each(function() {
			// try/catch handles cases where IE balks (such as
removing a property on window)
			try {
				this[ name ] = undefined;
				delete this[ name ];
			} catch( e ) {}
		});
	},

	addClass: function( value ) {
		var classes, elem, cur, clazz, j,
			i = 0,
			len = this.length,
			proceed = typeof value === "string" && value;

		if ( jQuery.isFunction( value ) ) {
			return this.each(function( j ) {
				jQuery( this ).addClass( value.call( this, j,
this.className ) );
			});
		}

		if ( proceed ) {
			// The disjunction here is for better compressibility (see
removeClass)
			classes = ( value || "" ).match( core_rnotwhite ) || [];

			for ( ; i < len; i++ ) {
				elem = this[ i ];
```

```
                        cur = elem.nodeType === 1 && ( elem.className ?
                                ( " " + elem.className + " " ).replace( rclass,
" " ) :
                                " "
                        );

                        if ( cur ) {
                                j = 0;
                                while ( (clazz = classes[j++]) ) {
                                        if ( cur.indexOf( " " + clazz + " " ) < 0 )
{
                                                cur += clazz + " ";
                                        }
                                }
                                elem.className = jQuery.trim( cur );

                        }
                }
        }

        return this;
    },

    removeClass: function( value ) {
        var classes, elem, cur, clazz, j,
            i = 0,
            len = this.length,
            proceed = arguments.length === 0 || typeof value ===
"string" && value;

        if ( jQuery.isFunction( value ) ) {
            return this.each(function( j ) {
                jQuery( this ).removeClass( value.call( this, j,
this.className ) );
            });
        }
        if ( proceed ) {
            classes = ( value || "" ).match( core_rnotwhite ) || [];
```

```
                    for ( ; i < len; i++ ) {
                        elem = this[ i ];
                        // This expression is here for better compressibility
(see addClass)
                        cur = elem.nodeType === 1 && ( elem.className ?
                            ( " " + elem.className + " " ).replace( rclass,
" " ) :
                            ""
                        );

                        if ( cur ) {
                            j = 0;
                            while ( (clazz = classes[j++]) ) {
                                // Remove *all* instances
                                while ( cur.indexOf( " " + clazz + " " )
>= 0 ) {
                                    cur = cur.replace( " " + clazz + " ",
" " );
                                }
                            }
                            elem.className = value ? jQuery.trim( cur ) :
"";
                        }
                    }
                }

                return this;
            },

            toggleClass: function( value, stateVal ) {
                var type = typeof value;

                if ( typeof stateVal === "boolean" && type === "string" ) {
                    return stateVal ? this.addClass( value ) : this.removeClass(
value );
                }
```

```
		if ( jQuery.isFunction( value ) ) {
			return this.each(function( i ) {
				jQuery( this ).toggleClass( value.call(this, i,
this.className, stateVal), stateVal );
			});
		}

		return this.each(function() {
			if ( type === "string" ) {
				// toggle individual class names
				var className,
					i = 0,
					self = jQuery( this ),
					classNames = value.match( core_rnotwhite )
|| [];

				while ( (className = classNames[ i++ ]) ) {
					// check each className given, space
separated list
					if ( self.hasClass( className ) ) {
						self.removeClass( className );
					} else {
						self.addClass( className );
					}
				}

			// Toggle whole class name
			} else if ( type === core_strundefined || type ===
"boolean" ) {
				if ( this.className ) {
					// store className if set
					jQuery._data( this, "__className__",
this.className );
				}

				// If the element has a class name or if we're passed
"false",
				// then remove the whole classname (if there was
```

```
one, the above saved it).
                        // Otherwise bring back whatever was previously
saved (if anything),
                        // falling back to the empty string if nothing was
stored.
                        this.className = this.className || value === false
? "" : jQuery._data( this, "__className__" ) || "";
                }
            });
        },

        hasClass: function( selector ) {
            var className = " " + selector + " ",
                i = 0,
                l = this.length;
            for ( ; i < l; i++ ) {
                if ( this[i].nodeType === 1 && (" " + this[i].className + "
").replace(rclass, " ").indexOf( className ) >= 0 ) {
                        return true;
                }
            }

            return false;
        },

        val: function( value ) {
            var ret, hooks, isFunction,
                elem = this[0];

            if ( !arguments.length ) {
                if ( elem ) {
                        hooks = jQuery.valHooks[ elem.type ] ||
jQuery.valHooks[ elem.nodeName.toLowerCase() ];

                        if ( hooks && "get" in hooks && (ret = hooks.get(
elem, "value" )) !== undefined ) {
                                return ret;
                        }
```

```
                        ret = elem.value;

                        return typeof ret === "string" ?
                                // handle most common string cases
                                ret.replace(rreturn, "") :
                                // handle cases where value is null/undef or
number
                                ret == null ? "" : ret;
                }

                return;
        }

        isFunction = jQuery.isFunction( value );

        return this.each(function( i ) {
                var val;

                if ( this.nodeType !== 1 ) {
                        return;
                }

                if ( isFunction ) {
                        val = value.call( this, i, jQuery( this ).val() );
                } else {
                        val = value;
                }

                // Treat null/undefined as ""; convert numbers to string
                if ( val == null ) {
                        val = "";
                } else if ( typeof val === "number" ) {
                        val += "";
                } else if ( jQuery.isArray( val ) ) {
                        val = jQuery.map(val, function ( value ) {
                                return value == null ? "" : value + "";
                        });
```

```
                }

                hooks = jQuery.valHooks[ this.type ] || jQuery.valHooks[
this.nodeName.toLowerCase() ];

                // If set returns undefined, fall back to normal setting
                if ( !hooks || !("set" in hooks) || hooks.set( this, val,
"value" ) === undefined ) {
                        this.value = val;
                }
            });
        }
});

jQuery.extend({
        valHooks: {
            option: {
                get: function( elem ) {
                        // Use proper attribute retrieval(#6932, #12072)
                        var val = jQuery.find.attr( elem, "value" );
                        return val != null ?
                                val :
                                elem.text;
                }
            },
            select: {
                get: function( elem ) {
                        var value, option,
                                options = elem.options,
                                index = elem.selectedIndex,
                                one = elem.type === "select-one" || index <
0,
                                values = one ? null : [],
                                max = one ? index + 1 : options.length,
                                i = index < 0 ?
                                        max :
                                        one ? index : 0;
```

```
                        // Loop through all the selected options
                        for ( ; i < max; i++ ) {
                                option = options[ i ];

                                // oldIE doesn't update selected after form
reset (#2551)
                                if ( ( option.selected || i === index ) &&
                                                // Don't return options that are
disabled or in a disabled optgroup
                                                ( jQuery.support.optDisabled ?
!option.disabled : option.getAttribute("disabled") === null ) &&
                                                ( !option.parentNode.disabled ||
!jQuery.nodeName( option.parentNode, "optgroup" ) ) ) {

                                        // Get the specific value for the option
                                        value = jQuery( option ).val();

                                        // We don't need an array for one selects
                                        if ( one ) {
                                                return value;
                                        }

                                        // Multi-Selects return an array
                                        values.push( value );
                                }
                        }

                        return values;
                },

                set: function( elem, value ) {
                        var optionSet, option,
                                options = elem.options,
                                values = jQuery.makeArray( value ),
                                i = options.length;

                        while ( i-- ) {
                                option = options[ i ];
```

```
                                if ( (option.selected = jQuery.inArray(
jQuery(option).val(), values ) >= 0) ) {
                                        optionSet = true;
                                }
                        }

                        // force browsers to behave consistently when non-
matching value is set
                        if ( !optionSet ) {
                                elem.selectedIndex = -1;
                        }
                        return values;
                }
        }
    },

    attr: function( elem, name, value ) {
        var hooks, ret,
                nType = elem.nodeType;

        // don't get/set attributes on text, comment and attribute nodes
        if ( !elem || nType === 3 || nType === 8 || nType === 2 ) {
                return;
        }

        // Fallback to prop when attributes are not supported
        if ( typeof elem.getAttribute === core_strundefined ) {
                return jQuery.prop( elem, name, value );
        }

        // All attributes are lowercase
        // Grab necessary hook if one is defined
        if ( nType !== 1 || !jQuery.isXMLDoc( elem ) ) {
                name = name.toLowerCase();
                hooks = jQuery.attrHooks[ name ] ||
                        ( jQuery.expr.match.bool.test( name ) ? boolHook :
nodeHook );
        }
```

```
		if ( value !== undefined ) {

			if ( value === null ) {
				jQuery.removeAttr( elem, name );

			} else if ( hooks && "set" in hooks && (ret = hooks.set(
elem, value, name )) !== undefined ) {
				return ret;

			} else {
				elem.setAttribute( name, value + "" );
				return value;
			}

		} else if ( hooks && "get" in hooks && (ret = hooks.get( elem,
name )) !== null ) {
			return ret;

		} else {
			ret = jQuery.find.attr( elem, name );

			// Non-existent attributes return null, we normalize to
undefined
			return ret == null ?
				undefined :
				ret;
		}
	},

	removeAttr: function( elem, value ) {
		var name, propName,
			i = 0,
			attrNames = value && value.match( core_rnotwhite );

		if ( attrNames && elem.nodeType === 1 ) {
			while ( (name = attrNames[i++]) ) {
				propName = jQuery.propFix[ name ] || name;
```

```
                        // Boolean attributes get special treatment (#10870)
                        if ( jQuery.expr.match.bool.test( name ) ) {
                                // Set corresponding property to false
                                if ( getSetInput && getSetAttribute ||
!ruseDefault.test( name ) ) {
                                        elem[ propName ] = false;
                                // Support: IE<9
                                // Also clear defaultChecked/defaultSelected (if
appropriate)
                                } else {
                                        elem[ jQuery.camelCase( "default-" +
name ) ] =
                                                elem[ propName ] = false;
                                }

                        // See #9699 for explanation of this approach
(setting first, then removal)
                        } else {
                                jQuery.attr( elem, name, "" );
                        }

                        elem.removeAttribute( getSetAttribute ? name :
propName );
                }
            }
        },

        attrHooks: {
            type: {
                set: function( elem, value ) {
                        if ( !jQuery.support.radioValue && value === "radio"
&& jQuery.nodeName(elem, "input") ) {
                                // Setting the type on a radio button after the
value resets the value in IE6-9
                                // Reset value to default in case type is set
after value during creation
                                var val = elem.value;
```

```
                                elem.setAttribute( "type", value );
                                if ( val ) {
                                        elem.value = val;
                                }
                                return value;
                        }
                }
        }
    },

    propFix: {
        "for": "htmlFor",
        "class": "className"
    },

    prop: function( elem, name, value ) {
        var ret, hooks, notxml,
            nType = elem.nodeType;

        // don't get/set properties on text, comment and attribute nodes
        if ( !elem || nType === 3 || nType === 8 || nType === 2 ) {
            return;
        }

        notxml = nType !== 1 || !jQuery.isXMLDoc( elem );

        if ( notxml ) {
            // Fix name and attach hooks
            name = jQuery.propFix[ name ] || name;
            hooks = jQuery.propHooks[ name ];
        }

        if ( value !== undefined ) {
            return hooks && "set" in hooks && (ret = hooks.set( elem,
value, name )) !== undefined ?
                    ret :
                    ( elem[ name ] = value );
```

```
		} else {
			return hooks && "get" in hooks && (ret = hooks.get( elem,
name )) !== null ?
				ret :
				elem[ name ];
		}
	},

	propHooks: {
		tabIndex: {
			get: function( elem ) {
				// elem.tabIndex doesn't always return the correct
value when it hasn't been explicitly set
				// http://fluidproject.org/blog/2008/01/09/getting-
setting-and-removing-tabindex-values-with-javascript/
				// Use proper attribute retrieval(#12072)
				var tabindex = jQuery.find.attr( elem, "tabindex" );

				return tabindex ?
					parseInt( tabindex, 10 ) :
					rfocusable.test( elem.nodeName ) ||
rclickable.test( elem.nodeName ) && elem.href ?
						0 :
						-1;
			}
		}
	}
});

// Hooks for boolean attributes
boolHook = {
	set: function( elem, value, name ) {
		if ( value === false ) {
			// Remove boolean attributes when set to false
			jQuery.removeAttr( elem, name );
		} else if ( getSetInput && getSetAttribute || !ruseDefault.test(
name ) ) {
			// IE<8 needs the *property* name
```

```
                    elem.setAttribute( !getSetAttribute && jQuery.propFix[
name ] || name, name );

            // Use defaultChecked and defaultSelected for oldIE
            } else {
                    elem[ jQuery.camelCase( "default-" + name ) ] = elem[
name ] = true;
            }

            return name;
        }
};
jQuery.each( jQuery.expr.match.bool.source.match( /\w+/g ), function( i,
name ) {
        var getter = jQuery.expr.attrHandle[ name ] || jQuery.find.attr;

        jQuery.expr.attrHandle[ name ] = getSetInput && getSetAttribute ||
!ruseDefault.test( name ) ?
            function( elem, name, isXML ) {
                    var fn = jQuery.expr.attrHandle[ name ],
                            ret = isXML ?
                                    undefined :
                                    /* jshint eqeqeq: false */
                                    (jQuery.expr.attrHandle[ name ] = undefined)
!=
                                            getter( elem, name, isXML ) ?

                                            name.toLowerCase() :
                                            null;
                    jQuery.expr.attrHandle[ name ] = fn;
                    return ret;
            } :
            function( elem, name, isXML ) {
                    return isXML ?
                            undefined :
                            elem[ jQuery.camelCase( "default-" + name ) ] ?
                                    name.toLowerCase() :
                                    null;
```

```
            };
});

// fix oldIE attroperties
if ( !getSetInput || !getSetAttribute ) {
        jQuery.attrHooks.value = {
                set: function( elem, value, name ) {
                        if ( jQuery.nodeName( elem, "input" ) ) {
                                // Does not return so that setAttribute is also used
                                elem.defaultValue = value;
                        } else {
                                // Use nodeHook if defined (#1954); otherwise
setAttribute is fine
                                return nodeHook && nodeHook.set( elem, value,
name );
                        }
                }
        };
}

// IE6/7 do not support getting/setting some attributes with get/setAttribute
if ( !getSetAttribute ) {

        // Use this for any attribute in IE6/7
        // This fixes almost every IE6/7 issue
        nodeHook = {
                set: function( elem, value, name ) {
                        // Set the existing or create a new attribute node
                        var ret = elem.getAttributeNode( name );
                        if ( !ret ) {
                                elem.setAttributeNode(
                                        (ret = elem.ownerDocument.createAttribute(
name ))
                                );
                        }

                        ret.value = value += "";
```

```
                    // Break association with cloned elements by also using
setAttribute (#9646)
                    return name === "value" || value === elem.getAttribute(
name ) ?
                        value :
                        undefined;
            }
        };
        jQuery.expr.attrHandle.id = jQuery.expr.attrHandle.name =
jQuery.expr.attrHandle.coords =
            // Some attributes are constructed with empty-string values
when not defined
            function( elem, name, isXML ) {
                var ret;
                return isXML ?
                    undefined :
                    (ret = elem.getAttributeNode( name )) && ret.value
!== "" ?
                        ret.value :
                        null;
            };
        jQuery.valHooks.button = {
            get: function( elem, name ) {
                var ret = elem.getAttributeNode( name );
                return ret && ret.specified ?
                    ret.value :
                    undefined;
            },
            set: nodeHook.set
        };

        // Set contenteditable to false on removals(#10429)
        // Setting to empty string throws an error as an invalid value
        jQuery.attrHooks.contenteditable = {
            set: function( elem, value, name ) {
                nodeHook.set( elem, value === "" ? false : value, name );
            }
        };
```

```
	// Set width and height to auto instead of 0 on empty string( Bug
#8150 )
	// This is for removals
	jQuery.each([ "width", "height" ], function( i, name ) {
		jQuery.attrHooks[ name ] = {
			set: function( elem, value ) {
				if ( value === "" ) {
					elem.setAttribute( name, "auto" );
					return value;
				}
			}
		};
	});
}


// Some attributes require a special call on IE
// http://msdn.microsoft.com/en-us/library/ms536429%28VS.85%29.aspx
if ( !jQuery.support.hrefNormalized ) {
	// href/src property should get the full normalized URL
(#10299/#12915)
	jQuery.each([ "href", "src" ], function( i, name ) {
		jQuery.propHooks[ name ] = {
			get: function( elem ) {
				return elem.getAttribute( name, 4 );
			}
		};
	});
}

if ( !jQuery.support.style ) {
	jQuery.attrHooks.style = {
		get: function( elem ) {
			// Return undefined in the case of empty string
			// Note: IE uppercases css property names, but if we were
to .toLowerCase()
			// .cssText, that would destroy case senstitivity in URL's,
```

```
like in "background"
                        return elem.style.cssText || undefined;
                },
                set: function( elem, value ) {
                        return ( elem.style.cssText = value + "" );
                }
        };
}

// Safari mis-reports the default selected property of an option
// Accessing the parent's selectedIndex property fixes it
if ( !jQuery.support.optSelected ) {
        jQuery.propHooks.selected = {
                get: function( elem ) {
                        var parent = elem.parentNode;

                        if ( parent ) {
                                parent.selectedIndex;

                                // Make sure that it also works with optgroups, see
#5701
                                if ( parent.parentNode ) {
                                        parent.parentNode.selectedIndex;
                                }
                        }
                        return null;
                }
        };
}

jQuery.each([
        "tabIndex",
        "readOnly",
        "maxLength",
        "cellSpacing",
        "cellPadding",
        "rowSpan",
        "colSpan",
```

```
	"useMap",
	"frameBorder",
	"contentEditable"
], function() {
	jQuery.propFix[ this.toLowerCase() ] = this;
});

// IE6/7 call enctype encoding
if ( !jQuery.support.enctype ) {
	jQuery.propFix.enctype = "encoding";
}

// Radios and checkboxes getter/setter
jQuery.each([ "radio", "checkbox" ], function() {
	jQuery.valHooks[ this ] = {
		set: function( elem, value ) {
			if ( jQuery.isArray( value ) ) {
				return ( elem.checked = jQuery.inArray(
jQuery(elem).val(), value ) >= 0 );
			}
		}
	};
	if ( !jQuery.support.checkOn ) {
		jQuery.valHooks[ this ].get = function( elem ) {
			// Support: Webkit
			// "" is returned instead of "on" if a value isn't specified
			return elem.getAttribute("value") === null ? "on" :
elem.value;
		};
	}
});
var rformElems = /^(?:input|select|textarea)$/i,
	rkeyEvent = /^key/,
	rmouseEvent = /^(?:mouse|contextmenu)|click/,
	rfocusMorph = /^(?:focusinfocus|focusoutblur)$/,
	rtypenamespace = /^([^.]*)(?:\.(.+)|)$/;

function returnTrue() {
```

```
	return true;
}

function returnFalse() {
	return false;
}

function safeActiveElement() {
	try {
		return document.activeElement;
	} catch ( err ) { }
}

/*
 * Helper functions for managing events -- not part of the public interface.
 * Props to Dean Edwards' addEvent library for many of the ideas.
 */
jQuery.event = {

	global: {},

	add: function( elem, types, handler, data, selector ) {
		var tmp, events, t, handleObjIn,
			special, eventHandle, handleObj,
			handlers, type, namespaces, origType,
			elemData = jQuery._data( elem );

		// Don't attach events to noData or text/comment nodes (but
allow plain objects)
		if ( !elemData ) {
			return;
		}

		// Caller can pass in an object of custom data in lieu of the
handler
		if ( handler.handler ) {
			handleObjIn = handler;
			handler = handleObjIn.handler;
```

```
                        selector = handleObjIn.selector;
                }

                // Make sure that the handler has a unique ID, used to
find/remove it later
                if ( !handler.guid ) {
                        handler.guid = jQuery.guid++;
                }

                // Init the element's event structure and main handler, if this is
the first
                if ( !(events = elemData.events) ) {
                        events = elemData.events = {};
                }
                if ( !(eventHandle = elemData.handle) ) {
                        eventHandle = elemData.handle = function( e ) {
                                // Discard the second event of a
jQuery.event.trigger() and
                                // when an event is called after a page has unloaded
                                return typeof jQuery !== core_strundefined && (!e
|| jQuery.event.triggered !== e.type) ?
                                        jQuery.event.dispatch.apply(
eventHandle.elem, arguments ) :
                                        undefined;
                        };
                        // Add elem as a property of the handle fn to prevent a
memory leak with IE non-native events
                        eventHandle.elem = elem;
                }

                // Handle multiple events separated by a space
                types = ( types || "" ).match( core_rnotwhite ) || [""];
                t = types.length;
                while ( t-- ) {
                        tmp = rtypenamespace.exec( types[t] ) || [];
                        type = origType = tmp[1];
                        namespaces = ( tmp[2] || "" ).split( "." ).sort();
```

```
                    // There *must* be a type, no attaching namespace-only
handlers
                    if ( !type ) {
                            continue;
                    }

                    // If event changes its type, use the special event handlers
for the changed type
                    special = jQuery.event.special[ type ] || {};

                    // If selector defined, determine special event api type,
otherwise given type
                    type = ( selector ? special.delegateType : special.bindType
) || type;

                    // Update special based on newly reset type
                    special = jQuery.event.special[ type ] || {};

                    // handleObj is passed to all event handlers
                    handleObj = jQuery.extend({
                            type: type,
                            origType: origType,
                            data: data,
                            handler: handler,
                            guid: handler.guid,
                            selector: selector,
                            needsContext: selector &&
jQuery.expr.match.needsContext.test( selector ),
                            namespace: namespaces.join(".")
                    }, handleObjIn );

                    // Init the event handler queue if we're the first
                    if ( !(handlers = events[ type ]) ) {
                            handlers = events[ type ] = [];
                            handlers.delegateCount = 0;

                            // Only use addEventListener/attachEvent if the
special events handler returns false
```

```
                        if ( !special.setup || special.setup.call( elem, data,
namespaces, eventHandle ) === false ) {
                                // Bind the global event handler to the element
                                if ( elem.addEventListener ) {
                                        elem.addEventListener( type,
eventHandle, false );

                                } else if ( elem.attachEvent ) {
                                        elem.attachEvent( "on" + type,
eventHandle );
                                }
                        }
                }

                if ( special.add ) {
                        special.add.call( elem, handleObj );

                        if ( !handleObj.handler.guid ) {
                                handleObj.handler.guid = handler.guid;
                        }
                }

                // Add to the element's handler list, delegates in front
                if ( selector ) {
                        handlers.splice( handlers.delegateCount++, 0,
handleObj );
                } else {
                        handlers.push( handleObj );
                }

                // Keep track of which events have ever been used, for
event optimization
                jQuery.event.global[ type ] = true;
        }

        // Nullify elem to prevent memory leaks in IE
        elem = null;
    },
```

```
	// Detach an event or set of events from an element
	remove: function( elem, types, handler, selector, mappedTypes ) {
		var j, handleObj, tmp,
			origCount, t, events,
			special, handlers, type,
			namespaces, origType,
			elemData = jQuery.hasData( elem ) && jQuery._data(
elem );

		if ( !elemData || !(events = elemData.events) ) {
			return;
		}

		// Once for each type.namespace in types; type may be omitted
		types = ( types || "" ).match( core_rnotwhite ) || [""];
		t = types.length;
		while ( t-- ) {
			tmp = rtypenamespace.exec( types[t] ) || [];
			type = origType = tmp[1];
			namespaces = ( tmp[2] || "" ).split( "." ).sort();

			// Unbind all events (on this namespace, if provided) for
the element
			if ( !type ) {
				for ( type in events ) {
					jQuery.event.remove( elem, type + types[ t ],
handler, selector, true );
				}
				continue;
			}

			special = jQuery.event.special[ type ] || {};
			type = ( selector ? special.delegateType : special.bindType
) || type;
			handlers = events[ type ] || [];
			tmp = tmp[2] && new RegExp( "(^|\\.)" +
namespaces.join("\\.(?:.*\\.|)") + "(\\.|$)" );
```

```
                    // Remove matching events
                    origCount = j = handlers.length;
                    while ( j-- ) {
                            handleObj = handlers[ j ];

                            if ( ( mappedTypes || origType ===
handleObj.origType ) &&
                                    ( !handler || handler.guid === handleObj.guid
) &&
                                    ( !tmp || tmp.test( handleObj.namespace ) )
&&
                                    ( !selector || selector === handleObj.selector
|| selector === "**" && handleObj.selector ) ) {
                                    handlers.splice( j, 1 );

                                    if ( handleObj.selector ) {
                                            handlers.delegateCount--;
                                    }
                                    if ( special.remove ) {
                                            special.remove.call( elem, handleObj );
                                    }
                            }
                    }

                    // Remove generic event handler if we removed something
and no more handlers exist
                    // (avoids potential for endless recursion during removal of
special event handlers)
                    if ( origCount && !handlers.length ) {
                            if ( !special.teardown || special.teardown.call( elem,
namespaces, elemData.handle ) === false ) {
                                    jQuery.removeEvent( elem, type,
elemData.handle );
                            }

                            delete events[ type ];
                    }
```

```
		}

		// Remove the expando if it's no longer used
		if ( jQuery.isEmptyObject( events ) ) {
			delete elemData.handle;

			// removeData also checks for emptiness and clears the
expando if empty
			// so use it instead of delete
			jQuery._removeData( elem, "events" );
		}
	},

	trigger: function( event, data, elem, onlyHandlers ) {
		var handle, ontype, cur,
			bubbleType, special, tmp, i,
			eventPath = [ elem || document ],
			type = core_hasOwn.call( event, "type" ) ? event.type :
event,
			namespaces = core_hasOwn.call( event, "namespace" ) ?
event.namespace.split(".") : [];

		cur = tmp = elem = elem || document;

		// Don't do events on text and comment nodes
		if ( elem.nodeType === 3 || elem.nodeType === 8 ) {
			return;
		}

		// focus/blur morphs to focusin/out; ensure we're not firing them
right now
		if ( rfocusMorph.test( type + jQuery.event.triggered ) ) {
			return;
		}

		if ( type.indexOf(".") >= 0 ) {
			// Namespaced trigger; create a regexp to match event
type in handle()
```

```
                        namespaces = type.split(".");
                        type = namespaces.shift();
                        namespaces.sort();
                }
                ontype = type.indexOf(":") < 0 && "on" + type;

                // Caller can pass in a jQuery.Event object, Object, or just an
event type string
                event = event[ jQuery.expando ] ?
                        event :
                        new jQuery.Event( type, typeof event === "object" &&
event );

                // Trigger bitmask: & 1 for native handlers; & 2 for jQuery
(always true)
                event.isTrigger = onlyHandlers ? 2 : 3;
                event.namespace = namespaces.join(".");
                event.namespace_re = event.namespace ?
                        new RegExp( "(^|\\.)" + namespaces.join("\\.(?:.*\\.|)")
+ "(\\.|$)" ) :
                        null;

                // Clean up the event in case it is being reused
                event.result = undefined;
                if ( !event.target ) {
                        event.target = elem;
                }

                // Clone any incoming data and prepend the event, creating the
handler arg list
                data = data == null ?
                        [ event ] :
                        jQuery.makeArray( data, [ event ] );

                // Allow special events to draw outside the lines
                special = jQuery.event.special[ type ] || {};
                if ( !onlyHandlers && special.trigger && special.trigger.apply(
elem, data ) === false ) {
```

```
                return;
            }

            // Determine event propagation path in advance, per W3C
events spec (#9951)
            // Bubble up to document, then to window; watch for a global
ownerDocument var (#9724)
            if ( !onlyHandlers && !special.noBubble && !jQuery.isWindow(
elem ) ) {

                bubbleType = special.delegateType || type;
                if ( !rfocusMorph.test( bubbleType + type ) ) {
                    cur = cur.parentNode;
                }
                for ( ; cur; cur = cur.parentNode ) {
                    eventPath.push( cur );
                    tmp = cur;
                }

                // Only add window if we got to document (e.g., not plain
obj or detached DOM)
                if ( tmp === (elem.ownerDocument || document) ) {
                    eventPath.push( tmp.defaultView ||
tmp.parentWindow || window );
                }
            }

            // Fire handlers on the event path
            i = 0;
            while ( (cur = eventPath[i++]) &&
!event.isPropagationStopped() ) {

                event.type = i > 1 ?
                    bubbleType :
                    special.bindType || type;

                // jQuery handler
                handle = ( jQuery._data( cur, "events" ) || {} )[
```

```
event.type ] && jQuery._data( cur, "handle" );
                    if ( handle ) {
                            handle.apply( cur, data );
                    }

                    // Native handler
                    handle = ontype && cur[ ontype ];
                    if ( handle && jQuery.acceptData( cur ) && handle.apply
&& handle.apply( cur, data ) === false ) {
                            event.preventDefault();
                    }
            }
            event.type = type;

            // If nobody prevented the default action, do it now
            if ( !onlyHandlers && !event.isDefaultPrevented() ) {

                    if ( (!special._default || special._default.apply(
eventPath.pop(), data ) === false) &&
                            jQuery.acceptData( elem ) ) {

                            // Call a native DOM method on the target with the
same name name as the event.
                            // Can't use an .isFunction() check here because
IE6/7 fails that test.
                            // Don't do default actions on window, that's where
global variables be (#6170)
                            if ( ontype && elem[ type ] && !jQuery.isWindow(
elem ) ) {

                                    // Don't re-trigger an onFOO event when we
call its FOO() method
                                    tmp = elem[ ontype ];

                                    if ( tmp ) {
                                            elem[ ontype ] = null;
                                    }
```

```
                                // Prevent re-triggering of the same event,
since we already bubbled it above
                                jQuery.event.triggered = type;
                                try {
                                        elem[ type ]();
                                } catch ( e ) {
                                        // IE<9 dies on focus/blur to hidden
element (#1486,#12518)
                                        // only reproducible on winXP IE8 native,
not IE9 in IE8 mode
                                }
                                jQuery.event.triggered = undefined;

                                if ( tmp ) {
                                        elem[ ontype ] = tmp;
                                }
                        }
                }
        }

        return event.result;
    },

    dispatch: function( event ) {

        // Make a writable jQuery.Event from the native event object
        event = jQuery.event.fix( event );

        var i, ret, handleObj, matched, j,
                handlerQueue = [],
                args = core_slice.call( arguments ),
                handlers = ( jQuery._data( this, "events" ) || {} )[
event.type ] || [],
                special = jQuery.event.special[ event.type ] || {};

        // Use the fix-ed jQuery.Event rather than the (read-only) native
event
        args[0] = event;
```

```
		event.delegateTarget = this;

		// Call the preDispatch hook for the mapped type, and let it bail
if desired
		if ( special.preDispatch && special.preDispatch.call( this, event )
=== false ) {
			return;
		}

		// Determine handlers
		handlerQueue = jQuery.event.handlers.call( this, event, handlers
);

		// Run delegates first; they may want to stop propagation
beneath us
		i = 0;
		while ( (matched = handlerQueue[ i++ ]) &&
!event.isPropagationStopped() ) {
			event.currentTarget = matched.elem;

			j = 0;
			while ( (handleObj = matched.handlers[ j++ ]) &&
!event.isImmediatePropagationStopped() ) {

				// Triggered event must either 1) have no
namespace, or
				// 2) have namespace(s) a subset or equal to those
in the bound event (both can have no namespace).
				if ( !event.namespace_re ||
event.namespace_re.test( handleObj.namespace ) ) {

					event.handleObj = handleObj;
					event.data = handleObj.data;

					ret = ( (jQuery.event.special[
handleObj.origType ] || {}).handle || handleObj.handler )
						.apply( matched.elem, args );
```

```
                                if ( ret !== undefined ) {
                                        if ( (event.result = ret) === false ) {
                                                event.preventDefault();
                                                event.stopPropagation();
                                        }
                                }
                        }
                }
        }

        // Call the postDispatch hook for the mapped type
        if ( special.postDispatch ) {
                special.postDispatch.call( this, event );
        }

        return event.result;
    },

    handlers: function( event, handlers ) {
        var sel, handleObj, matches, i,
                handlerQueue = [],
                delegateCount = handlers.delegateCount,
                cur = event.target;

        // Find delegate handlers
        // Black-hole SVG <use> instance trees (#13180)
        // Avoid non-left-click bubbling in Firefox (#3861)
        if ( delegateCount && cur.nodeType && (!event.button ||
event.type !== "click") ) {

                /* jshint eqeqeq: false */
                for ( ; cur != this; cur = cur.parentNode || this ) {
                        /* jshint eqeqeq: true */

                        // Don't check non-elements (#13208)
                        // Don't process clicks on disabled elements (#6911,
#8165, #11382, #11764)
                        if ( cur.nodeType === 1 && (cur.disabled !== true
```

```
|| event.type !== "click") ) {
                                matches = [];
                                for ( i = 0; i < delegateCount; i++ ) {
                                        handleObj = handlers[ i ];

                                        // Don't conflict with Object.prototype
properties (#13203)
                                        sel = handleObj.selector + " ";

                                        if ( matches[ sel ] === undefined ) {
                                                matches[ sel ] =
handleObj.needsContext ?
                                                        jQuery( sel, this ).index( cur
) >= 0 :
                                                        jQuery.find( sel, this, null, [
cur ] ).length;
                                        }
                                        if ( matches[ sel ] ) {
                                                matches.push( handleObj );
                                        }
                                }
                                if ( matches.length ) {
                                        handlerQueue.push({ elem: cur,
handlers: matches });
                                }
                        }
                }
        }

        // Add the remaining (directly-bound) handlers
        if ( delegateCount < handlers.length ) {
                handlerQueue.push({ elem: this, handlers: handlers.slice(
delegateCount ) });
        }

        return handlerQueue;
    },
```

```
        fix: function( event ) {
                if ( event[ jQuery.expando ] ) {
                        return event;
                }

                // Create a writable copy of the event object and normalize some
properties
                var i, prop, copy,
                        type = event.type,
                        originalEvent = event,
                        fixHook = this.fixHooks[ type ];

                if ( !fixHook ) {
                        this.fixHooks[ type ] = fixHook =
                                rmouseEvent.test( type ) ? this.mouseHooks :
                                rkeyEvent.test( type ) ? this.keyHooks :
                                {};
                }
                copy = fixHook.props ? this.props.concat( fixHook.props ) :
this.props;

                event = new jQuery.Event( originalEvent );

                i = copy.length;
                while ( i-- ) {
                        prop = copy[ i ];
                        event[ prop ] = originalEvent[ prop ];
                }

                // Support: IE<9
                // Fix target property (#1925)
                if ( !event.target ) {
                        event.target = originalEvent.srcElement || document;
                }

                // Support: Chrome 23+, Safari?
                // Target should not be a text node (#504, #13143)
                if ( event.target.nodeType === 3 ) {
```

```
                        event.target = event.target.parentNode;
                }

                // Support: IE<9
                // For mouse/key events, metaKey==false if it's undefined
(#3368, #11328)
                event.metaKey = !!event.metaKey;

                return fixHook.filter ? fixHook.filter( event, originalEvent ) :
event;
        },

        // Includes some event props shared by KeyEvent and MouseEvent
        props: "altKey bubbles cancelable ctrlKey currentTarget eventPhase
metaKey relatedTarget shiftKey target timeStamp view which".split(" "),

        fixHooks: {},

        keyHooks: {
                props: "char charCode key keyCode".split(" "),
                filter: function( event, original ) {

                        // Add which for key events
                        if ( event.which == null ) {
                                event.which = original.charCode != null ?
original.charCode : original.keyCode;
                        }

                        return event;
                }
        },

        mouseHooks: {
                props: "button buttons clientX clientY fromElement offsetX
offsetY pageX pageY screenX screenY toElement".split(" "),
                filter: function( event, original ) {
                        var body, eventDoc, doc,
                                button = original.button,
```

```
                        fromElement = original.fromElement;

                // Calculate pageX/Y if missing and clientX/Y available
                if ( event.pageX == null && original.clientX != null ) {
                        eventDoc = event.target.ownerDocument ||
document;
                        doc = eventDoc.documentElement;
                        body = eventDoc.body;

                        event.pageX = original.clientX + ( doc &&
doc.scrollLeft || body && body.scrollLeft || 0 ) - ( doc && doc.clientLeft ||
body && body.clientLeft || 0 );
                        event.pageY = original.clientY + ( doc &&
doc.scrollTop  || body && body.scrollTop  || 0 ) - ( doc && doc.clientTop  ||
body && body.clientTop  || 0 );
                }

                // Add relatedTarget, if necessary
                if ( !event.relatedTarget && fromElement ) {
                        event.relatedTarget = fromElement ===
event.target ? original.toElement : fromElement;
                }

                // Add which for click: 1 === left; 2 === middle; 3 ===
right
                // Note: button is not normalized, so don't use it
                if ( !event.which && button !== undefined ) {
                        event.which = ( button & 1 ? 1 : ( button & 2 ? 3 : (
button & 4 ? 2 : 0 ) ) );
                }

                return event;
        }
    },

    special: {
        load: {
                // Prevent triggered image.load events from bubbling to
```

```
window.load
                noBubble: true
            },
            focus: {
                // Fire native event if possible so blur/focus sequence is
correct
                trigger: function() {
                    if ( this !== safeActiveElement() && this.focus ) {
                        try {
                            this.focus();
                            return false;
                        } catch ( e ) {
                            // Support: IE<9
                            // If we error on focus to hidden element
(#1486, #12518),
                            // let .trigger() run the handlers
                        }
                    }
                },
                delegateType: "focusin"
            },
            blur: {
                trigger: function() {
                    if ( this === safeActiveElement() && this.blur ) {
                        this.blur();
                        return false;
                    }
                },
                delegateType: "focusout"
            },
            click: {
                // For checkbox, fire native event so checked state will be
right
                trigger: function() {
                    if ( jQuery.nodeName( this, "input" ) && this.type
=== "checkbox" && this.click ) {
                        this.click();
                        return false;
```

```
                        }
                },

                // For cross-browser consistency, don't fire native .click()
on links
                _default: function( event ) {
                        return jQuery.nodeName( event.target, "a" );
                }
        },

        beforeunload: {
                postDispatch: function( event ) {

                        // Even when returnValue equals to undefined
Firefox will still show alert
                        if ( event.result !== undefined ) {
                                event.originalEvent.returnValue = event.result;
                        }
                }
        }
    },

    simulate: function( type, elem, event, bubble ) {
        // Piggyback on a donor event to simulate a different one.
        // Fake originalEvent to avoid donor's stopPropagation, but if the
        // simulated event prevents default then we do the same on the
donor.
        var e = jQuery.extend(
                new jQuery.Event(),
                event,
                {
                        type: type,
                        isSimulated: true,
                        originalEvent: {}
                }
        );
        if ( bubble ) {
                jQuery.event.trigger( e, null, elem );
```

```
		} else {
			jQuery.event.dispatch.call( elem, e );
		}
		if ( e.isDefaultPrevented() ) {
			event.preventDefault();
		}
	}
};

jQuery.removeEvent = document.removeEventListener ?
	function( elem, type, handle ) {
		if ( elem.removeEventListener ) {
			elem.removeEventListener( type, handle, false );
		}
	} :
	function( elem, type, handle ) {
		var name = "on" + type;

		if ( elem.detachEvent ) {

			// #8545, #7054, preventing memory leaks for custom
events in IE6-8
			// detachEvent needed property on element, by name of
that event, to properly expose it to GC
			if ( typeof elem[ name ] === core_strundefined ) {
				elem[ name ] = null;
			}

			elem.detachEvent( name, handle );
		}
	};

jQuery.Event = function( src, props ) {
	// Allow instantiation without the 'new' keyword
	if ( !(this instanceof jQuery.Event) ) {
		return new jQuery.Event( src, props );
	}
```

```
	// Event object
	if ( src && src.type ) {
		this.originalEvent = src;
		this.type = src.type;

		// Events bubbling up the document may have been marked as
prevented
		// by a handler lower down the tree; reflect the correct value.
		this.isDefaultPrevented = ( src.defaultPrevented ||
src.returnValue === false ||
			src.getPreventDefault && src.getPreventDefault() ) ?
returnTrue : returnFalse;

	// Event type
	} else {
		this.type = src;
	}

	// Put explicitly provided properties onto the event object
	if ( props ) {
		jQuery.extend( this, props );
	}

	// Create a timestamp if incoming event doesn't have one
	this.timeStamp = src && src.timeStamp || jQuery.now();

	// Mark it as fixed
	this[ jQuery.expando ] = true;
};

// jQuery.Event is based on DOM3 Events as specified by the ECMAScript
Language Binding
// http://www.w3.org/TR/2003/WD-DOM-Level-3-Events-20030331/ecma-
script-binding.html
jQuery.Event.prototype = {
	isDefaultPrevented: returnFalse,
	isPropagationStopped: returnFalse,
	isImmediatePropagationStopped: returnFalse,
```

```
	preventDefault: function() {
		var e = this.originalEvent;

		this.isDefaultPrevented = returnTrue;
		if ( !e ) {
			return;
		}

		// If preventDefault exists, run it on the original event
		if ( e.preventDefault ) {
			e.preventDefault();

		// Support: IE
		// Otherwise set the returnValue property of the original event to
false
		} else {
			e.returnValue = false;
		}
	},
	stopPropagation: function() {
		var e = this.originalEvent;

		this.isPropagationStopped = returnTrue;
		if ( !e ) {
			return;
		}
		// If stopPropagation exists, run it on the original event
		if ( e.stopPropagation ) {
			e.stopPropagation();
		}

		// Support: IE
		// Set the cancelBubble property of the original event to true
		e.cancelBubble = true;
	},
	stopImmediatePropagation: function() {
		this.isImmediatePropagationStopped = returnTrue;
```

```
                this.stopPropagation();
        }
};

// Create mouseenter/leave events using mouseover/out and event-time
checks
jQuery.each({
        mouseenter: "mouseover",
        mouseleave: "mouseout"
}, function( orig, fix ) {
        jQuery.event.special[ orig ] = {
                delegateType: fix,
                bindType: fix,

                handle: function( event ) {
                        var ret,
                                target = this,
                                related = event.relatedTarget,
                                handleObj = event.handleObj;

                        // For mousenter/leave call the handler if related is outside
the target.
                        // NB: No relatedTarget if the mouse left/entered the
browser window
                        if ( !related || (related !== target && !jQuery.contains(
target, related )) ) {
                                event.type = handleObj.origType;
                                ret = handleObj.handler.apply( this, arguments );
                                event.type = fix;
                        }
                        return ret;
                }
        };
});

// IE submit delegation
if ( !jQuery.support.submitBubbles ) {
```

```
	jQuery.event.special.submit = {
		setup: function() {
			// Only need this for delegated form submit events
			if ( jQuery.nodeName( this, "form" ) ) {
				return false;
			}

			// Lazy-add a submit handler when a descendant form may
potentially be submitted
			jQuery.event.add( this, "click._submit keypress._submit",
function( e ) {
				// Node name check avoids a VML-related crash in IE
(#9807)
				var elem = e.target,
					form = jQuery.nodeName( elem, "input" ) ||
jQuery.nodeName( elem, "button" ) ? elem.form : undefined;
				if ( form && !jQuery._data( form, "submitBubbles" )
) {
					jQuery.event.add( form, "submit._submit",
function( event ) {
						event._submit_bubble = true;
					});
					jQuery._data( form, "submitBubbles", true );
				}
			});
			// return undefined since we don't need an event listener
		},

		postDispatch: function( event ) {
			// If form was submitted by the user, bubble the event up
the tree
			if ( event._submit_bubble ) {
				delete event._submit_bubble;
				if ( this.parentNode && !event.isTrigger ) {
					jQuery.event.simulate( "submit",
this.parentNode, event, true );
				}
			}
```

```
		},

		teardown: function() {
			// Only need this for delegated form submit events
			if ( jQuery.nodeName( this, "form" ) ) {
				return false;
			}

			// Remove delegated handlers; cleanData eventually reaps
submit handlers attached above
			jQuery.event.remove( this, "._submit" );
		}
	};
}

// IE change delegation and checkbox/radio fix
if ( !jQuery.support.changeBubbles ) {

	jQuery.event.special.change = {

		setup: function() {

			if ( rformElems.test( this.nodeName ) ) {
				// IE doesn't fire change on a check/radio until blur;
trigger it on click
				// after a propertychange. Eat the blur-change in
special.change.handle.
				// This still fires onchange a second time for
check/radio after blur.
				if ( this.type === "checkbox" || this.type ===
"radio" ) {
					jQuery.event.add( this,
"propertychange._change", function( event ) {
						if ( event.originalEvent.propertyName
=== "checked" ) {
							this._just_changed = true;
						}
					});
```

```
                                jQuery.event.add( this, "click._change",
function( event ) {
                                        if ( this._just_changed &&
!event.isTrigger ) {
                                                this._just_changed = false;
                                        }
                                        // Allow triggered, simulated change
events (#11500)
                                        jQuery.event.simulate( "change", this,
event, true );
                                });
                        }
                        return false;
                }
                // Delegated event; lazy-add a change handler on
descendant inputs
                jQuery.event.add( this, "beforeactivate._change",
function( e ) {
                        var elem = e.target;

                        if ( rformElems.test( elem.nodeName ) &&
!jQuery._data( elem, "changeBubbles" ) ) {
                                jQuery.event.add( elem, "change._change",
function( event ) {
                                        if ( this.parentNode &&
!event.isSimulated && !event.isTrigger ) {
                                                jQuery.event.simulate( "change",
this.parentNode, event, true );
                                        }
                                });
                                jQuery._data( elem, "changeBubbles", true );
                        }
                });
        },

        handle: function( event ) {
                var elem = event.target;
```

```
                        // Swallow native change events from checkbox/radio, we
already triggered them above
                        if ( this !== elem || event.isSimulated || event.isTrigger ||
(elem.type !== "radio" && elem.type !== "checkbox") ) {
                                return event.handleObj.handler.apply( this,
arguments );
                        }
                },

                teardown: function() {
                        jQuery.event.remove( this, "._change" );

                        return !rformElems.test( this.nodeName );
                }
        };
}

// Create "bubbling" focus and blur events
if ( !jQuery.support.focusinBubbles ) {
        jQuery.each({ focus: "focusin", blur: "focusout" }, function( orig, fix )
{

                // Attach a single capturing handler while someone wants
focusin/focusout
                var attaches = 0,
                        handler = function( event ) {
                                jQuery.event.simulate( fix, event.target,
jQuery.event.fix( event ), true );
                        };

                jQuery.event.special[ fix ] = {
                        setup: function() {
                                if ( attaches++ === 0 ) {
                                        document.addEventListener( orig, handler,
true );
                                }
                        },
                        teardown: function() {
```

```
                        if ( --attaches === 0 ) {
                                document.removeEventListener( orig, handler,
true );
                        }
                }
            };
        });
}

jQuery.fn.extend({

        on: function( types, selector, data, fn, /*INTERNAL*/ one ) {
            var type, origFn;

            // Types can be a map of types/handlers
            if ( typeof types === "object" ) {
                // ( types-Object, selector, data )
                if ( typeof selector !== "string" ) {
                    // ( types-Object, data )
                    data = data || selector;
                    selector = undefined;
                }
                for ( type in types ) {
                    this.on( type, selector, data, types[ type ], one );
                }
                return this;
            }

            if ( data == null && fn == null ) {
                // ( types, fn )
                fn = selector;
                data = selector = undefined;
            } else if ( fn == null ) {
                if ( typeof selector === "string" ) {
                    // ( types, selector, fn )
                    fn = data;
                    data = undefined;
                } else {
```

```
                        // ( types, data, fn )
                        fn = data;
                        data = selector;
                        selector = undefined;
                }
        }
        if ( fn === false ) {
                fn = returnFalse;
        } else if ( !fn ) {
                return this;
        }

        if ( one === 1 ) {
                origFn = fn;
                fn = function( event ) {
                        // Can use an empty set, since event contains the
info
                        jQuery().off( event );
                        return origFn.apply( this, arguments );
                };
                // Use same guid so caller can remove using origFn
                fn.guid = origFn.guid || ( origFn.guid = jQuery.guid++ );
        }
        return this.each( function() {
                jQuery.event.add( this, types, fn, data, selector );
        });
    },
    one: function( types, selector, data, fn ) {
        return this.on( types, selector, data, fn, 1 );
    },
    off: function( types, selector, fn ) {
        var handleObj, type;
        if ( types && types.preventDefault && types.handleObj ) {
                // ( event )  dispatched jQuery.Event
                handleObj = types.handleObj;
                jQuery( types.delegateTarget ).off(
                        handleObj.namespace ? handleObj.origType + "." +
handleObj.namespace : handleObj.origType,
```

```
                        handleObj.selector,
                        handleObj.handler
                );
                return this;
        }
        if ( typeof types === "object" ) {
                // ( types-object [, selector] )
                for ( type in types ) {
                        this.off( type, selector, types[ type ] );
                }
                return this;
        }
        if ( selector === false || typeof selector === "function" ) {
                // ( types [, fn] )
                fn = selector;
                selector = undefined;
        }
        if ( fn === false ) {
                fn = returnFalse;
        }
        return this.each(function() {
                jQuery.event.remove( this, types, fn, selector );
        });
    },

    trigger: function( type, data ) {
        return this.each(function() {
                jQuery.event.trigger( type, data, this );
        });
    },
    triggerHandler: function( type, data ) {
        var elem = this[0];
        if ( elem ) {
                return jQuery.event.trigger( type, data, elem, true );
        }
    }
});
var isSimple = /^.[^:#\[\.,]*$/,
```

```
	rparentsprev = /^(?:parents|prev(?:Until|All))/,
	rneedsContext = jQuery.expr.match.needsContext,
	// methods guaranteed to produce a unique set when starting from a
unique set
	guaranteedUnique = {
		children: true,
		contents: true,
		next: true,
		prev: true
	};

jQuery.fn.extend({
	find: function( selector ) {
		var i,
			ret = [],
			self = this,
			len = self.length;

		if ( typeof selector !== "string" ) {
			return this.pushStack( jQuery( selector ).filter(function() {
				for ( i = 0; i < len; i++ ) {
					if ( jQuery.contains( self[ i ], this ) ) {
						return true;
					}
				}
			}) );
		}

		for ( i = 0; i < len; i++ ) {
			jQuery.find( selector, self[ i ], ret );
		}

		// Needed because $( selector, context ) becomes $( context
).find( selector )
		ret = this.pushStack( len > 1 ? jQuery.unique( ret ) : ret );
		ret.selector = this.selector ? this.selector + " " + selector :
selector;
		return ret;
```

```
	},

	has: function( target ) {
		var i,
			targets = jQuery( target, this ),
			len = targets.length;

		return this.filter(function() {
			for ( i = 0; i < len; i++ ) {
				if ( jQuery.contains( this, targets[i] ) ) {
					return true;
				}
			}
		});
	},

	not: function( selector ) {
		return this.pushStack( winnow(this, selector || [], true) );
	},

	filter: function( selector ) {
		return this.pushStack( winnow(this, selector || [], false) );
	},

	is: function( selector ) {
		return !!winnow(
			this,

			// If this is a positional/relative selector, check
membership in the returned set
			// so $("p:first").is("p:last") won't return true for a doc
with two "p".
			typeof selector === "string" && rneedsContext.test(
selector ) ?
				jQuery( selector ) :
				selector || [],
			false
		).length;
```

```
	},

	closest: function( selectors, context ) {
		var cur,
			i = 0,
			l = this.length,
			ret = [],
			pos = rneedsContext.test( selectors ) || typeof selectors
!== "string" ?
				jQuery( selectors, context || this.context ) :
				0;

		for ( ; i < l; i++ ) {
			for ( cur = this[i]; cur && cur !== context; cur =
cur.parentNode ) {
				// Always skip document fragments
				if ( cur.nodeType < 11 && (pos ?
					pos.index(cur) > -1 :

					// Don't pass non-elements to Sizzle
					cur.nodeType === 1 &&
						jQuery.find.matchesSelector(cur,
selectors)) ) {

					cur = ret.push( cur );
					break;
				}
			}
		}

		return this.pushStack( ret.length > 1 ? jQuery.unique( ret ) : ret
);
	},

	// Determine the position of an element within
	// the matched set of elements
	index: function( elem ) {
```

```
		// No argument, return index in parent
		if ( !elem ) {
			return ( this[0] && this[0].parentNode ) ?
this.first().prevAll().length : -1;
		}

		// index in selector
		if ( typeof elem === "string" ) {
			return jQuery.inArray( this[0], jQuery( elem ) );
		}

		// Locate the position of the desired element
		return jQuery.inArray(
			// If it receives a jQuery object, the first element is used
			elem.jquery ? elem[0] : elem, this );
	},

	add: function( selector, context ) {
		var set = typeof selector === "string" ?
				jQuery( selector, context ) :
				jQuery.makeArray( selector && selector.nodeType ?
[ selector ] : selector ),
			all = jQuery.merge( this.get(), set );

		return this.pushStack( jQuery.unique(all) );
	},

	addBack: function( selector ) {
		return this.add( selector == null ?
			this.prevObject : this.prevObject.filter(selector)
		);
	}
});

function sibling( cur, dir ) {
	do {
		cur = cur[ dir ];
	} while ( cur && cur.nodeType !== 1 );
```

```
	return cur;
}

jQuery.each({
	parent: function( elem ) {
		var parent = elem.parentNode;
		return parent && parent.nodeType !== 11 ? parent : null;
	},
	parents: function( elem ) {
		return jQuery.dir( elem, "parentNode" );
	},
	parentsUntil: function( elem, i, until ) {
		return jQuery.dir( elem, "parentNode", until );
	},
	next: function( elem ) {
		return sibling( elem, "nextSibling" );
	},
	prev: function( elem ) {
		return sibling( elem, "previousSibling" );
	},
	nextAll: function( elem ) {
		return jQuery.dir( elem, "nextSibling" );
	},
	prevAll: function( elem ) {
		return jQuery.dir( elem, "previousSibling" );
	},
	nextUntil: function( elem, i, until ) {
		return jQuery.dir( elem, "nextSibling", until );
	},
	prevUntil: function( elem, i, until ) {
		return jQuery.dir( elem, "previousSibling", until );
	},
	siblings: function( elem ) {
		return jQuery.sibling( ( elem.parentNode || {} ).firstChild, elem
);
	},
	children: function( elem ) {
```

```
		return jQuery.sibling( elem.firstChild );
	},
	contents: function( elem ) {
		return jQuery.nodeName( elem, "iframe" ) ?
			elem.contentDocument || elem.contentWindow.document :
			jQuery.merge( [], elem.childNodes );
	}
}, function( name, fn ) {
	jQuery.fn[ name ] = function( until, selector ) {
		var ret = jQuery.map( this, fn, until );

		if ( name.slice( -5 ) !== "Until" ) {
			selector = until;
		}

		if ( selector && typeof selector === "string" ) {
			ret = jQuery.filter( selector, ret );
		}

		if ( this.length > 1 ) {
			// Remove duplicates
			if ( !guaranteedUnique[ name ] ) {
				ret = jQuery.unique( ret );
			}

			// Reverse order for parents* and prev-derivatives
			if ( rparentsprev.test( name ) ) {
				ret = ret.reverse();
			}
		}

		return this.pushStack( ret );
	};
});

jQuery.extend({
	filter: function( expr, elems, not ) {
		var elem = elems[ 0 ];
```

```
		if ( not ) {
			expr = ":not(" + expr + ")";
		}

		return elems.length === 1 && elem.nodeType === 1 ?
			jQuery.find.matchesSelector( elem, expr ) ? [ elem ] : [] :
			jQuery.find.matches( expr, jQuery.grep( elems, function(
elem ) {
				return elem.nodeType === 1;
			}));
	},

	dir: function( elem, dir, until ) {
		var matched = [],
			cur = elem[ dir ];

		while ( cur && cur.nodeType !== 9 && (until === undefined ||
cur.nodeType !== 1 || !jQuery( cur ).is( until )) ) {
			if ( cur.nodeType === 1 ) {
				matched.push( cur );
			}
			cur = cur[dir];
		}
		return matched;
	},

	sibling: function( n, elem ) {
		var r = [];

		for ( ; n; n = n.nextSibling ) {
			if ( n.nodeType === 1 && n !== elem ) {
				r.push( n );
			}
		}

		return r;
	}
```

```
});

// Implement the identical functionality for filter and not
function winnow( elements, qualifier, not ) {
	if ( jQuery.isFunction( qualifier ) ) {
		return jQuery.grep( elements, function( elem, i ) {
			/* jshint -W018 */
			return !!qualifier.call( elem, i, elem ) !== not;
		});

	}

	if ( qualifier.nodeType ) {
		return jQuery.grep( elements, function( elem ) {
			return ( elem === qualifier ) !== not;
		});

	}

	if ( typeof qualifier === "string" ) {
		if ( isSimple.test( qualifier ) ) {
			return jQuery.filter( qualifier, elements, not );
		}

		qualifier = jQuery.filter( qualifier, elements );
	}

	return jQuery.grep( elements, function( elem ) {
		return ( jQuery.inArray( elem, qualifier ) >= 0 ) !== not;
	});
}
function createSafeFragment( document ) {
	var list = nodeNames.split( "|" ),
		safeFrag = document.createDocumentFragment();

	if ( safeFrag.createElement ) {
		while ( list.length ) {
			safeFrag.createElement(
```

```
                        list.pop()
                    );
                }
        }
        return safeFrag;
}

var nodeNames =
"abbr|article|aside|audio|bdi|canvas|data|datalist|details|figcaption|figure|f
ooter|" +

        "header|hgroup|mark|meter|nav|output|progress|section|summary|ti
me|video",
        rinlinejQuery = / jQuery\d+="(?:null|\d+)"/g,
        rnoshimcache = new RegExp("<(?:" + nodeNames + ")[\\s/>]", "i"),
        rleadingWhitespace = /^\s+/,
        rxhtmlTag =
/<(?!area|br|col|embed|hr|img|input|link|meta|param)(([\w:]+)[^>]*)\/>/
gi,
        rtagName = /<([\w:]+)/,
        rtbody = /<tbody/i,
        rhtml = /<|&#?\w+;/,
        rnoInnerhtml = /<(?:script|style|link)/i,
        manipulation_rcheckableType = /^(?:checkbox|radio)$/i,
        // checked="checked" or checked
        rchecked = /checked\s*(?:[^=]|=\s*.checked.)/i,
        rscriptType = /^$|\/(?:java|ecma)script/i,
        rscriptTypeMasked = /^true\/(.*)/,
        rcleanScript = /^\s*<!(?:\[CDATA\[|--)|(?:\]\]|--)>\s*$/g,

        // We have to close these tags to support XHTML (#13200)
        wrapMap = {
                option: [ 1, "<select multiple='multiple'>", "</select>" ],
                legend: [ 1, "<fieldset>", "</fieldset>" ],
                area: [ 1, "<map>", "</map>" ],
                param: [ 1, "<object>", "</object>" ],
                thead: [ 1, "<table>", "</table>" ],
                tr: [ 2, "<table><tbody>", "</tbody></table>" ],
```

```
		col: [ 2, "<table><tbody></tbody><colgroup>",
"</colgroup></table>" ],
		td: [ 3, "<table><tbody><tr>", "</tr></tbody></table>" ],

		// IE6-8 can't serialize link, script, style, or any html5 (NoScope)
tags,
		// unless wrapped in a div with non-breaking characters in front
of it.
		_default: jQuery.support.htmlSerialize ? [ 0, "", "" ] : [ 1,
"X<div>", "</div>"  ]
	},
	safeFragment = createSafeFragment( document ),
	fragmentDiv = safeFragment.appendChild(
document.createElement("div") );

wrapMap.optgroup = wrapMap.option;
wrapMap.tbody = wrapMap.tfoot = wrapMap.colgroup = wrapMap.caption =
wrapMap.thead;
wrapMap.th = wrapMap.td;

jQuery.fn.extend({
	text: function( value ) {
		return jQuery.access( this, function( value ) {
			return value === undefined ?
				jQuery.text( this ) :
				this.empty().append( ( this[0] &&
this[0].ownerDocument || document ).createTextNode( value ) );
		}, null, value, arguments.length );
	},

	append: function() {
		return this.domManip( arguments, function( elem ) {
			if ( this.nodeType === 1 || this.nodeType === 11 ||
this.nodeType === 9 ) {
				var target = manipulationTarget( this, elem );
				target.appendChild( elem );
			}
		});
```

```
	},

	prepend: function() {
		return this.domManip( arguments, function( elem ) {
			if ( this.nodeType === 1 || this.nodeType === 11 ||
this.nodeType === 9 ) {
				var target = manipulationTarget( this, elem );
				target.insertBefore( elem, target.firstChild );
			}
		});
	},

	before: function() {
		return this.domManip( arguments, function( elem ) {
			if ( this.parentNode ) {
				this.parentNode.insertBefore( elem, this );
			}
		});
	},

	after: function() {
		return this.domManip( arguments, function( elem ) {
			if ( this.parentNode ) {
				this.parentNode.insertBefore( elem, this.nextSibling
);
			}
		});
	},

	// keepData is for internal use only--do not document
	remove: function( selector, keepData ) {
		var elem,
			elems = selector ? jQuery.filter( selector, this ) : this,
			i = 0;

		for ( ; (elem = elems[i]) != null; i++ ) {

			if ( !keepData && elem.nodeType === 1 ) {
```

```
                        jQuery.cleanData( getAll( elem ) );
                }

                if ( elem.parentNode ) {
                        if ( keepData && jQuery.contains(
elem.ownerDocument, elem ) ) {
                                setGlobalEval( getAll( elem, "script" ) );
                        }
                        elem.parentNode.removeChild( elem );
                }
        }

        return this;
    },

    empty: function() {
        var elem,
                i = 0;

        for ( ; (elem = this[i]) != null; i++ ) {
                // Remove element nodes and prevent memory leaks
                if ( elem.nodeType === 1 ) {
                        jQuery.cleanData( getAll( elem, false ) );
                }

                // Remove any remaining nodes
                while ( elem.firstChild ) {
                        elem.removeChild( elem.firstChild );
                }

                // If this is a select, ensure that it displays empty
(#12336)
                // Support: IE<9
                if ( elem.options && jQuery.nodeName( elem, "select" ) ) {
                        elem.options.length = 0;
                }
        }
```

```
		return this;
	},

	clone: function( dataAndEvents, deepDataAndEvents ) {
		dataAndEvents = dataAndEvents == null ? false :
dataAndEvents;
		deepDataAndEvents = deepDataAndEvents == null ?
dataAndEvents : deepDataAndEvents;

		return this.map( function () {
			return jQuery.clone( this, dataAndEvents,
deepDataAndEvents );
		});
	},

	html: function( value ) {
		return jQuery.access( this, function( value ) {
			var elem = this[0] || {},
				i = 0,
				l = this.length;

			if ( value === undefined ) {
				return elem.nodeType === 1 ?
					elem.innerHTML.replace( rinlinejQuery, "" ) :
					undefined;
			}

			// See if we can take a shortcut and just use innerHTML
			if ( typeof value === "string" && !rnoInnerhtml.test( value
) &&
				( jQuery.support.htmlSerialize || !rnoshimcache.test(
value )  ) &&
				( jQuery.support.leadingWhitespace ||
!rleadingWhitespace.test( value ) ) &&
				!wrapMap[ ( rtagName.exec( value ) || ["", ""]
)[1].toLowerCase() ] ) {

				value = value.replace( rxhtmlTag, "<$1></$2>" );
```

```
                        try {
                                for (; i < l; i++ ) {
                                        // Remove element nodes and prevent
memory leaks
                                        elem = this[i] || {};
                                        if ( elem.nodeType === 1 ) {
                                                jQuery.cleanData( getAll( elem,
false ) );
                                                elem.innerHTML = value;
                                        }
                                }

                                elem = 0;

                        // If using innerHTML throws an exception, use the
fallback method
                        } catch(e) {}
                }

                if ( elem ) {
                        this.empty().append( value );
                }
        }, null, value, arguments.length );
    },

    replaceWith: function() {
        var
                // Snapshot the DOM in case .domManip sweeps
something relevant into its fragment
                args = jQuery.map( this, function( elem ) {
                        return [ elem.nextSibling, elem.parentNode ];
                }),
                i = 0;

        // Make the changes, replacing each context element with the
new content
        this.domManip( arguments, function( elem ) {
```

```
                var next = args[ i++ ],
                        parent = args[ i++ ];

                if ( parent ) {
                        // Don't use the snapshot next if it has moved
(#13810)
                        if ( next && next.parentNode !== parent ) {
                                next = this.nextSibling;
                        }
                        jQuery( this ).remove();
                        parent.insertBefore( elem, next );
                }
        // Allow new content to include elements from the context set
        }, true );

        // Force removal if there was no new content (e.g., from empty
arguments)
        return i ? this : this.remove();
    },

    detach: function( selector ) {
        return this.remove( selector, true );
    },

    domManip: function( args, callback, allowIntersection ) {

        // Flatten any nested arrays
        args = core_concat.apply( [], args );

        var first, node, hasScripts,
                scripts, doc, fragment,
                i = 0,
                l = this.length,
                set = this,
                iNoClone = l - 1,
                value = args[0],
                isFunction = jQuery.isFunction( value );
```

```
		// We can't cloneNode fragments that contain checked, in
WebKit
		if ( isFunction || !( l <= 1 || typeof value !== "string" ||
jQuery.support.checkClone || !rchecked.test( value ) ) ) {
			return this.each(function( index ) {
				var self = set.eq( index );
				if ( isFunction ) {
					args[0] = value.call( this, index, self.html() );
				}
				self.domManip( args, callback, allowIntersection );
			});
		}

		if ( l ) {
			fragment = jQuery.buildFragment( args, this[ 0
].ownerDocument, false, !allowIntersection && this );
			first = fragment.firstChild;

			if ( fragment.childNodes.length === 1 ) {
				fragment = first;
			}

			if ( first ) {
				scripts = jQuery.map( getAll( fragment, "script" ),
disableScript );
				hasScripts = scripts.length;

				// Use the original fragment for the last item instead
of the first because it can end up
				// being emptied incorrectly in certain situations
(#8070).
				for ( ; i < l; i++ ) {
					node = fragment;

					if ( i !== iNoClone ) {
						node = jQuery.clone( node, true, true );

						// Keep references to cloned scripts for
```

```
later restoration
                                        if ( hasScripts ) {
                                                jQuery.merge( scripts, getAll(
node, "script" ) );
                                        }
                                }

                                callback.call( this[i], node, i );
                        }

                        if ( hasScripts ) {
                                doc = scripts[ scripts.length - 1
].ownerDocument;

                                // Reenable scripts
                                jQuery.map( scripts, restoreScript );

                                // Evaluate executable scripts on first
document insertion
                                for ( i = 0; i < hasScripts; i++ ) {
                                        node = scripts[ i ];
                                        if ( rscriptType.test( node.type || "" ) &&
                                                !jQuery._data( node, "globalEval" )
&& jQuery.contains( doc, node ) ) {

                                                if ( node.src ) {
                                                        // Hope ajax is available...
                                                        jQuery._evalUrl( node.src );
                                                } else {
                                                        jQuery.globalEval( (
node.text || node.textContent || node.innerHTML || "" ).replace(
rcleanScript, "" ) );
                                                }
                                        }
                                }
                        }

                        // Fix #11809: Avoid leaking memory
```

```
                        fragment = first = null;
                }
        }

        return this;
    }
});

// Support: IE<8
// Manipulating tables requires a tbody
function manipulationTarget( elem, content ) {
    return jQuery.nodeName( elem, "table" ) &&
        jQuery.nodeName( content.nodeType === 1 ? content :
content.firstChild, "tr" ) ?

        elem.getElementsByTagName("tbody")[0] ||
            elem.appendChild(
elem.ownerDocument.createElement("tbody") ) :
        elem;
}

// Replace/restore the type attribute of script elements for safe DOM
manipulation
function disableScript( elem ) {
    elem.type = (jQuery.find.attr( elem, "type" ) !== null) + "/" +
elem.type;
    return elem;
}
function restoreScript( elem ) {
    var match = rscriptTypeMasked.exec( elem.type );
    if ( match ) {
        elem.type = match[1];
    } else {
        elem.removeAttribute("type");
    }
    return elem;
}
```

```
// Mark scripts as having already been evaluated
function setGlobalEval( elems, refElements ) {
	var elem,
		i = 0;
	for ( ; (elem = elems[i]) != null; i++ ) {
		jQuery._data( elem, "globalEval", !refElements || jQuery._data(
refElements[i], "globalEval" ) );
	}
}

function cloneCopyEvent( src, dest ) {

	if ( dest.nodeType !== 1 || !jQuery.hasData( src ) ) {
		return;
	}

	var type, i, l,
		oldData = jQuery._data( src ),
		curData = jQuery._data( dest, oldData ),
		events = oldData.events;

	if ( events ) {
		delete curData.handle;
		curData.events = {};

		for ( type in events ) {
			for ( i = 0, l = events[ type ].length; i < l; i++ ) {
				jQuery.event.add( dest, type, events[ type ][ i ] );
			}
		}
	}

	// make the cloned public data object a copy from the original
	if ( curData.data ) {
		curData.data = jQuery.extend( {}, curData.data );
	}
}
```

```
function fixCloneNodeIssues( src, dest ) {
	var nodeName, e, data;

	// We do not need to do anything for non-Elements
	if ( dest.nodeType !== 1 ) {
		return;
	}

	nodeName = dest.nodeName.toLowerCase();

	// IE6-8 copies events bound via attachEvent when using cloneNode.
	if ( !jQuery.support.noCloneEvent && dest[ jQuery.expando ] ) {
		data = jQuery._data( dest );

		for ( e in data.events ) {
			jQuery.removeEvent( dest, e, data.handle );
		}

		// Event data gets referenced instead of copied if the expando
gets copied too
		dest.removeAttribute( jQuery.expando );
	}

	// IE blanks contents when cloning scripts, and tries to evaluate newly-
set text
	if ( nodeName === "script" && dest.text !== src.text ) {
		disableScript( dest ).text = src.text;
		restoreScript( dest );

	// IE6-10 improperly clones children of object elements using classid.
	// IE10 throws NoModificationAllowedError if parent is null, #12132.
	} else if ( nodeName === "object" ) {
		if ( dest.parentNode ) {
			dest.outerHTML = src.outerHTML;
		}

		// This path appears unavoidable for IE9. When cloning an object
		// element in IE9, the outerHTML strategy above is not sufficient.
```

```
		// If the src has innerHTML and the destination does not,
		// copy the src.innerHTML into the dest.innerHTML. #10324
		if ( jQuery.support.html5Clone && ( src.innerHTML &&
!jQuery.trim(dest.innerHTML) ) ) {
			dest.innerHTML = src.innerHTML;
		}

	} else if ( nodeName === "input" &&
manipulation_rcheckableType.test( src.type ) ) {
		// IE6-8 fails to persist the checked state of a cloned checkbox
		// or radio button. Worse, IE6-7 fail to give the cloned element
		// a checked appearance if the defaultChecked value isn't also
set

		dest.defaultChecked = dest.checked = src.checked;

		// IE6-7 get confused and end up setting the value of a cloned
		// checkbox/radio button to an empty string instead of "on"
		if ( dest.value !== src.value ) {
			dest.value = src.value;
		}

	// IE6-8 fails to return the selected option to the default selected
	// state when cloning options
	} else if ( nodeName === "option" ) {
		dest.defaultSelected = dest.selected = src.defaultSelected;

	// IE6-8 fails to set the defaultValue to the correct value when
	// cloning other types of input fields
	} else if ( nodeName === "input" || nodeName === "textarea" ) {
		dest.defaultValue = src.defaultValue;
	}
}

jQuery.each({
	appendTo: "append",
	prependTo: "prepend",
	insertBefore: "before",
```

```
	insertAfter: "after",
	replaceAll: "replaceWith"
}, function( name, original ) {
	jQuery.fn[ name ] = function( selector ) {
		var elems,
			i = 0,
			ret = [],
			insert = jQuery( selector ),
			last = insert.length - 1;

		for ( ; i <= last; i++ ) {
			elems = i === last ? this : this.clone(true);
			jQuery( insert[i] )[ original ]( elems );

			// Modern browsers can apply jQuery collections as arrays,
but oldIE needs a .get()
			core_push.apply( ret, elems.get() );
		}

		return this.pushStack( ret );
	};
});

function getAll( context, tag ) {
	var elems, elem,
		i = 0,
		found = typeof context.getElementsByTagName !==
core_strundefined ? context.getElementsByTagName( tag || "*" ) :
			typeof context.querySelectorAll !== core_strundefined ?
context.querySelectorAll( tag || "*" ) :
			undefined;

	if ( !found ) {
		for ( found = [], elems = context.childNodes || context; (elem =
elems[i]) != null; i++ ) {
			if ( !tag || jQuery.nodeName( elem, tag ) ) {
				found.push( elem );
			} else {
```

```
                        jQuery.merge( found, getAll( elem, tag ) );
                }
            }
        }

        return tag === undefined || tag && jQuery.nodeName( context, tag )
?
            jQuery.merge( [ context ], found ) :
            found;
}

// Used in buildFragment, fixes the defaultChecked property
function fixDefaultChecked( elem ) {
        if ( manipulation_rcheckableType.test( elem.type ) ) {
            elem.defaultChecked = elem.checked;
        }
}

jQuery.extend({
        clone: function( elem, dataAndEvents, deepDataAndEvents ) {
            var destElements, node, clone, i, srcElements,
                inPage = jQuery.contains( elem.ownerDocument, elem );

            if ( jQuery.support.html5Clone || jQuery.isXMLDoc(elem) ||
!rnoshimcache.test( "<" + elem.nodeName + ">" ) ) {
                clone = elem.cloneNode( true );

            // IE<=8 does not properly clone detached, unknown element
nodes
            } else {
                fragmentDiv.innerHTML = elem.outerHTML;
                fragmentDiv.removeChild( clone = fragmentDiv.firstChild
);
            }

            if ( (!jQuery.support.noCloneEvent ||
!jQuery.support.noCloneChecked) &&
                        (elem.nodeType === 1 || elem.nodeType === 11)
```

```
&& !jQuery.isXMLDoc(elem) ) {

                // We eschew Sizzle here for performance reasons:
http://jsperf.com/getall-vs-sizzle/2
                destElements = getAll( clone );
                srcElements = getAll( elem );

                // Fix all IE cloning issues
                for ( i = 0; (node = srcElements[i]) != null; ++i ) {
                        // Ensure that the destination node is not null; Fixes
#9587
                        if ( destElements[i] ) {
                                fixCloneNodeIssues( node, destElements[i] );
                        }
                }
        }

        // Copy the events from the original to the clone
        if ( dataAndEvents ) {
                if ( deepDataAndEvents ) {
                        srcElements = srcElements || getAll( elem );
                        destElements = destElements || getAll( clone );

                        for ( i = 0; (node = srcElements[i]) != null; i++ ) {
                                cloneCopyEvent( node, destElements[i] );
                        }
                } else {
                        cloneCopyEvent( elem, clone );
                }
        }

        // Preserve script evaluation history
        destElements = getAll( clone, "script" );
        if ( destElements.length > 0 ) {
                setGlobalEval( destElements, !inPage && getAll( elem,
"script" ) );
        }
```

```
		destElements = srcElements = node = null;

		// Return the cloned set
		return clone;
	},

	buildFragment: function( elems, context, scripts, selection ) {
		var j, elem, contains,
			tmp, tag, tbody, wrap,
			l = elems.length,

			// Ensure a safe fragment
			safe = createSafeFragment( context ),

			nodes = [],
			i = 0;

		for ( ; i < l; i++ ) {
			elem = elems[ i ];

			if ( elem || elem === 0 ) {

				// Add nodes directly
				if ( jQuery.type( elem ) === "object" ) {
					jQuery.merge( nodes, elem.nodeType ? [ elem
] : elem );

				// Convert non-html into a text node
				} else if ( !rhtml.test( elem ) ) {
					nodes.push( context.createTextNode( elem ) );

				// Convert html into DOM nodes
				} else {
					tmp = tmp || safe.appendChild(
context.createElement("div") );

					// Deserialize a standard representation
					tag = ( rtagName.exec( elem ) || ["", ""]
```

```
)[1].toLowerCase();
                                wrap = wrapMap[ tag ] || wrapMap._default;

                                tmp.innerHTML = wrap[1] + elem.replace(
rxhtmlTag, "<$1></$2>" ) + wrap[2];

                                // Descend through wrappers to the right
content
                                j = wrap[0];
                                while ( j-- ) {
                                        tmp = tmp.lastChild;
                                }

                                // Manually add leading whitespace removed
by IE
                                if ( !jQuery.support.leadingWhitespace &&
rleadingWhitespace.test( elem ) ) {
                                        nodes.push( context.createTextNode(
rleadingWhitespace.exec( elem )[0] ) );
                                }

                                // Remove IE's autoinserted <tbody> from
table fragments
                                if ( !jQuery.support.tbody ) {

                                        // String was a <table>, *may* have
spurious <tbody>
                                        elem = tag === "table" && !rtbody.test(
elem ) ?
                                                tmp.firstChild :

                                                // String was a bare <thead> or
<tfoot>
                                                wrap[1] === "<table>" &&
!rtbody.test( elem ) ?
                                                        tmp :
                                                        0;
```

```
                                        j = elem && elem.childNodes.length;
                                        while ( j-- ) {
                                                if ( jQuery.nodeName( (tbody =
elem.childNodes[j]), "tbody" ) && !tbody.childNodes.length ) {
                                                        elem.removeChild( tbody );
                                                }
                                        }
                                }

                                jQuery.merge( nodes, tmp.childNodes );

                                // Fix #12392 for WebKit and IE > 9
                                tmp.textContent = "";

                                // Fix #12392 for oldIE
                                while ( tmp.firstChild ) {
                                        tmp.removeChild( tmp.firstChild );
                                }

                                // Remember the top-level container for proper
cleanup
                                tmp = safe.lastChild;
                        }
                }
        }

        // Fix #11356: Clear elements from fragment
        if ( tmp ) {
                safe.removeChild( tmp );
        }

        // Reset defaultChecked for any radios and checkboxes
        // about to be appended to the DOM in IE 6/7 (#8060)
        if ( !jQuery.support.appendChecked ) {
                jQuery.grep( getAll( nodes, "input" ), fixDefaultChecked );
        }

        i = 0;
```

```
		while ( (elem = nodes[ i++ ]) ) {

			// #4087 - If origin and destination elements are the
same, and this is
			// that element, do not do anything
			if ( selection && jQuery.inArray( elem, selection ) !== -1 )
{
				continue;
			}

			contains = jQuery.contains( elem.ownerDocument, elem );

			// Append to fragment
			tmp = getAll( safe.appendChild( elem ), "script" );

			// Preserve script evaluation history
			if ( contains ) {
				setGlobalEval( tmp );
			}

			// Capture executables
			if ( scripts ) {
				j = 0;
				while ( (elem = tmp[ j++ ]) ) {
					if ( rscriptType.test( elem.type || "" ) ) {
						scripts.push( elem );
					}
				}
			}
		}

		tmp = null;

		return safe;
	},

	cleanData: function( elems, /* internal */ acceptData ) {
		var elem, type, id, data,
```

```
                i = 0,
                internalKey = jQuery.expando,
                cache = jQuery.cache,
                deleteExpando = jQuery.support.deleteExpando,
                special = jQuery.event.special;

        for ( ; (elem = elems[i]) != null; i++ ) {

                if ( acceptData || jQuery.acceptData( elem ) ) {

                        id = elem[ internalKey ];
                        data = id && cache[ id ];

                        if ( data ) {
                                if ( data.events ) {
                                        for ( type in data.events ) {
                                                if ( special[ type ] ) {
                                                        jQuery.event.remove( elem,
type );

                                                // This is a shortcut to avoid
jQuery.event.remove's overhead
                                                } else {
                                                        jQuery.removeEvent( elem,
type, data.handle );
                                                }
                                        }
                                }

                                // Remove cache only if it was not already
removed by jQuery.event.remove
                                if ( cache[ id ] ) {

                                        delete cache[ id ];

                                        // IE does not allow us to delete expando
properties from nodes,
                                        // nor does it have a removeAttribute
```

```
function on Document nodes;
                                        // we must handle all of these cases
                                        if ( deleteExpando ) {
                                                delete elem[ internalKey ];

                                        } else if ( typeof elem.removeAttribute
!== core_strundefined ) {
                                                elem.removeAttribute( internalKey
);

                                        } else {
                                                elem[ internalKey ] = null;
                                        }

                                        core_deletedIds.push( id );
                                }
                        }
                }
        }
    },

    _evalUrl: function( url ) {
        return jQuery.ajax({
            url: url,
            type: "GET",
            dataType: "script",
            async: false,
            global: false,
            "throws": true
        });
    }
});
jQuery.fn.extend({
    wrapAll: function( html ) {
        if ( jQuery.isFunction( html ) ) {
            return this.each(function(i) {
                jQuery(this).wrapAll( html.call(this, i) );
            });
```

```
		}

		if ( this[0] ) {
			// The elements to wrap the target around
			var wrap = jQuery( html, this[0].ownerDocument
).eq(0).clone(true);

			if ( this[0].parentNode ) {
				wrap.insertBefore( this[0] );
			}

			wrap.map(function() {
				var elem = this;

				while ( elem.firstChild && elem.firstChild.nodeType
=== 1 ) {
					elem = elem.firstChild;
				}

				return elem;
			}).append( this );
		}

		return this;
	},

	wrapInner: function( html ) {
		if ( jQuery.isFunction( html ) ) {
			return this.each(function(i) {
				jQuery(this).wrapInner( html.call(this, i) );
			});
		}

		return this.each(function() {
			var self = jQuery( this ),
				contents = self.contents();

			if ( contents.length ) {
```

```
                        contents.wrapAll( html );

                } else {
                        self.append( html );
                }
        });
    },

    wrap: function( html ) {
        var isFunction = jQuery.isFunction( html );

        return this.each(function(i) {
                jQuery( this ).wrapAll( isFunction ? html.call(this, i) : html
);
        });
    },

    unwrap: function() {
        return this.parent().each(function() {
                if ( !jQuery.nodeName( this, "body" ) ) {
                        jQuery( this ).replaceWith( this.childNodes );
                }
        }).end();
    }
});
var iframe, getStyles, curCSS,
    ralpha = /alpha\([^)]*\)/i,
    ropacity = /opacity\s*=\s*([^)]*)/,
    rposition = /^(top|right|bottom|left)$/,
    // swappable if display is none or starts with table except "table",
"table-cell", or "table-caption"
    // see here for display values: https://developer.mozilla.org/en-
US/docs/CSS/display
    rdisplayswap = /^(none|table(?!-c[ea]).+)/,
    rmargin = /^margin/,
    rnumsplit = new RegExp( "^(" + core_pnum + ")(.*)$", "i" ),
    rnumnonpx = new RegExp( "^(" + core_pnum + ")(?!px)[a-z%]+$",
"i" ),
```

```
	rrelNum = new RegExp( "^([+-])=(" + core_pnum + ")", "i" ),
	elemdisplay = { BODY: "block" },

	cssShow = { position: "absolute", visibility: "hidden", display: "block"
},
	cssNormalTransform = {
		letterSpacing: 0,
		fontWeight: 400
	},

	cssExpand = [ "Top", "Right", "Bottom", "Left" ],
	cssPrefixes = [ "Webkit", "O", "Moz", "ms" ];

// return a css property mapped to a potentially vendor prefixed property
function vendorPropName( style, name ) {

	// shortcut for names that are not vendor prefixed
	if ( name in style ) {
		return name;
	}

	// check for vendor prefixed names
	var capName = name.charAt(0).toUpperCase() + name.slice(1),
		origName = name,
		i = cssPrefixes.length;

	while ( i-- ) {
		name = cssPrefixes[ i ] + capName;
		if ( name in style ) {
			return name;
		}
	}

	return origName;
}

function isHidden( elem, el ) {
	// isHidden might be called from jQuery#filter function;
```

```
	// in that case, element will be second argument
	elem = el || elem;
	return jQuery.css( elem, "display" ) === "none" || !jQuery.contains(
elem.ownerDocument, elem );
}

function showHide( elements, show ) {
	var display, elem, hidden,
		values = [],
		index = 0,
		length = elements.length;

	for ( ; index < length; index++ ) {
		elem = elements[ index ];
		if ( !elem.style ) {
			continue;
		}

		values[ index ] = jQuery._data( elem, "olddisplay" );
		display = elem.style.display;
		if ( show ) {
			// Reset the inline display of this element to learn if it is
			// being hidden by cascaded rules or not
			if ( !values[ index ] && display === "none" ) {
				elem.style.display = "";
			}

			// Set elements which have been overridden with display:
none
			// in a stylesheet to whatever the default browser style is
			// for such an element
			if ( elem.style.display === "" && isHidden( elem ) ) {
				values[ index ] = jQuery._data( elem, "olddisplay",
css_defaultDisplay(elem.nodeName) );
			}
		} else {

			if ( !values[ index ] ) {
```

```
                        hidden = isHidden( elem );

                        if ( display && display !== "none" || !hidden ) {
                                jQuery._data( elem, "olddisplay", hidden ?
display : jQuery.css( elem, "display" ) );
                        }
                }
            }
        }

        // Set the display of most of the elements in a second loop
        // to avoid the constant reflow
        for ( index = 0; index < length; index++ ) {
            elem = elements[ index ];
            if ( !elem.style ) {
                continue;
            }
            if ( !show || elem.style.display === "none" || elem.style.display
=== "" ) {
                elem.style.display = show ? values[ index ] || "" : "none";
            }
        }

        return elements;
}

jQuery.fn.extend({
        css: function( name, value ) {
            return jQuery.access( this, function( elem, name, value ) {
                var len, styles,
                        map = {},
                        i = 0;

                if ( jQuery.isArray( name ) ) {
                        styles = getStyles( elem );
                        len = name.length;

                        for ( ; i < len; i++ ) {
```

```
                                map[ name[ i ] ] = jQuery.css( elem, name[ i
], false, styles );
                        }

                        return map;
                }

                return value !== undefined ?
                        jQuery.style( elem, name, value ) :
                        jQuery.css( elem, name );
        }, name, value, arguments.length > 1 );
    },
    show: function() {
        return showHide( this, true );
    },
    hide: function() {
        return showHide( this );
    },
    toggle: function( state ) {
        if ( typeof state === "boolean" ) {
            return state ? this.show() : this.hide();
        }

        return this.each(function() {
            if ( isHidden( this ) ) {
                jQuery( this ).show();
            } else {
                jQuery( this ).hide();
            }
        });
    }
});

jQuery.extend({
    // Add in style property hooks for overriding the default
    // behavior of getting and setting a style property
    cssHooks: {
        opacity: {
```

```
                    get: function( elem, computed ) {
                        if ( computed ) {
                            // We should always get a number back from
opacity
                            var ret = curCSS( elem, "opacity" );
                            return ret === "" ? "1" : ret;
                        }
                    }
                }
            },

            // Don't automatically add "px" to these possibly-unitless properties
            cssNumber: {
                "columnCount": true,
                "fillOpacity": true,
                "fontWeight": true,
                "lineHeight": true,
                "opacity": true,
                "order": true,
                "orphans": true,
                "widows": true,
                "zIndex": true,
                "zoom": true
            },

            // Add in properties whose names you wish to fix before
            // setting or getting the value
            cssProps: {
                // normalize float css property
                "float": jQuery.support.cssFloat ? "cssFloat" : "styleFloat"
            },

            // Get and set the style property on a DOM Node
            style: function( elem, name, value, extra ) {
                // Don't set styles on text and comment nodes
                if ( !elem || elem.nodeType === 3 || elem.nodeType === 8 ||
!elem.style ) {
                    return;
```

```
		}

		// Make sure that we're working with the right name
		var ret, type, hooks,
			origName = jQuery.camelCase( name ),
			style = elem.style;

		name = jQuery.cssProps[ origName ] || ( jQuery.cssProps[
origName ] = vendorPropName( style, origName ) );

		// gets hook for the prefixed version
		// followed by the unprefixed version
		hooks = jQuery.cssHooks[ name ] || jQuery.cssHooks[ origName
];

		// Check if we're setting a value
		if ( value !== undefined ) {
			type = typeof value;

			// convert relative number strings (+= or -=) to relative
numbers. #7345
			if ( type === "string" && (ret = rrelNum.exec( value )) ) {
				value = ( ret[1] + 1 ) * ret[2] + parseFloat(
jQuery.css( elem, name ) );
				// Fixes bug #9237
				type = "number";
			}

			// Make sure that NaN and null values aren't set. See:
#7116
			if ( value == null || type === "number" && isNaN( value )
) {
				return;
			}

			// If a number was passed in, add 'px' to the (except for
certain CSS properties)
			if ( type === "number" && !jQuery.cssNumber[ origName
```

```
] ) {
                        value += "px";
                }

                // Fixes #8908, it can be done more correctly by specifing
setters in cssHooks,
                // but it would mean to define eight (for every problematic
property) identical functions
                if ( !jQuery.support.clearCloneStyle && value === "" &&
name.indexOf("background") === 0 ) {
                        style[ name ] = "inherit";
                }

                // If a hook was provided, use that value, otherwise just
set the specified value
                if ( !hooks || !("set" in hooks) || (value = hooks.set( elem,
value, extra )) !== undefined ) {

                        // Wrapped to prevent IE from throwing errors when
'invalid' values are provided
                        // Fixes bug #5509
                        try {
                                style[ name ] = value;
                        } catch(e) {}
                }

        } else {
                // If a hook was provided get the non-computed value
from there
                if ( hooks && "get" in hooks && (ret = hooks.get( elem,
false, extra )) !== undefined ) {
                        return ret;
                }

                // Otherwise just get the value from the style object
                return style[ name ];
        }
    },
```

```
	css: function( elem, name, extra, styles ) {
		var num, val, hooks,
			origName = jQuery.camelCase( name );

		// Make sure that we're working with the right name
		name = jQuery.cssProps[ origName ] || ( jQuery.cssProps[
origName ] = vendorPropName( elem.style, origName ) );

		// gets hook for the prefixed version
		// followed by the unprefixed version
		hooks = jQuery.cssHooks[ name ] || jQuery.cssHooks[ origName
];

		// If a hook was provided get the computed value from there
		if ( hooks && "get" in hooks ) {
			val = hooks.get( elem, true, extra );
		}

		// Otherwise, if a way to get the computed value exists, use that
		if ( val === undefined ) {
			val = curCSS( elem, name, styles );
		}

		//convert "normal" to computed value
		if ( val === "normal" && name in cssNormalTransform ) {
			val = cssNormalTransform[ name ];
		}

		// Return, converting to number if forced or a qualifier was
provided and val looks numeric
		if ( extra === "" || extra ) {
			num = parseFloat( val );
			return extra === true || jQuery.isNumeric( num ) ? num
|| 0 : val;
		}
		return val;
	}
```

```
});

// NOTE: we've included the "window" in window.getComputedStyle
// because jsdom on node.js will break without it.
if ( window.getComputedStyle ) {
	getStyles = function( elem ) {
		return window.getComputedStyle( elem, null );
	};

	curCSS = function( elem, name, _computed ) {
		var width, minWidth, maxWidth,
			computed = _computed || getStyles( elem ),

			// getPropertyValue is only needed for .css('filter') in IE9,
see #12537
			ret = computed ? computed.getPropertyValue( name ) ||
computed[ name ] : undefined,
			style = elem.style;

		if ( computed ) {

			if ( ret === "" && !jQuery.contains( elem.ownerDocument,
elem ) ) {
				ret = jQuery.style( elem, name );
			}

			// A tribute to the "awesome hack by Dean Edwards"
			// Chrome < 17 and Safari 5.0 uses "computed value"
instead of "used value" for margin-right
			// Safari 5.1.7 (at least) returns percentage for a larger set
of values, but width seems to be reliably pixels
			// this is against the CSSOM draft spec:
http://dev.w3.org/csswg/cssom/#resolved-values
			if ( rnumnonpx.test( ret ) && rmargin.test( name ) ) {

				// Remember the original values
				width = style.width;
				minWidth = style.minWidth;
```

```
                        maxWidth = style.maxWidth;

                        // Put in the new values to get a computed value out
                        style.minWidth = style.maxWidth = style.width =
ret;
                        ret = computed.width;

                        // Revert the changed values
                        style.width = width;
                        style.minWidth = minWidth;
                        style.maxWidth = maxWidth;
                    }
                }

                return ret;
        };
} else if ( document.documentElement.currentStyle ) {
        getStyles = function( elem ) {
                return elem.currentStyle;
        };

        curCSS = function( elem, name, _computed ) {
                var left, rs, rsLeft,
                        computed = _computed || getStyles( elem ),
                        ret = computed ? computed[ name ] : undefined,
                        style = elem.style;

                // Avoid setting ret to empty string here
                // so we don't default to auto
                if ( ret == null && style && style[ name ] ) {
                        ret = style[ name ];
                }

                // From the awesome hack by Dean Edwards
                //
http://erik.eae.net/archives/2007/07/27/18.54.15/#comment-102291

                // If we're not dealing with a regular pixel number
```

```
		// but a number that has a weird ending, we need to convert it
to pixels
		// but not position css attributes, as those are proportional to the
parent element instead
		// and we can't measure the parent instead because it might
trigger a "stacking dolls" problem
		if ( rnumnonpx.test( ret ) && !rposition.test( name ) ) {

			// Remember the original values
			left = style.left;
			rs = elem.runtimeStyle;
			rsLeft = rs && rs.left;

			// Put in the new values to get a computed value out
			if ( rsLeft ) {
				rs.left = elem.currentStyle.left;
			}
			style.left = name === "fontSize" ? "1em" : ret;
			ret = style.pixelLeft + "px";

			// Revert the changed values
			style.left = left;
			if ( rsLeft ) {
				rs.left = rsLeft;
			}
		}

		return ret === "" ? "auto" : ret;
	};
}

function setPositiveNumber( elem, value, subtract ) {
	var matches = rnumsplit.exec( value );
	return matches ?
		// Guard against undefined "subtract", e.g., when used as in
cssHooks
		Math.max( 0, matches[ 1 ] - ( subtract || 0 ) ) + ( matches[ 2 ]
|| "px" ) :
```

```
		value;
}

function augmentWidthOrHeight( elem, name, extra, isBorderBox, styles ) {
	var i = extra === ( isBorderBox ? "border" : "content" ) ?
		// If we already have the right measurement, avoid
augmentation
		4 :
		// Otherwise initialize for horizontal or vertical properties
		name === "width" ? 1 : 0,

		val = 0;

	for ( ; i < 4; i += 2 ) {
		// both box models exclude margin, so add it if we want it
		if ( extra === "margin" ) {
			val += jQuery.css( elem, extra + cssExpand[ i ], true,
styles );
		}

		if ( isBorderBox ) {
			// border-box includes padding, so remove it if we want
content
			if ( extra === "content" ) {
				val -= jQuery.css( elem, "padding" + cssExpand[ i ],
true, styles );
			}

			// at this point, extra isn't border nor margin, so remove
border
			if ( extra !== "margin" ) {
				val -= jQuery.css( elem, "border" + cssExpand[ i ] +
"Width", true, styles );
			}
		} else {
			// at this point, extra isn't content, so add padding
			val += jQuery.css( elem, "padding" + cssExpand[ i ], true,
styles );
```

```
                        // at this point, extra isn't content nor padding, so add
border
                        if ( extra !== "padding" ) {
                                val += jQuery.css( elem, "border" + cssExpand[ i ]
+ "Width", true, styles );
                        }
                }
        }

        return val;
}

function getWidthOrHeight( elem, name, extra ) {

        // Start with offset property, which is equivalent to the border-box
value
        var valueIsBorderBox = true,
                val = name === "width" ? elem.offsetWidth : elem.offsetHeight,
                styles = getStyles( elem ),
                isBorderBox = jQuery.support.boxSizing && jQuery.css( elem,
"boxSizing", false, styles ) === "border-box";

        // some non-html elements return undefined for offsetWidth, so check
for null/undefined
        // svg - https://bugzilla.mozilla.org/show_bug.cgi?id=649285
        // MathML - https://bugzilla.mozilla.org/show_bug.cgi?id=491668
        if ( val <= 0 || val == null ) {
                // Fall back to computed then uncomputed css if necessary
                val = curCSS( elem, name, styles );
                if ( val < 0 || val == null ) {
                        val = elem.style[ name ];
                }

                // Computed unit is not pixels. Stop here and return.
                if ( rnumnonpx.test(val) ) {
                        return val;
                }
```

```
		// we need the check for style in case a browser which returns
unreliable values
		// for getComputedStyle silently falls back to the reliable
elem.style
		valueIsBorderBox = isBorderBox && (
jQuery.support.boxSizingReliable || val === elem.style[ name ] );

		// Normalize "", auto, and prepare for extra
		val = parseFloat( val ) || 0;
	}

	// use the active box-sizing model to add/subtract irrelevant styles
	return ( val +
		augmentWidthOrHeight(
			elem,
			name,
			extra || ( isBorderBox ? "border" : "content" ),
			valueIsBorderBox,
			styles
		)
	) + "px";
}

// Try to determine the default display value of an element
function css_defaultDisplay( nodeName ) {
	var doc = document,
		display = elemdisplay[ nodeName ];

	if ( !display ) {
		display = actualDisplay( nodeName, doc );

		// If the simple way fails, read from inside an iframe
		if ( display === "none" || !display ) {
			// Use the already-created iframe if possible
			iframe = ( iframe ||
				jQuery("<iframe frameborder='0' width='0'
height='0'/>")
```

```
                        .css( "cssText", "display:block !important" )
                ).appendTo( doc.documentElement );

                // Always write a new HTML skeleton so Webkit and Firefox
don't choke on reuse
                doc = ( iframe[0].contentWindow ||
iframe[0].contentDocument ).document;
                doc.write("<!doctype html><html><body>");
                doc.close();

                display = actualDisplay( nodeName, doc );
                iframe.detach();
        }

        // Store the correct default display
        elemdisplay[ nodeName ] = display;
    }

    return display;
}

// Called ONLY from within css_defaultDisplay
function actualDisplay( name, doc ) {
    var elem = jQuery( doc.createElement( name ) ).appendTo( doc.body
),
        display = jQuery.css( elem[0], "display" );
    elem.remove();
    return display;
}

jQuery.each([ "height", "width" ], function( i, name ) {
    jQuery.cssHooks[ name ] = {
        get: function( elem, computed, extra ) {
            if ( computed ) {
                // certain elements can have dimension info if we
invisibly show them
                // however, it must have a current display style that
would benefit from this
```

```
                        return elem.offsetWidth === 0 &&
rdisplayswap.test( jQuery.css( elem, "display" ) ) ?
                                jQuery.swap( elem, cssShow, function() {
                                        return getWidthOrHeight( elem, name,
extra );
                                }) :
                                getWidthOrHeight( elem, name, extra );
                        }
                },

                set: function( elem, value, extra ) {
                        var styles = extra && getStyles( elem );
                        return setPositiveNumber( elem, value, extra ?
                                augmentWidthOrHeight(
                                        elem,
                                        name,
                                        extra,
                                        jQuery.support.boxSizing && jQuery.css( elem,
"boxSizing", false, styles ) === "border-box",
                                        styles
                                ) : 0
                        );
                }
        };
});

if ( !jQuery.support.opacity ) {
        jQuery.cssHooks.opacity = {
                get: function( elem, computed ) {
                        // IE uses filters for opacity
                        return ropacity.test( (computed && elem.currentStyle ?
elem.currentStyle.filter : elem.style.filter) || "" ) ?
                                ( 0.01 * parseFloat( RegExp.$1 ) ) + "" :
                                computed ? "1" : "";
                },

                set: function( elem, value ) {
                        var style = elem.style,
```

```
                        currentStyle = elem.currentStyle,
                        opacity = jQuery.isNumeric( value ) ?
"alpha(opacity=" + value * 100 + ")" : "",
                        filter = currentStyle && currentStyle.filter ||
style.filter || "";

                // IE has trouble with opacity if it does not have layout
                // Force it by setting the zoom level
                style.zoom = 1;

                // if setting opacity to 1, and no other filters exist -
attempt to remove filter attribute #6652
                // if value === "", then remove inline opacity #12685
                if ( ( value >= 1 || value === "" ) &&
                                jQuery.trim( filter.replace( ralpha, "" ) ) ===
"" &&
                                style.removeAttribute ) {

                        // Setting style.filter to null, "" & " " still leave
"filter:" in the cssText
                        // if "filter:" is present at all, clearType is disabled,
we want to avoid this
                        // style.removeAttribute is IE Only, but so apparently
is this code path...
                        style.removeAttribute( "filter" );

                        // if there is no filter style applied in a css rule or
unset inline opacity, we are done
                        if ( value === "" || currentStyle &&
!currentStyle.filter ) {
                                return;
                        }
                }

                // otherwise, set new filter values
                style.filter = ralpha.test( filter ) ?
                        filter.replace( ralpha, opacity ) :
                        filter + " " + opacity;
```

```
		}
	};
}

// These hooks cannot be added until DOM ready because the support test
// for it is not run until after DOM ready
jQuery(function() {
	if ( !jQuery.support.reliableMarginRight ) {
		jQuery.cssHooks.marginRight = {
			get: function( elem, computed ) {
				if ( computed ) {
					// WebKit Bug 13343 - getComputedStyle
returns wrong value for margin-right
					// Work around by temporarily setting element
display to inline-block
					return jQuery.swap( elem, { "display": "inline-
block" },
						curCSS, [ elem, "marginRight" ] );
				}
			}
		};
	}

	// Webkit bug: https://bugs.webkit.org/show_bug.cgi?id=29084
	// getComputedStyle returns percent when specified for
top/left/bottom/right
	// rather than make the css module depend on the offset module, we
just check for it here
	if ( !jQuery.support.pixelPosition && jQuery.fn.position ) {
		jQuery.each( [ "top", "left" ], function( i, prop ) {
			jQuery.cssHooks[ prop ] = {
				get: function( elem, computed ) {
					if ( computed ) {
						computed = curCSS( elem, prop );
						// if curCSS returns percentage, fallback
to offset
						return rnumnonpx.test( computed ) ?
							jQuery( elem ).position()[ prop ] +
```

```
"px" :
                                                computed;
                                        }
                                }
                        };
                });
        }

});

if ( jQuery.expr && jQuery.expr.filters ) {
        jQuery.expr.filters.hidden = function( elem ) {
                // Support: Opera <= 12.12
                // Opera reports offsetWidths and offsetHeights less than zero on
some elements
                return elem.offsetWidth <= 0 && elem.offsetHeight <= 0 ||
                        (!jQuery.support.reliableHiddenOffsets && ((elem.style &&
elem.style.display) || jQuery.css( elem, "display" )) === "none");
        };

        jQuery.expr.filters.visible = function( elem ) {
                return !jQuery.expr.filters.hidden( elem );
        };
}

// These hooks are used by animate to expand properties
jQuery.each({
        margin: "",
        padding: "",
        border: "Width"
}, function( prefix, suffix ) {
        jQuery.cssHooks[ prefix + suffix ] = {
                expand: function( value ) {
                        var i = 0,
                                expanded = {},

                                // assumes a single number if not a string
                                parts = typeof value === "string" ? value.split(" ") :
```

```
[ value ];

                for ( ; i < 4; i++ ) {
                        expanded[ prefix + cssExpand[ i ] + suffix ] =
                                parts[ i ] || parts[ i - 2 ] || parts[ 0 ];
                }

                return expanded;
        }
    };

    if ( !rmargin.test( prefix ) ) {
        jQuery.cssHooks[ prefix + suffix ].set = setPositiveNumber;
    }
});
var r20 = /%20/g,
    rbracket = /\[\]$/,
    rCRLF = /\r?\n/g,
    rsubmitterTypes = /^(?:submit|button|image|reset|file)$/i,
    rsubmittable = /^(?:input|select|textarea|keygen)/i;

jQuery.fn.extend({
    serialize: function() {
        return jQuery.param( this.serializeArray() );
    },
    serializeArray: function() {
        return this.map(function(){
                // Can add propHook for "elements" to filter or add form
elements
                var elements = jQuery.prop( this, "elements" );
                return elements ? jQuery.makeArray( elements ) : this;
        })
        .filter(function(){
                var type = this.type;
                // Use .is(":disabled") so that fieldset[disabled] works
                return this.name && !jQuery( this ).is( ":disabled" ) &&
                        rsubmittable.test( this.nodeName ) &&
!rsubmitterTypes.test( type ) &&
```

```
                        ( this.checked || !manipulation_rcheckableType.test(
type ) );
            })
            .map(function( i, elem ){
                var val = jQuery( this ).val();

                return val == null ?
                    null :
                    jQuery.isArray( val ) ?
                        jQuery.map( val, function( val ){
                            return { name: elem.name, value:
val.replace( rCRLF, "\r\n" ) };
                        }) :
                        { name: elem.name, value: val.replace(
rCRLF, "\r\n" ) };
            }).get();
        }
});

//Serialize an array of form elements or a set of
//key/values into a query string
jQuery.param = function( a, traditional ) {
        var prefix,
            s = [],
            add = function( key, value ) {
                // If value is a function, invoke it and return its value
                value = jQuery.isFunction( value ) ? value() : ( value ==
null ? "" : value );
                s[ s.length ] = encodeURIComponent( key ) + "=" +
encodeURIComponent( value );
            };

        // Set traditional to true for jQuery <= 1.3.2 behavior.
        if ( traditional === undefined ) {
            traditional = jQuery.ajaxSettings &&
jQuery.ajaxSettings.traditional;
        }
```

```
	// If an array was passed in, assume that it is an array of form
elements.
	if ( jQuery.isArray( a ) || ( a.jquery && !jQuery.isPlainObject( a ) ) ) {
		// Serialize the form elements
		jQuery.each( a, function() {
			add( this.name, this.value );
		});

	} else {
		// If traditional, encode the "old" way (the way 1.3.2 or older
		// did it), otherwise encode params recursively.
		for ( prefix in a ) {
			buildParams( prefix, a[ prefix ], traditional, add );
		}
	}

	// Return the resulting serialization
	return s.join( "&" ).replace( r20, "+" );
};

function buildParams( prefix, obj, traditional, add ) {
	var name;

	if ( jQuery.isArray( obj ) ) {
		// Serialize array item.
		jQuery.each( obj, function( i, v ) {
			if ( traditional || rbracket.test( prefix ) ) {
				// Treat each array item as a scalar.
				add( prefix, v );

			} else {
				// Item is non-scalar (array or object), encode its
numeric index.
				buildParams( prefix + "[" + ( typeof v === "object"
? i : "" ) + "]", v, traditional, add );
			}
		});
```

```
	} else if ( !traditional && jQuery.type( obj ) === "object" ) {
		// Serialize object item.
		for ( name in obj ) {
			buildParams( prefix + "[" + name + "]", obj[ name ],
traditional, add );
		}

	} else {
		// Serialize scalar item.
		add( prefix, obj );
	}
}
jQuery.each( ("blur focus focusin focusout load resize scroll unload click
dblclick " +
	"mousedown mouseup mousemove mouseover mouseout mouseenter
mouseleave " +
	"change select submit keydown keypress keyup error
contextmenu").split(" "), function( i, name ) {

	// Handle event binding
	jQuery.fn[ name ] = function( data, fn ) {
		return arguments.length > 0 ?
			this.on( name, null, data, fn ) :
			this.trigger( name );
	};
});

jQuery.fn.extend({
	hover: function( fnOver, fnOut ) {
		return this.mouseenter( fnOver ).mouseleave( fnOut || fnOver );
	},

	bind: function( types, data, fn ) {
		return this.on( types, null, data, fn );
	},
	unbind: function( types, fn ) {
		return this.off( types, null, fn );
	},
```

```
        delegate: function( selector, types, data, fn ) {
                return this.on( types, selector, data, fn );
        },
        undelegate: function( selector, types, fn ) {
                // ( namespace ) or ( selector, types [, fn] )
                return arguments.length === 1 ? this.off( selector, "**" ) :
this.off( types, selector || "**", fn );
        }
});
var
        // Document location
        ajaxLocParts,
        ajaxLocation,
        ajax_nonce = jQuery.now(),

        ajax_rquery = /\?/,
        rhash = /#.*$/,
        rts = /([?&])_=[^&]*/,
        rheaders = /^(.*?):[ \t]*([^\r\n]*)\r?$/mg, // IE leaves an \r
character at EOL
        // #7653, #8125, #8152: local protocol detection
        rlocalProtocol = /^(?:about|app|app-storage|.+-
extension|file|res|widget):$/,
        rnoContent = /^(?:GET|HEAD)$/,
        rprotocol = /^\/\//,
        rurl = /^([\w.+-]+:)(?:\/\/([^\/?#:]*)(?::(\d+)|)|)/,

        // Keep a copy of the old load method
        _load = jQuery.fn.load,

        /* Prefilters
         * 1) They are useful to introduce custom dataTypes (see ajax/jsonp.js
for an example)
         * 2) These are called:
         *    - BEFORE asking for a transport
         *    - AFTER param serialization (s.data is a string if s.processData is
true)
```

```
         * 3) key is the dataType
         * 4) the catchall symbol "*" can be used
         * 5) execution will start with transport dataType and THEN continue
down to "*" if needed
         */
        prefilters = {},

        /* Transports bindings
         * 1) key is the dataType
         * 2) the catchall symbol "*" can be used
         * 3) selection will start with transport dataType and THEN go to "*" if
needed
         */
        transports = {},

        // Avoid comment-prolog char sequence (#10098); must appease lint
and evade compression
        allTypes = "*/".concat("*");

// #8138, IE may throw an exception when accessing
// a field from window.location if document.domain has been set
try {
        ajaxLocation = location.href;
} catch( e ) {
        // Use the href attribute of an A element
        // since IE will modify it given document.location
        ajaxLocation = document.createElement( "a" );
        ajaxLocation.href = "";
        ajaxLocation = ajaxLocation.href;
}

// Segment location into parts
ajaxLocParts = rurl.exec( ajaxLocation.toLowerCase() ) || [];

// Base "constructor" for jQuery.ajaxPrefilter and jQuery.ajaxTransport
function addToPrefiltersOrTransports( structure ) {

        // dataTypeExpression is optional and defaults to "*"
```

```
	return function( dataTypeExpression, func ) {

		if ( typeof dataTypeExpression !== "string" ) {
			func = dataTypeExpression;
			dataTypeExpression = "*";
		}

		var dataType,
			i = 0,
			dataTypes = dataTypeExpression.toLowerCase().match(
core_rnotwhite ) || [];

		if ( jQuery.isFunction( func ) ) {
			// For each dataType in the dataTypeExpression
			while ( (dataType = dataTypes[i++]) ) {
				// Prepend if requested
				if ( dataType[0] === "+" ) {
					dataType = dataType.slice( 1 ) || "*";
					(structure[ dataType ] = structure[ dataType ]
|| []).unshift( func );

				// Otherwise append
				} else {
					(structure[ dataType ] = structure[ dataType ]
|| []).push( func );
				}
			}
		}
	};
}

// Base inspection function for prefilters and transports
function inspectPrefiltersOrTransports( structure, options, originalOptions,
jqXHR ) {

	var inspected = {},
		seekingTransport = ( structure === transports );
```

```
	function inspect( dataType ) {
		var selected;
		inspected[ dataType ] = true;
		jQuery.each( structure[ dataType ] || [], function( _,
prefilterOrFactory ) {
			var dataTypeOrTransport = prefilterOrFactory( options,
originalOptions, jqXHR );
			if( typeof dataTypeOrTransport === "string" &&
!seekingTransport && !inspected[ dataTypeOrTransport ] ) {
				options.dataTypes.unshift( dataTypeOrTransport );
				inspect( dataTypeOrTransport );
				return false;
			} else if ( seekingTransport ) {
				return !( selected = dataTypeOrTransport );
			}
		});
		return selected;
	}

	return inspect( options.dataTypes[ 0 ] ) || !inspected[ "*" ] &&
inspect( "*" );
}

// A special extend for ajax options
// that takes "flat" options (not to be deep extended)
// Fixes #9887
function ajaxExtend( target, src ) {
	var deep, key,
		flatOptions = jQuery.ajaxSettings.flatOptions || {};

	for ( key in src ) {
		if ( src[ key ] !== undefined ) {
			( flatOptions[ key ] ? target : ( deep || (deep = {}) ) )[
key ] = src[ key ];
		}
	}
	if ( deep ) {
		jQuery.extend( true, target, deep );
```

```
	}

	return target;
}

jQuery.fn.load = function( url, params, callback ) {
	if ( typeof url !== "string" && _load ) {
		return _load.apply( this, arguments );
	}

	var selector, response, type,
		self = this,
		off = url.indexOf(" ");

	if ( off >= 0 ) {
		selector = url.slice( off, url.length );
		url = url.slice( 0, off );
	}

	// If it's a function
	if ( jQuery.isFunction( params ) ) {

		// We assume that it's the callback
		callback = params;
		params = undefined;

	// Otherwise, build a param string
	} else if ( params && typeof params === "object" ) {
		type = "POST";
	}

	// If we have elements to modify, make the request
	if ( self.length > 0 ) {
		jQuery.ajax({
			url: url,

			// if "type" variable is undefined, then "GET" method will
be used
```

```
                    type: type,
                    dataType: "html",
                    data: params
            }).done(function( responseText ) {

                    // Save response for use in complete callback
                    response = arguments;

                    self.html( selector ?

                            // If a selector was specified, locate the right
elements in a dummy div
                            // Exclude scripts to avoid IE 'Permission Denied'
errors
                            jQuery("<div>").append( jQuery.parseHTML(
responseText ) ).find( selector ) :

                            // Otherwise use the full result
                            responseText );

            }).complete( callback && function( jqXHR, status ) {
                    self.each( callback, response || [ jqXHR.responseText,
status, jqXHR ] );
            });
    }

    return this;
};

// Attach a bunch of functions for handling common AJAX events
jQuery.each( [ "ajaxStart", "ajaxStop", "ajaxComplete", "ajaxError",
"ajaxSuccess", "ajaxSend" ], function( i, type ){
    jQuery.fn[ type ] = function( fn ){
            return this.on( type, fn );
    };
});

jQuery.extend({
```

```
        // Counter for holding the number of active queries
        active: 0,

        // Last-Modified header cache for next request
        lastModified: {},
        etag: {},

        ajaxSettings: {
                url: ajaxLocation,
                type: "GET",
                isLocal: rlocalProtocol.test( ajaxLocParts[ 1 ] ),
                global: true,
                processData: true,
                async: true,
                contentType: "application/x-www-form-urlencoded;
charset=UTF-8",
                /*
                timeout: 0,
                data: null,
                dataType: null,
                username: null,
                password: null,
                cache: null,
                throws: false,
                traditional: false,
                headers: {},
                */

                accepts: {
                        "*": allTypes,
                        text: "text/plain",
                        html: "text/html",
                        xml: "application/xml, text/xml",
                        json: "application/json, text/javascript"
                },

                contents: {
```

```
                xml: /xml/,
                html: /html/,
                json: /json/
            },

            responseFields: {
                xml: "responseXML",
                text: "responseText",
                json: "responseJSON"
            },

            // Data converters
            // Keys separate source (or catchall "*") and destination types
with a single space
            converters: {

                // Convert anything to text
                "* text": String,

                // Text to html (true = no transformation)
                "text html": true,

                // Evaluate text as a json expression
                "text json": jQuery.parseJSON,

                // Parse text as xml
                "text xml": jQuery.parseXML
            },

            // For options that shouldn't be deep extended:
            // you can add your own custom options here if
            // and when you create one that shouldn't be
            // deep extended (see ajaxExtend)
            flatOptions: {
                url: true,
                context: true
            }
        },
```

```
	// Creates a full fledged settings object into target
	// with both ajaxSettings and settings fields.
	// If target is omitted, writes into ajaxSettings.
	ajaxSetup: function( target, settings ) {
		return settings ?

			// Building a settings object
			ajaxExtend( ajaxExtend( target, jQuery.ajaxSettings ),
settings ) :

			// Extending ajaxSettings
			ajaxExtend( jQuery.ajaxSettings, target );
	},

	ajaxPrefilter: addToPrefiltersOrTransports( prefilters ),
	ajaxTransport: addToPrefiltersOrTransports( transports ),

	// Main method
	ajax: function( url, options ) {

		// If url is an object, simulate pre-1.5 signature
		if ( typeof url === "object" ) {
			options = url;
			url = undefined;
		}

		// Force options to be an object
		options = options || {};

		var // Cross-domain detection vars
			parts,
			// Loop variable
			i,
			// URL without anti-cache param
			cacheURL,
			// Response headers as string
			responseHeadersString,
```

```
                    // timeout handle
                    timeoutTimer,

                    // To know if global events are to be dispatched
                    fireGlobals,

                    transport,
                    // Response headers
                    responseHeaders,
                    // Create the final options object
                    s = jQuery.ajaxSetup( {}, options ),
                    // Callbacks context
                    callbackContext = s.context || s,
                    // Context for global events is callbackContext if it is a
DOM node or jQuery collection
                    globalEventContext = s.context && (
callbackContext.nodeType || callbackContext.jquery ) ?
                            jQuery( callbackContext ) :
                            jQuery.event,
                    // Deferreds
                    deferred = jQuery.Deferred(),
                    completeDeferred = jQuery.Callbacks("once memory"),
                    // Status-dependent callbacks
                    statusCode = s.statusCode || {},
                    // Headers (they are sent all at once)
                    requestHeaders = {},
                    requestHeadersNames = {},
                    // The jqXHR state
                    state = 0,
                    // Default abort message
                    strAbort = "canceled",
                    // Fake xhr
                    jqXHR = {
                            readyState: 0,

                            // Builds headers hashtable if needed
                            getResponseHeader: function( key ) {
                                    var match;
```

```
                                if ( state === 2 ) {
                                        if ( !responseHeaders ) {
                                                responseHeaders = {};
                                                while ( (match = rheaders.exec(
responseHeadersString )) ) {
                                                        responseHeaders[
match[1].toLowerCase() ] = match[ 2 ];
                                                }
                                        }
                                        match = responseHeaders[
key.toLowerCase() ];
                                }
                                return match == null ? null : match;
                        },

                        // Raw string
                        getAllResponseHeaders: function() {
                                return state === 2 ? responseHeadersString :
null;
                        },

                        // Caches the header
                        setRequestHeader: function( name, value ) {
                                var lname = name.toLowerCase();
                                if ( !state ) {
                                        name = requestHeadersNames[ lname ]
= requestHeadersNames[ lname ] || name;
                                        requestHeaders[ name ] = value;
                                }
                                return this;
                        },

                        // Overrides response content-type header
                        overrideMimeType: function( type ) {
                                if ( !state ) {
                                        s.mimeType = type;
                                }
                                return this;
```

```
                        },

                        // Status-dependent callbacks
                        statusCode: function( map ) {
                                var code;
                                if ( map ) {
                                        if ( state < 2 ) {
                                                for ( code in map ) {
                                                        // Lazy-add the new callback
in a way that preserves old ones
                                                        statusCode[ code ] = [
statusCode[ code ], map[ code ] ];
                                                }
                                        } else {
                                                // Execute the appropriate
callbacks
                                                jqXHR.always( map[ jqXHR.status
] );
                                        }
                                }
                                return this;
                        },

                        // Cancel the request
                        abort: function( statusText ) {
                                var finalText = statusText || strAbort;
                                if ( transport ) {
                                        transport.abort( finalText );
                                }
                                done( 0, finalText );
                                return this;
                        }
                };

        // Attach deferreds
        deferred.promise( jqXHR ).complete = completeDeferred.add;
        jqXHR.success = jqXHR.done;
        jqXHR.error = jqXHR.fail;
```

```
		// Remove hash character (#7531: and string promotion)
		// Add protocol if not provided (#5866: IE7 issue with protocol-
less urls)
		// Handle falsy url in the settings object (#10093: consistency
with old signature)
		// We also use the url parameter if available
		s.url = ( ( url || s.url || ajaxLocation ) + "" ).replace( rhash, ""
).replace( rprotocol, ajaxLocParts[ 1 ] + "//" );

		// Alias method option to type as per ticket #12004
		s.type = options.method || options.type || s.method || s.type;

		// Extract dataTypes list
		s.dataTypes = jQuery.trim( s.dataType || "*"
).toLowerCase().match( core_rnotwhite ) || [""];

		// A cross-domain request is in order when we have a
protocol:host:port mismatch
		if ( s.crossDomain == null ) {
			parts = rurl.exec( s.url.toLowerCase() );
			s.crossDomain = !!( parts &&
				( parts[ 1 ] !== ajaxLocParts[ 1 ] || parts[ 2 ] !==
ajaxLocParts[ 2 ] ||
					( parts[ 3 ] || ( parts[ 1 ] === "http:" ? "80" :
"443" ) ) !==
						( ajaxLocParts[ 3 ] || ( ajaxLocParts[ 1 ]
=== "http:" ? "80" : "443" ) ) )
			);
		}

		// Convert data if not already a string
		if ( s.data && s.processData && typeof s.data !== "string" ) {
			s.data = jQuery.param( s.data, s.traditional );
		}

		// Apply prefilters
		inspectPrefiltersOrTransports( prefilters, s, options, jqXHR );
```

```
		// If request was aborted inside a prefilter, stop there
		if ( state === 2 ) {
			return jqXHR;
		}

		// We can fire global events as of now if asked to
		fireGlobals = s.global;

		// Watch for a new set of requests
		if ( fireGlobals && jQuery.active++ === 0 ) {
			jQuery.event.trigger("ajaxStart");
		}

		// Uppercase the type
		s.type = s.type.toUpperCase();

		// Determine if request has content
		s.hasContent = !rnoContent.test( s.type );

		// Save the URL in case we're toying with the If-Modified-Since
		// and/or If-None-Match header later on
		cacheURL = s.url;

		// More options handling for requests with no content
		if ( !s.hasContent ) {

			// If data is available, append data to url
			if ( s.data ) {
				cacheURL = ( s.url += ( ajax_rquery.test( cacheURL
) ? "&" : "?" ) + s.data );
				// #9682: remove data so that it's not used in an
eventual retry
				delete s.data;
			}

			// Add anti-cache in url if needed
			if ( s.cache === false ) {
```

```
                        s.url = rts.test( cacheURL ) ?

                                // If there is already a '_' parameter, set its
value
                                cacheURL.replace( rts, "$1_=" +
ajax_nonce++ ) :

                                // Otherwise add one to the end
                                cacheURL + ( ajax_rquery.test( cacheURL ) ?
"&" : "?" ) + "_=" + ajax_nonce++;
                        }
                }

                // Set the If-Modified-Since and/or If-None-Match header, if in
ifModified mode.
                if ( s.ifModified ) {
                        if ( jQuery.lastModified[ cacheURL ] ) {
                                jqXHR.setRequestHeader( "If-Modified-Since",
jQuery.lastModified[ cacheURL ] );
                        }
                        if ( jQuery.etag[ cacheURL ] ) {
                                jqXHR.setRequestHeader( "If-None-Match",
jQuery.etag[ cacheURL ] );
                        }
                }

                // Set the correct header, if data is being sent
                if ( s.data && s.hasContent && s.contentType !== false ||
options.contentType ) {
                        jqXHR.setRequestHeader( "Content-Type", s.contentType
);
                }

                // Set the Accepts header for the server, depending on the
dataType
                jqXHR.setRequestHeader(
                        "Accept",
                        s.dataTypes[ 0 ] && s.accepts[ s.dataTypes[0] ] ?
```

```
                        s.accepts[ s.dataTypes[0] ] + ( s.dataTypes[ 0 ] !==
"*" ? ", " + allTypes + "; q=0.01" : "" ) :
                        s.accepts[ "*" ]
            );

            // Check for headers option
            for ( i in s.headers ) {
                    jqXHR.setRequestHeader( i, s.headers[ i ] );
            }

            // Allow custom headers/mimetypes and early abort
            if ( s.beforeSend && ( s.beforeSend.call( callbackContext, jqXHR,
s ) === false || state === 2 ) ) {
                    // Abort if not done already and return
                    return jqXHR.abort();
            }

            // aborting is no longer a cancellation
            strAbort = "abort";

            // Install callbacks on deferreds
            for ( i in { success: 1, error: 1, complete: 1 } ) {
                    jqXHR[ i ]( s[ i ] );
            }

            // Get transport
            transport = inspectPrefiltersOrTransports( transports, s, options,
jqXHR );

            // If no transport, we auto-abort
            if ( !transport ) {
                    done( -1, "No Transport" );
            } else {
                    jqXHR.readyState = 1;

                    // Send global event
                    if ( fireGlobals ) {
                            globalEventContext.trigger( "ajaxSend", [ jqXHR, s ]
```

```
);
                }
                // Timeout
                if ( s.async && s.timeout > 0 ) {
                        timeoutTimer = setTimeout(function() {
                                jqXHR.abort("timeout");
                        }, s.timeout );
                }

                try {
                        state = 1;
                        transport.send( requestHeaders, done );
                } catch ( e ) {
                        // Propagate exception as error if not done
                        if ( state < 2 ) {
                                done( -1, e );
                        // Simply rethrow otherwise
                        } else {
                                throw e;
                        }
                }
        }

        // Callback for when everything is done
        function done( status, nativeStatusText, responses, headers ) {
                var isSuccess, success, error, response, modified,
                        statusText = nativeStatusText;

                // Called once
                if ( state === 2 ) {
                        return;
                }

                // State is "done" now
                state = 2;

                // Clear timeout if it exists
                if ( timeoutTimer ) {
```

```
                        clearTimeout( timeoutTimer );
                }

                // Dereference transport for early garbage collection
                // (no matter how long the jqXHR object will be used)
                transport = undefined;

                // Cache response headers
                responseHeadersString = headers || "";

                // Set readyState
                jqXHR.readyState = status > 0 ? 4 : 0;

                // Determine if successful
                isSuccess = status >= 200 && status < 300 || status ===
304;

                // Get response data
                if ( responses ) {
                        response = ajaxHandleResponses( s, jqXHR,
responses );
                }

                // Convert no matter what (that way responseXXX fields
are always set)
                response = ajaxConvert( s, response, jqXHR, isSuccess );

                // If successful, handle type chaining
                if ( isSuccess ) {

                        // Set the If-Modified-Since and/or If-None-Match
header, if in ifModified mode.
                        if ( s.ifModified ) {
                                modified = jqXHR.getResponseHeader("Last-
Modified");
                                if ( modified ) {
                                        jQuery.lastModified[ cacheURL ] =
modified;
```

```
                        }
                        modified = jqXHR.getResponseHeader("etag");
                        if ( modified ) {
                                jQuery.etag[ cacheURL ] = modified;
                        }
                }

                // if no content
                if ( status === 204 || s.type === "HEAD" ) {
                        statusText = "nocontent";

                // if not modified
                } else if ( status === 304 ) {
                        statusText = "notmodified";

                // If we have data, let's convert it
                } else {
                        statusText = response.state;
                        success = response.data;
                        error = response.error;
                        isSuccess = !error;
                }
        } else {
                // We extract error from statusText
                // then normalize statusText and status for non-
aborts
                error = statusText;
                if ( status || !statusText ) {
                        statusText = "error";
                        if ( status < 0 ) {
                                status = 0;
                        }
                }
        }

        // Set data for the fake xhr object
        jqXHR.status = status;
        jqXHR.statusText = ( nativeStatusText || statusText ) +
```

```
"";

                    // Success/Error
                    if ( isSuccess ) {
                            deferred.resolveWith( callbackContext, [ success,
statusText, jqXHR ] );
                    } else {
                            deferred.rejectWith( callbackContext, [ jqXHR,
statusText, error ] );
                    }

                    // Status-dependent callbacks
                    jqXHR.statusCode( statusCode );
                    statusCode = undefined;

                    if ( fireGlobals ) {
                            globalEventContext.trigger( isSuccess ?
"ajaxSuccess" : "ajaxError",
                                    [ jqXHR, s, isSuccess ? success : error ] );
                    }

                    // Complete
                    completeDeferred.fireWith( callbackContext, [ jqXHR,
statusText ] );

                    if ( fireGlobals ) {
                            globalEventContext.trigger( "ajaxComplete", [
jqXHR, s ] );
                            // Handle the global AJAX counter
                            if ( !( --jQuery.active ) ) {
                                    jQuery.event.trigger("ajaxStop");
                            }
                    }
            }

            return jqXHR;
    },
```

```
	getJSON: function( url, data, callback ) {
		return jQuery.get( url, data, callback, "json" );
	},

	getScript: function( url, callback ) {
		return jQuery.get( url, undefined, callback, "script" );
	}
});

jQuery.each( [ "get", "post" ], function( i, method ) {
	jQuery[ method ] = function( url, data, callback, type ) {
		// shift arguments if data argument was omitted
		if ( jQuery.isFunction( data ) ) {
			type = type || callback;
			callback = data;
			data = undefined;
		}

		return jQuery.ajax({
			url: url,
			type: method,
			dataType: type,
			data: data,
			success: callback
		});
	};
});

/* Handles responses to an ajax request:
 * - finds the right dataType (mediates between content-type and expected
dataType)
 * - returns the corresponding response
 */
function ajaxHandleResponses( s, jqXHR, responses ) {
	var firstDataType, ct, finalDataType, type,
		contents = s.contents,
		dataTypes = s.dataTypes;
```

```
	// Remove auto dataType and get content-type in the process
	while( dataTypes[ 0 ] === "*" ) {
		dataTypes.shift();
		if ( ct === undefined ) {
			ct = s.mimeType || jqXHR.getResponseHeader("Content-
Type");
		}
	}

	// Check if we're dealing with a known content-type
	if ( ct ) {
		for ( type in contents ) {
			if ( contents[ type ] && contents[ type ].test( ct ) ) {
				dataTypes.unshift( type );
				break;
			}
		}
	}

	// Check to see if we have a response for the expected dataType
	if ( dataTypes[ 0 ] in responses ) {
		finalDataType = dataTypes[ 0 ];
	} else {
		// Try convertible dataTypes
		for ( type in responses ) {
			if ( !dataTypes[ 0 ] || s.converters[ type + " " +
dataTypes[0] ] ) {
				finalDataType = type;
				break;
			}
			if ( !firstDataType ) {
				firstDataType = type;
			}
		}
		// Or just use first one
		finalDataType = finalDataType || firstDataType;
	}
```

```
	// If we found a dataType
	// We add the dataType to the list if needed
	// and return the corresponding response
	if ( finalDataType ) {
		if ( finalDataType !== dataTypes[ 0 ] ) {
			dataTypes.unshift( finalDataType );
		}
		return responses[ finalDataType ];
	}
}

/* Chain conversions given the request and the original response
 * Also sets the responseXXX fields on the jqXHR instance
 */
function ajaxConvert( s, response, jqXHR, isSuccess ) {
	var conv2, current, conv, tmp, prev,
		converters = {},
		// Work with a copy of dataTypes in case we need to modify it
for conversion
		dataTypes = s.dataTypes.slice();

	// Create converters map with lowercased keys
	if ( dataTypes[ 1 ] ) {
		for ( conv in s.converters ) {
			converters[ conv.toLowerCase() ] = s.converters[ conv ];
		}
	}

	current = dataTypes.shift();

	// Convert to each sequential dataType
	while ( current ) {

		if ( s.responseFields[ current ] ) {
			jqXHR[ s.responseFields[ current ] ] = response;
		}

		// Apply the dataFilter if provided
```

```
		if ( !prev && isSuccess && s.dataFilter ) {
			response = s.dataFilter( response, s.dataType );
		}

		prev = current;
		current = dataTypes.shift();

		if ( current ) {

			// There's only work to do if current dataType is non-auto
			if ( current === "*" ) {

				current = prev;

			// Convert response if prev dataType is non-auto and
differs from current
			} else if ( prev !== "*" && prev !== current ) {

				// Seek a direct converter
				conv = converters[ prev + " " + current ] ||
converters[ "* " + current ];

				// If none found, seek a pair
				if ( !conv ) {
					for ( conv2 in converters ) {

						// If conv2 outputs current
						tmp = conv2.split( " " );
						if ( tmp[ 1 ] === current ) {

							// If prev can be converted to
accepted input
							conv = converters[ prev + " " +
tmp[ 0 ] ] ||
								converters[ "* " + tmp[ 0 ]
];
							if ( conv ) {
								// Condense equivalence
```

```
converters
                                                        if ( conv === true ) {
                                                                conv = converters[
conv2 ];

                                                        // Otherwise, insert the
intermediate dataType
                                                        } else if ( converters[ conv2
] !== true ) {
                                                                current = tmp[ 0 ];
                                                                dataTypes.unshift(
tmp[ 1 ] );
                                                        }
                                                        break;
                                                }
                                        }
                                }
                        }

                        // Apply converter (if not an equivalence)
                        if ( conv !== true ) {

                                // Unless errors are allowed to bubble, catch
and return them
                                if ( conv && s[ "throws" ] ) {
                                        response = conv( response );
                                } else {
                                        try {
                                                response = conv( response );
                                        } catch ( e ) {
                                                return { state: "parsererror",
error: conv ? e : "No conversion from " + prev + " to " + current };
                                        }
                                }
                        }
                }
            }
        }
```

```
	return { state: "success", data: response };
}
// Install script dataType
jQuery.ajaxSetup({
	accepts: {
		script: "text/javascript, application/javascript,
application/ecmascript, application/x-ecmascript"
	},
	contents: {
		script: /(?:java|ecma)script/
	},
	converters: {
		"text script": function( text ) {
			jQuery.globalEval( text );
			return text;
		}
	}
});

// Handle cache's special case and global
jQuery.ajaxPrefilter( "script", function( s ) {
	if ( s.cache === undefined ) {
		s.cache = false;
	}
	if ( s.crossDomain ) {
		s.type = "GET";
		s.global = false;
	}
});

// Bind script tag hack transport
jQuery.ajaxTransport( "script", function(s) {

	// This transport only deals with cross domain requests
	if ( s.crossDomain ) {

		var script,
```

```
                    head = document.head || jQuery("head")[0] ||
document.documentElement;

            return {

                    send: function( _, callback ) {

                            script = document.createElement("script");

                            script.async = true;

                            if ( s.scriptCharset ) {
                                    script.charset = s.scriptCharset;
                            }

                            script.src = s.url;

                            // Attach handlers for all browsers
                            script.onload = script.onreadystatechange =
function( _, isAbort ) {

                                    if ( isAbort || !script.readyState ||
/loaded|complete/.test( script.readyState ) ) {

                                            // Handle memory leak in IE
                                            script.onload =
script.onreadystatechange = null;

                                            // Remove the script
                                            if ( script.parentNode ) {
                                                    script.parentNode.removeChild(
script );
                                            }

                                            // Dereference the script
                                            script = null;

                                            // Callback if not abort
```

```
                                        if ( !isAbort ) {
                                                callback( 200, "success" );
                                        }
                                }
                        };

                        // Circumvent IE6 bugs with base elements (#2709
and #4378) by prepending
                        // Use native DOM manipulation to avoid our
domManip AJAX trickery
                        head.insertBefore( script, head.firstChild );
                },

                abort: function() {
                        if ( script ) {
                                script.onload( undefined, true );
                        }
                }
        };
    }
});
var oldCallbacks = [],
    rjsonp = /(=)\?(?=&|$)|\?\?/;

// Default jsonp settings
jQuery.ajaxSetup({
    jsonp: "callback",
    jsonpCallback: function() {
        var callback = oldCallbacks.pop() || ( jQuery.expando + "_" + (
ajax_nonce++ ) );
        this[ callback ] = true;
        return callback;
    }
});

// Detect, normalize options and install callbacks for jsonp requests
jQuery.ajaxPrefilter( "json jsonp", function( s, originalSettings, jqXHR ) {
```

```
	var callbackName, overwritten, responseContainer,
		jsonProp = s.jsonp !== false && ( rjsonp.test( s.url ) ?
			"url" :
			typeof s.data === "string" && !( s.contentType || ""
).indexOf("application/x-www-form-urlencoded") && rjsonp.test( s.data ) &&
"data"
		);

	// Handle iff the expected data type is "jsonp" or we have a parameter
to set
	if ( jsonProp || s.dataTypes[ 0 ] === "jsonp" ) {

		// Get callback name, remembering preexisting value associated
with it
		callbackName = s.jsonpCallback = jQuery.isFunction(
s.jsonpCallback ) ?
			s.jsonpCallback() :
			s.jsonpCallback;

		// Insert callback into url or form data
		if ( jsonProp ) {
			s[ jsonProp ] = s[ jsonProp ].replace( rjsonp, "$1" +
callbackName );
		} else if ( s.jsonp !== false ) {
			s.url += ( ajax_rquery.test( s.url ) ? "&" : "?" ) + s.jsonp
+ "=" + callbackName;
		}

		// Use data converter to retrieve json after script execution
		s.converters["script json"] = function() {
			if ( !responseContainer ) {
				jQuery.error( callbackName + " was not called" );
			}
			return responseContainer[ 0 ];
		};

		// force json dataType
		s.dataTypes[ 0 ] = "json";
```

```
		// Install callback
		overwritten = window[ callbackName ];
		window[ callbackName ] = function() {
			responseContainer = arguments;
		};

		// Clean-up function (fires after converters)
		jqXHR.always(function() {
			// Restore preexisting value
			window[ callbackName ] = overwritten;

			// Save back as free
			if ( s[ callbackName ] ) {
				// make sure that re-using the options doesn't screw
things around
				s.jsonpCallback = originalSettings.jsonpCallback;

				// save the callback name for future use
				oldCallbacks.push( callbackName );
			}

			// Call if it was a function and we have a response
			if ( responseContainer && jQuery.isFunction( overwritten )
) {
				overwritten( responseContainer[ 0 ] );
			}

			responseContainer = overwritten = undefined;
		});

		// Delegate to script
		return "script";
	}
});
var xhrCallbacks, xhrSupported,
	xhrId = 0,
	// #5280: Internet Explorer will keep connections alive if we don't
```

```
abort on unload
	xhrOnUnloadAbort = window.ActiveXObject && function() {
		// Abort all pending requests
		var key;
		for ( key in xhrCallbacks ) {
			xhrCallbacks[ key ]( undefined, true );
		}
	};

// Functions to create xhrs
function createStandardXHR() {
	try {
		return new window.XMLHttpRequest();
	} catch( e ) {}
}

function createActiveXHR() {
	try {
		return new window.ActiveXObject("Microsoft.XMLHTTP");
	} catch( e ) {}
}

// Create the request object
// (This is still attached to ajaxSettings for backward compatibility)
jQuery.ajaxSettings.xhr = window.ActiveXObject ?
	/* Microsoft failed to properly
	 * implement the XMLHttpRequest in IE7 (can't request local files),
	 * so we use the ActiveXObject when it is available
	 * Additionally XMLHttpRequest can be disabled in IE7/IE8 so
	 * we need a fallback.
	 */
	function() {
		return !this.isLocal && createStandardXHR() ||
createActiveXHR();
	} :
	// For all other browsers, use the standard XMLHttpRequest object
	createStandardXHR;
```

```
// Determine support properties
xhrSupported = jQuery.ajaxSettings.xhr();
jQuery.support.cors = !!xhrSupported && ( "withCredentials" in
xhrSupported );
xhrSupported = jQuery.support.ajax = !!xhrSupported;

// Create transport if the browser can provide an xhr
if ( xhrSupported ) {

	jQuery.ajaxTransport(function( s ) {
		// Cross domain only allowed if supported through
XMLHttpRequest
		if ( !s.crossDomain || jQuery.support.cors ) {

			var callback;

			return {
				send: function( headers, complete ) {

					// Get a new xhr
					var handle, i,
						xhr = s.xhr();

					// Open the socket
					// Passing null username, generates a login
popup on Opera (#2865)
					if ( s.username ) {
						xhr.open( s.type, s.url, s.async,
s.username, s.password );
					} else {
						xhr.open( s.type, s.url, s.async );
					}

					// Apply custom fields if provided
					if ( s.xhrFields ) {
						for ( i in s.xhrFields ) {
							xhr[ i ] = s.xhrFields[ i ];
						}
```

```
                                }

                                // Override mime type if needed
                                if ( s.mimeType && xhr.overrideMimeType ) {
                                        xhr.overrideMimeType( s.mimeType );
                                }

                                // X-Requested-With header
                                // For cross-domain requests, seeing as
conditions for a preflight are
                                // akin to a jigsaw puzzle, we simply never set
it to be sure.
                                // (it can always be set on a per-request basis
or even using ajaxSetup)
                                // For same-domain requests, won't change
header if already provided.
                                if ( !s.crossDomain && !headers["X-Requested-
With"] ) {
                                        headers["X-Requested-With"] =
"XMLHttpRequest";
                                }

                                // Need an extra try/catch for cross domain
requests in Firefox 3
                                try {
                                        for ( i in headers ) {
                                                xhr.setRequestHeader( i, headers[
i ] );
                                        }
                                } catch( err ) {}

                                // Do send the request
                                // This may raise an exception which is actually
                                // handled in jQuery.ajax (so no try/catch
here)
                                xhr.send( ( s.hasContent && s.data ) || null );

                                // Listener
```

```
                                callback = function( _, isAbort ) {
                                        var status, responseHeaders, statusText,
responses;

                                        // Firefox throws exceptions when
accessing properties
                                        // of an xhr when a network error
occurred
                                        // http://helpful.knobs-
dials.com/index.php/Component_returned_failure_code:_0x80040111_(NS_
ERROR_NOT_AVAILABLE)
                                        try {

                                                // Was never called and is aborted
or complete
                                                if ( callback && ( isAbort ||
xhr.readyState === 4 ) ) {

                                                        // Only called once
                                                        callback = undefined;

                                                        // Do not keep as active
anymore
                                                        if ( handle ) {

        xhr.onreadystatechange = jQuery.noop;
                                                                if ( xhrOnUnloadAbort )
{
                                                                        delete
xhrCallbacks[ handle ];
                                                                }
                                                        }

                                                        // If it's an abort
                                                        if ( isAbort ) {
                                                                // Abort it manually if
needed
                                                                if ( xhr.readyState !==
```

```
4 ) {
                                                                xhr.abort();
                                                        }
                                                } else {
                                                        responses = {};
                                                        status = xhr.status;
                                                        responseHeaders =
xhr.getAllResponseHeaders();

                                                        // When requesting
binary data, IE6-9 will throw an exception
                                                        // on any attempt to
access responseText (#11426)
                                                        if ( typeof
xhr.responseText === "string" ) {
                                                                responses.text =
xhr.responseText;
                                                        }

                                                        // Firefox throws an
exception when accessing
                                                        // statusText for faulty
cross-domain requests
                                                        try {
                                                                statusText =
xhr.statusText;
                                                        } catch( e ) {
                                                                // We normalize
with Webkit giving an empty statusText
                                                                statusText = "";
                                                        }

                                                        // Filter status for non
standard behaviors

                                                        // If the request is local
and we have data: assume a success
                                                        // (success with no
```

```
data won't get notified, that's the best we
                                                        // can do given current
implementations)
                                                        if ( !status && s.isLocal
&& !s.crossDomain ) {
                                                                status =
responses.text ? 200 : 404;
                                                        // IE - #1450:
sometimes returns 1223 when it should be 204
                                                        } else if ( status ===
1223 ) {
                                                                status = 204;
                                                        }
                                                }
                                        }
                                } catch( firefoxAccessException ) {
                                        if ( !isAbort ) {
                                                complete( -1,
firefoxAccessException );
                                        }
                                }

                                // Call complete if needed
                                if ( responses ) {
                                        complete( status, statusText,
responses, responseHeaders );
                                }
                        };

                        if ( !s.async ) {
                                // if we're in sync mode we fire the
callback
                                callback();
                        } else if ( xhr.readyState === 4 ) {
                                // (IE6 & IE7) if it's in cache and has
been
                                // retrieved directly we need to fire the
callback
```

```
                                        setTimeout( callback );
                                } else {
                                        handle = ++xhrId;
                                        if ( xhrOnUnloadAbort ) {
                                                // Create the active xhrs callbacks
list if needed
                                                // and attach the unload handler
                                                if ( !xhrCallbacks ) {
                                                        xhrCallbacks = {};
                                                        jQuery( window ).unload(
xhrOnUnloadAbort );
                                                }
                                                // Add to list of active xhrs
callbacks
                                                xhrCallbacks[ handle ] = callback;
                                        }
                                        xhr.onreadystatechange = callback;
                                }
                        },

                        abort: function() {
                                if ( callback ) {
                                        callback( undefined, true );
                                }
                        }
                };
        }
    });
}
var fxNow, timerId,
    rfxtypes = /^(?:toggle|show|hide)$/,
    rfxnum = new RegExp( "^(?:([+-])=|)(" + core_pnum + ")([a-
z%]*)$", "i" ),
    rrun = /queueHooks$/,
    animationPrefilters = [ defaultPrefilter ],
    tweeners = {
        "*": [function( prop, value ) {
                var tween = this.createTween( prop, value ),
```

```
                        target = tween.cur(),
                        parts = rfxnum.exec( value ),
                        unit = parts && parts[ 3 ] || ( jQuery.cssNumber[
prop ] ? "" : "px" ),

                        // Starting value computation is required for
potential unit mismatches
                        start = ( jQuery.cssNumber[ prop ] || unit !== "px"
&& +target ) &&
                                rfxnum.exec( jQuery.css( tween.elem, prop )
),
                        scale = 1,
                        maxIterations = 20;

                    if ( start && start[ 3 ] !== unit ) {
                        // Trust units reported by jQuery.css
                        unit = unit || start[ 3 ];

                        // Make sure we update the tween properties later
on
                        parts = parts || [];

                        // Iteratively approximate from a nonzero starting
point
                        start = +target || 1;

                        do {
                                // If previous iteration zeroed out, double until
we get *something*
                                // Use a string for doubling factor so we don't
accidentally see scale as unchanged below
                                scale = scale || ".5";

                                // Adjust and apply
                                start = start / scale;
                                jQuery.style( tween.elem, prop, start + unit );

                        // Update scale, tolerating zero or NaN from
```

```
tween.cur()
                        // And breaking the loop if scale is unchanged or
perfect, or if we've just had enough
                        } while ( scale !== (scale = tween.cur() / target) &&
scale !== 1 && --maxIterations );
                }

                // Update tween properties
                if ( parts ) {
                        start = tween.start = +start || +target || 0;
                        tween.unit = unit;
                        // If a +=/-= token was provided, we're doing a
relative animation
                        tween.end = parts[ 1 ] ?
                                start + ( parts[ 1 ] + 1 ) * parts[ 2 ] :
                                +parts[ 2 ];
                }

                return tween;
        }]
};

// Animations created synchronously will run synchronously
function createFxNow() {
        setTimeout(function() {
                fxNow = undefined;
        });
        return ( fxNow = jQuery.now() );
}

function createTween( value, prop, animation ) {
        var tween,
                collection = ( tweeners[ prop ] || [] ).concat( tweeners[ "*" ] ),
                index = 0,
                length = collection.length;
        for ( ; index < length; index++ ) {
                if ( (tween = collection[ index ].call( animation, prop, value )) )
{
```

```
                        // we're done with this property
                        return tween;
                }
        }
}

function Animation( elem, properties, options ) {
        var result,
                stopped,
                index = 0,
                length = animationPrefilters.length,
                deferred = jQuery.Deferred().always( function() {
                        // don't match elem in the :animated selector
                        delete tick.elem;
                }),
                tick = function() {
                        if ( stopped ) {
                                return false;
                        }
                        var currentTime = fxNow || createFxNow(),
                                remaining = Math.max( 0, animation.startTime +
animation.duration - currentTime ),
                                // archaic crash bug won't allow us to use 1 - ( 0.5 ||
0 ) (#12497)
                                temp = remaining / animation.duration || 0,
                                percent = 1 - temp,
                                index = 0,
                                length = animation.tweens.length;

                        for ( ; index < length ; index++ ) {
                                animation.tweens[ index ].run( percent );
                        }

                        deferred.notifyWith( elem, [ animation, percent, remaining
]);

                        if ( percent < 1 && length ) {
```

```
                        return remaining;
                } else {
                        deferred.resolveWith( elem, [ animation ] );
                        return false;
                }
        },
        animation = deferred.promise({
                elem: elem,
                props: jQuery.extend( {}, properties ),
                opts: jQuery.extend( true, { specialEasing: {} }, options ),
                originalProperties: properties,
                originalOptions: options,
                startTime: fxNow || createFxNow(),
                duration: options.duration,
                tweens: [],
                createTween: function( prop, end ) {
                        var tween = jQuery.Tween( elem, animation.opts,
prop, end,
                                        animation.opts.specialEasing[ prop ] ||
animation.opts.easing );
                        animation.tweens.push( tween );
                        return tween;
                },
                stop: function( gotoEnd ) {
                        var index = 0,
                                // if we are going to the end, we want to run
all the tweens
                                // otherwise we skip this part
                                length = gotoEnd ? animation.tweens.length :
0;
                        if ( stopped ) {
                                return this;
                        }
                        stopped = true;
                        for ( ; index < length ; index++ ) {
                                animation.tweens[ index ].run( 1 );
                        }
```

```
                        // resolve when we played the last frame
                        // otherwise, reject
                        if ( gotoEnd ) {
                                deferred.resolveWith( elem, [ animation,
gotoEnd ] );
                        } else {
                                deferred.rejectWith( elem, [ animation,
gotoEnd ] );
                        }
                        return this;
                }
            }),
            props = animation.props;

        propFilter( props, animation.opts.specialEasing );

        for ( ; index < length ; index++ ) {
            result = animationPrefilters[ index ].call( animation, elem,
props, animation.opts );
            if ( result ) {
                return result;
            }
        }

        jQuery.map( props, createTween, animation );

        if ( jQuery.isFunction( animation.opts.start ) ) {
            animation.opts.start.call( elem, animation );
        }

        jQuery.fx.timer(
            jQuery.extend( tick, {
                elem: elem,
                anim: animation,
                queue: animation.opts.queue
            })
        );
```

```
	// attach callbacks from options
	return animation.progress( animation.opts.progress )
		.done( animation.opts.done, animation.opts.complete )
		.fail( animation.opts.fail )
		.always( animation.opts.always );
}

function propFilter( props, specialEasing ) {
	var index, name, easing, value, hooks;

	// camelCase, specialEasing and expand cssHook pass
	for ( index in props ) {
		name = jQuery.camelCase( index );
		easing = specialEasing[ name ];
		value = props[ index ];
		if ( jQuery.isArray( value ) ) {
			easing = value[ 1 ];
			value = props[ index ] = value[ 0 ];
		}

		if ( index !== name ) {
			props[ name ] = value;
			delete props[ index ];
		}

		hooks = jQuery.cssHooks[ name ];
		if ( hooks && "expand" in hooks ) {
			value = hooks.expand( value );
			delete props[ name ];

			// not quite $.extend, this wont overwrite keys already
present.
			// also - reusing 'index' from above because we have the
correct "name"
			for ( index in value ) {
				if ( !( index in props ) ) {
					props[ index ] = value[ index ];
					specialEasing[ index ] = easing;
```

```
                        }
                }
            } else {
                specialEasing[ name ] = easing;
            }
        }
}

jQuery.Animation = jQuery.extend( Animation, {

        tweener: function( props, callback ) {
            if ( jQuery.isFunction( props ) ) {
                callback = props;
                props = [ "*" ];
            } else {
                props = props.split(" ");
            }

            var prop,
                index = 0,
                length = props.length;

            for ( ; index < length ; index++ ) {
                prop = props[ index ];
                tweeners[ prop ] = tweeners[ prop ] || [];
                tweeners[ prop ].unshift( callback );
            }
        },

        prefilter: function( callback, prepend ) {
            if ( prepend ) {
                animationPrefilters.unshift( callback );
            } else {
                animationPrefilters.push( callback );
            }
        }
});
```

```
function defaultPrefilter( elem, props, opts ) {
	/* jshint validthis: true */
	var prop, value, toggle, tween, hooks, oldfire,
		anim = this,
		orig = {},
		style = elem.style,
		hidden = elem.nodeType && isHidden( elem ),
		dataShow = jQuery._data( elem, "fxshow" );

	// handle queue: false promises
	if ( !opts.queue ) {
		hooks = jQuery._queueHooks( elem, "fx" );
		if ( hooks.unqueued == null ) {
			hooks.unqueued = 0;
			oldfire = hooks.empty.fire;
			hooks.empty.fire = function() {
				if ( !hooks.unqueued ) {
					oldfire();
				}
			};
		}
		hooks.unqueued++;

		anim.always(function() {
			// doing this makes sure that the complete handler will be
called
			// before this completes
			anim.always(function() {
				hooks.unqueued--;
				if ( !jQuery.queue( elem, "fx" ).length ) {
					hooks.empty.fire();
				}
			});
		});
	}

	// height/width overflow pass
	if ( elem.nodeType === 1 && ( "height" in props || "width" in props )
```

```
) {
		// Make sure that nothing sneaks out
		// Record all 3 overflow attributes because IE does not
		// change the overflow attribute when overflowX and
		// overflowY are set to the same value
		opts.overflow = [ style.overflow, style.overflowX, style.overflowY
];

		// Set display property to inline-block for height/width
		// animations on inline elements that are having width/height
animated
		if ( jQuery.css( elem, "display" ) === "inline" &&
				jQuery.css( elem, "float" ) === "none" ) {

			// inline-level elements accept inline-block;
			// block-level elements need to be inline with layout
			if ( !jQuery.support.inlineBlockNeedsLayout ||
css_defaultDisplay( elem.nodeName ) === "inline" ) {
				style.display = "inline-block";

			} else {
				style.zoom = 1;
			}
		}
	}

	if ( opts.overflow ) {
		style.overflow = "hidden";
		if ( !jQuery.support.shrinkWrapBlocks ) {
			anim.always(function() {
				style.overflow = opts.overflow[ 0 ];
				style.overflowX = opts.overflow[ 1 ];
				style.overflowY = opts.overflow[ 2 ];
			});
		}
	}
```

```
	// show/hide pass
	for ( prop in props ) {
		value = props[ prop ];
		if ( rfxtypes.exec( value ) ) {
			delete props[ prop ];
			toggle = toggle || value === "toggle";
			if ( value === ( hidden ? "hide" : "show" ) ) {
				continue;
			}
			orig[ prop ] = dataShow && dataShow[ prop ] ||
jQuery.style( elem, prop );
		}
	}

	if ( !jQuery.isEmptyObject( orig ) ) {
		if ( dataShow ) {
			if ( "hidden" in dataShow ) {
				hidden = dataShow.hidden;
			}
		} else {
			dataShow = jQuery._data( elem, "fxshow", {} );
		}

		// store state if its toggle - enables .stop().toggle() to "reverse"
		if ( toggle ) {
			dataShow.hidden = !hidden;
		}
		if ( hidden ) {
			jQuery( elem ).show();
		} else {
			anim.done(function() {
				jQuery( elem ).hide();
			});
		}
		anim.done(function() {
			var prop;
			jQuery._removeData( elem, "fxshow" );
			for ( prop in orig ) {
```

```
                        jQuery.style( elem, prop, orig[ prop ] );
                }
            });
            for ( prop in orig ) {
                tween = createTween( hidden ? dataShow[ prop ] : 0,
prop, anim );

                if ( !( prop in dataShow ) ) {
                    dataShow[ prop ] = tween.start;
                    if ( hidden ) {
                        tween.end = tween.start;
                        tween.start = prop === "width" || prop ===
"height" ? 1 : 0;
                    }
                }
            }
        }
}

function Tween( elem, options, prop, end, easing ) {
        return new Tween.prototype.init( elem, options, prop, end, easing );
}
jQuery.Tween = Tween;

Tween.prototype = {
        constructor: Tween,
        init: function( elem, options, prop, end, easing, unit ) {
            this.elem = elem;
            this.prop = prop;
            this.easing = easing || "swing";
            this.options = options;
            this.start = this.now = this.cur();
            this.end = end;
            this.unit = unit || ( jQuery.cssNumber[ prop ] ? "" : "px" );
        },
        cur: function() {
            var hooks = Tween.propHooks[ this.prop ];
```

```
		return hooks && hooks.get ?
			hooks.get( this ) :
			Tween.propHooks._default.get( this );
	},
	run: function( percent ) {
		var eased,
			hooks = Tween.propHooks[ this.prop ];

		if ( this.options.duration ) {
			this.pos = eased = jQuery.easing[ this.easing ](
				percent, this.options.duration * percent, 0, 1,
this.options.duration
			);
		} else {
			this.pos = eased = percent;
		}
		this.now = ( this.end - this.start ) * eased + this.start;

		if ( this.options.step ) {
			this.options.step.call( this.elem, this.now, this );
		}

		if ( hooks && hooks.set ) {
			hooks.set( this );
		} else {
			Tween.propHooks._default.set( this );
		}
		return this;
	}
};

Tween.prototype.init.prototype = Tween.prototype;

Tween.propHooks = {
	_default: {
		get: function( tween ) {
			var result;
```

```
                        if ( tween.elem[ tween.prop ] != null &&
                                (!tween.elem.style || tween.elem.style[ tween.prop ]
== null) ) {
                                return tween.elem[ tween.prop ];
                        }

                        // passing an empty string as a 3rd parameter to .css will
automatically
                        // attempt a parseFloat and fallback to a string if the parse
fails
                        // so, simple values such as "10px" are parsed to Float.
                        // complex values such as "rotate(1rad)" are returned as
is.
                        result = jQuery.css( tween.elem, tween.prop, "" );
                        // Empty strings, null, undefined and "auto" are converted
to 0.
                        return !result || result === "auto" ? 0 : result;
                },
                set: function( tween ) {
                        // use step hook for back compat - use cssHook if its there
- use .style if its
                        // available and use plain properties where available
                        if ( jQuery.fx.step[ tween.prop ] ) {
                                jQuery.fx.step[ tween.prop ]( tween );
                        } else if ( tween.elem.style && ( tween.elem.style[
jQuery.cssProps[ tween.prop ] ] != null || jQuery.cssHooks[ tween.prop ] ) )
{
                                jQuery.style( tween.elem, tween.prop, tween.now +
tween.unit );
                        } else {
                                tween.elem[ tween.prop ] = tween.now;
                        }
                }
        }
};

// Support: IE <=9
// Panic based approach to setting things on disconnected nodes
```

```
Tween.propHooks.scrollTop = Tween.propHooks.scrollLeft = {
	set: function( tween ) {
		if ( tween.elem.nodeType && tween.elem.parentNode ) {
			tween.elem[ tween.prop ] = tween.now;
		}
	}
};

jQuery.each([ "toggle", "show", "hide" ], function( i, name ) {
	var cssFn = jQuery.fn[ name ];
	jQuery.fn[ name ] = function( speed, easing, callback ) {
		return speed == null || typeof speed === "boolean" ?
			cssFn.apply( this, arguments ) :
			this.animate( genFx( name, true ), speed, easing, callback
);
	};
});

jQuery.fn.extend({
	fadeTo: function( speed, to, easing, callback ) {

		// show any hidden elements after setting opacity to 0
		return this.filter( isHidden ).css( "opacity", 0 ).show()

			// animate to the value specified
			.end().animate({ opacity: to }, speed, easing, callback );
	},
	animate: function( prop, speed, easing, callback ) {
		var empty = jQuery.isEmptyObject( prop ),
			optall = jQuery.speed( speed, easing, callback ),
			doAnimation = function() {
				// Operate on a copy of prop so per-property easing
won't be lost
				var anim = Animation( this, jQuery.extend( {}, prop
), optall );

				// Empty animations, or finishing resolves
```

```
immediately
                        if ( empty || jQuery._data( this, "finish" ) ) {
                                anim.stop( true );
                        }
                };
                doAnimation.finish = doAnimation;

        return empty || optall.queue === false ?
                this.each( doAnimation ) :
                this.queue( optall.queue, doAnimation );
    },
    stop: function( type, clearQueue, gotoEnd ) {
        var stopQueue = function( hooks ) {
                var stop = hooks.stop;
                delete hooks.stop;
                stop( gotoEnd );
        };

        if ( typeof type !== "string" ) {
                gotoEnd = clearQueue;
                clearQueue = type;
                type = undefined;
        }
        if ( clearQueue && type !== false ) {
                this.queue( type || "fx", [] );
        }

        return this.each(function() {
                var dequeue = true,
                        index = type != null && type + "queueHooks",
                        timers = jQuery.timers,
                        data = jQuery._data( this );

                if ( index ) {
                        if ( data[ index ] && data[ index ].stop ) {
                                stopQueue( data[ index ] );
                        }
                } else {
```

```
                        for ( index in data ) {
                                if ( data[ index ] && data[ index ].stop &&
rrun.test( index ) ) {
                                        stopQueue( data[ index ] );
                                }
                        }
                }

                for ( index = timers.length; index--; ) {
                        if ( timers[ index ].elem === this && (type == null
|| timers[ index ].queue === type) ) {
                                timers[ index ].anim.stop( gotoEnd );
                                dequeue = false;
                                timers.splice( index, 1 );
                        }
                }

                // start the next in the queue if the last step wasn't forced
                // timers currently will call their complete callbacks, which
will dequeue
                // but only if they were gotoEnd
                if ( dequeue || !gotoEnd ) {
                        jQuery.dequeue( this, type );
                }
        });
    },
    finish: function( type ) {
        if ( type !== false ) {
                type = type || "fx";
        }
        return this.each(function() {
                var index,
                        data = jQuery._data( this ),
                        queue = data[ type + "queue" ],
                        hooks = data[ type + "queueHooks" ],
                        timers = jQuery.timers,
                        length = queue ? queue.length : 0;
```

```
                        // enable finishing flag on private data
                        data.finish = true;

                        // empty the queue first
                        jQuery.queue( this, type, [] );

                        if ( hooks && hooks.stop ) {
                                hooks.stop.call( this, true );
                        }

                        // look for any active animations, and finish them
                        for ( index = timers.length; index--; ) {
                                if ( timers[ index ].elem === this && timers[ index
].queue === type ) {
                                        timers[ index ].anim.stop( true );
                                        timers.splice( index, 1 );
                                }
                        }

                        // look for any animations in the old queue and finish them
                        for ( index = 0; index < length; index++ ) {
                                if ( queue[ index ] && queue[ index ].finish ) {
                                        queue[ index ].finish.call( this );
                                }
                        }

                        // turn off finishing flag
                        delete data.finish;
                });
        }
});

// Generate parameters to create a standard animation
function genFx( type, includeWidth ) {
        var which,
                attrs = { height: type },
                i = 0;
```

```
	// if we include width, step value is 1 to do all cssExpand values,
	// if we don't include width, step value is 2 to skip over Left and Right
	includeWidth = includeWidth? 1 : 0;
	for( ; i < 4 ; i += 2 - includeWidth ) {
		which = cssExpand[ i ];
		attrs[ "margin" + which ] = attrs[ "padding" + which ] = type;
	}

	if ( includeWidth ) {
		attrs.opacity = attrs.width = type;
	}

	return attrs;
}

// Generate shortcuts for custom animations
jQuery.each({
	slideDown: genFx("show"),
	slideUp: genFx("hide"),
	slideToggle: genFx("toggle"),
	fadeIn: { opacity: "show" },
	fadeOut: { opacity: "hide" },
	fadeToggle: { opacity: "toggle" }
}, function( name, props ) {
	jQuery.fn[ name ] = function( speed, easing, callback ) {
		return this.animate( props, speed, easing, callback );
	};
});

jQuery.speed = function( speed, easing, fn ) {
	var opt = speed && typeof speed === "object" ? jQuery.extend( {},
speed ) : {
		complete: fn || !fn && easing ||
			jQuery.isFunction( speed ) && speed,
		duration: speed,
		easing: fn && easing || easing && !jQuery.isFunction( easing )
&& easing
	};
```

```
	opt.duration = jQuery.fx.off ? 0 : typeof opt.duration === "number" ? 
opt.duration :
		opt.duration in jQuery.fx.speeds ? jQuery.fx.speeds[ 
opt.duration ] : jQuery.fx.speeds._default;

	// normalize opt.queue - true/undefined/null -> "fx"
	if ( opt.queue == null || opt.queue === true ) {
		opt.queue = "fx";
	}

	// Queueing
	opt.old = opt.complete;

	opt.complete = function() {
		if ( jQuery.isFunction( opt.old ) ) {
			opt.old.call( this );
		}

		if ( opt.queue ) {
			jQuery.dequeue( this, opt.queue );
		}
	};

	return opt;
};

jQuery.easing = {
	linear: function( p ) {
		return p;
	},
	swing: function( p ) {
		return 0.5 - Math.cos( p*Math.PI ) / 2;
	}
};

jQuery.timers = [];
jQuery.fx = Tween.prototype.init;
```

```
jQuery.fx.tick = function() {
	var timer,
		timers = jQuery.timers,
		i = 0;

	fxNow = jQuery.now();

	for ( ; i < timers.length; i++ ) {
		timer = timers[ i ];
		// Checks the timer has not already been removed
		if ( !timer() && timers[ i ] === timer ) {
			timers.splice( i--, 1 );
		}
	}

	if ( !timers.length ) {
		jQuery.fx.stop();
	}
	fxNow = undefined;
};

jQuery.fx.timer = function( timer ) {
	if ( timer() && jQuery.timers.push( timer ) ) {
		jQuery.fx.start();
	}
};

jQuery.fx.interval = 13;

jQuery.fx.start = function() {
	if ( !timerId ) {
		timerId = setInterval( jQuery.fx.tick, jQuery.fx.interval );
	}
};

jQuery.fx.stop = function() {
	clearInterval( timerId );
	timerId = null;
```

```
};

jQuery.fx.speeds = {
	slow: 600,
	fast: 200,
	// Default speed
	_default: 400
};

// Back Compat <1.8 extension point
jQuery.fx.step = {};

if ( jQuery.expr && jQuery.expr.filters ) {
	jQuery.expr.filters.animated = function( elem ) {
		return jQuery.grep(jQuery.timers, function( fn ) {
			return elem === fn.elem;
		}).length;
	};
}
jQuery.fn.offset = function( options ) {
	if ( arguments.length ) {
		return options === undefined ?
			this :
			this.each(function( i ) {
				jQuery.offset.setOffset( this, options, i );
			});
	}

	var docElem, win,
		box = { top: 0, left: 0 },
		elem = this[ 0 ],
		doc = elem && elem.ownerDocument;

	if ( !doc ) {
		return;
	}

	docElem = doc.documentElement;
```

```
	// Make sure it's not a disconnected DOM node
	if ( !jQuery.contains( docElem, elem ) ) {
		return box;
	}

	// If we don't have gBCR, just use 0,0 rather than error
	// BlackBerry 5, iOS 3 (original iPhone)
	if ( typeof elem.getBoundingClientRect !== core_strundefined ) {
		box = elem.getBoundingClientRect();
	}
	win = getWindow( doc );
	return {
		top: box.top  + ( win.pageYOffset || docElem.scrollTop )  - (
docElem.clientTop  || 0 ),
		left: box.left + ( win.pageXOffset || docElem.scrollLeft ) - (
docElem.clientLeft || 0 )
	};
};

jQuery.offset = {

	setOffset: function( elem, options, i ) {
		var position = jQuery.css( elem, "position" );

		// set position first, in-case top/left are set even on static elem
		if ( position === "static" ) {
			elem.style.position = "relative";
		}

		var curElem = jQuery( elem ),
			curOffset = curElem.offset(),
			curCSSTop = jQuery.css( elem, "top" ),
			curCSSLeft = jQuery.css( elem, "left" ),
			calculatePosition = ( position === "absolute" || position
=== "fixed" ) && jQuery.inArray("auto", [curCSSTop, curCSSLeft]) > -1,
			props = {}, curPosition = {}, curTop, curLeft;
```

```
		// need to be able to calculate position if either top or left is auto
and position is either absolute or fixed
		if ( calculatePosition ) {
			curPosition = curElem.position();
			curTop = curPosition.top;
			curLeft = curPosition.left;
		} else {
			curTop = parseFloat( curCSSTop ) || 0;
			curLeft = parseFloat( curCSSLeft ) || 0;
		}

		if ( jQuery.isFunction( options ) ) {
			options = options.call( elem, i, curOffset );
		}

		if ( options.top != null ) {
			props.top = ( options.top - curOffset.top ) + curTop;
		}
		if ( options.left != null ) {
			props.left = ( options.left - curOffset.left ) + curLeft;
		}

		if ( "using" in options ) {
			options.using.call( elem, props );
		} else {
			curElem.css( props );
		}
	}
};


jQuery.fn.extend({

	position: function() {
		if ( !this[ 0 ] ) {
			return;
		}
```

```
		var offsetParent, offset,
			parentOffset = { top: 0, left: 0 },
			elem = this[ 0 ];

		// fixed elements are offset from window (parentOffset = {top:0,
left: 0}, because it is it's only offset parent
		if ( jQuery.css( elem, "position" ) === "fixed" ) {
			// we assume that getBoundingClientRect is available when
computed position is fixed
			offset = elem.getBoundingClientRect();
		} else {
			// Get *real* offsetParent
			offsetParent = this.offsetParent();

			// Get correct offsets
			offset = this.offset();
			if ( !jQuery.nodeName( offsetParent[ 0 ], "html" ) ) {
				parentOffset = offsetParent.offset();
			}

			// Add offsetParent borders
			parentOffset.top  += jQuery.css( offsetParent[ 0 ],
"borderTopWidth", true );
			parentOffset.left += jQuery.css( offsetParent[ 0 ],
"borderLeftWidth", true );
		}

		// Subtract parent offsets and element margins
		// note: when an element has margin: auto the offsetLeft and
marginLeft
		// are the same in Safari causing offset.left to incorrectly be 0
		return {
			top:  offset.top  - parentOffset.top - jQuery.css( elem,
"marginTop", true ),
			left: offset.left - parentOffset.left - jQuery.css( elem,
"marginLeft", true)
		};
	},
```

```
	offsetParent: function() {
		return this.map(function() {
			var offsetParent = this.offsetParent || docElem;
			while ( offsetParent && ( !jQuery.nodeName( offsetParent,
"html" ) && jQuery.css( offsetParent, "position") === "static" ) ) {
				offsetParent = offsetParent.offsetParent;
			}
			return offsetParent || docElem;
		});
	}
});


// Create scrollLeft and scrollTop methods
jQuery.each( {scrollLeft: "pageXOffset", scrollTop: "pageYOffset"}, function(
method, prop ) {
	var top = /Y/.test( prop );

	jQuery.fn[ method ] = function( val ) {
		return jQuery.access( this, function( elem, method, val ) {
			var win = getWindow( elem );

			if ( val === undefined ) {
				return win ? (prop in win) ? win[ prop ] :
					win.document.documentElement[ method ] :
					elem[ method ];
			}

			if ( win ) {
				win.scrollTo(
					!top ? val : jQuery( win ).scrollLeft(),
					top ? val : jQuery( win ).scrollTop()
				);

			} else {
				elem[ method ] = val;
			}
```

```
		}, method, val, arguments.length, null );
	};
});

function getWindow( elem ) {
	return jQuery.isWindow( elem ) ?
		elem :
		elem.nodeType === 9 ?
			elem.defaultView || elem.parentWindow :
			false;
}
// Create innerHeight, innerWidth, height, width, outerHeight and
outerWidth methods
jQuery.each( { Height: "height", Width: "width" }, function( name, type ) {
	jQuery.each( { padding: "inner" + name, content: type, "": "outer" +
name }, function( defaultExtra, funcName ) {
		// margin is only for outerHeight, outerWidth
		jQuery.fn[ funcName ] = function( margin, value ) {
			var chainable = arguments.length && ( defaultExtra ||
typeof margin !== "boolean" ),
				extra = defaultExtra || ( margin === true || value
=== true ? "margin" : "border" );

			return jQuery.access( this, function( elem, type, value ) {
				var doc;

				if ( jQuery.isWindow( elem ) ) {
					// As of 5/8/2012 this will yield incorrect
results for Mobile Safari, but there
					// isn't a whole lot we can do. See pull request
at this URL for discussion:
					// https://github.com/jquery/jquery/pull/764
					return elem.document.documentElement[
"client" + name ];
				}

				// Get document width or height
				if ( elem.nodeType === 9 ) {
```

```
                                doc = elem.documentElement;

                                // Either scroll[Width/Height] or
offset[Width/Height] or client[Width/Height], whichever is greatest
                                // unfortunately, this causes bug #3838 in
IE6/8 only, but there is currently no good, small way to fix it.
                                return Math.max(
                                        elem.body[ "scroll" + name ], doc[
"scroll" + name ],
                                        elem.body[ "offset" + name ], doc[
"offset" + name ],
                                        doc[ "client" + name ]
                                );
                        }

                        return value === undefined ?
                                // Get width or height on the element,
requesting but not forcing parseFloat
                                jQuery.css( elem, type, extra ) :

                                // Set width or height on the element
                                jQuery.style( elem, type, value, extra );
                }, type, chainable ? margin : undefined, chainable, null );
        };
    });
});
// Limit scope pollution from any deprecated API
// (function() {

// The number of elements contained in the matched element set
jQuery.fn.size = function() {
    return this.length;
};

jQuery.fn.andSelf = jQuery.fn.addBack;

// })();
if ( typeof module === "object" && module && typeof module.exports ===
```

```
"object" ) {
	// Expose jQuery as module.exports in loaders that implement the
Node
	// module pattern (including browserify). Do not create the global,
since
	// the user will be storing it themselves locally, and globals are
frowned
	// upon in the Node module world.
	module.exports = jQuery;
} else {
	// Otherwise expose jQuery to the global object as usual
	window.jQuery = window.$ = jQuery;

	// Register as a named AMD module, since jQuery can be
concatenated with other
	// files that may use define, but not via a proper concatenation script
that
	// understands anonymous AMD modules. A named AMD is safest and
most robust
	// way to register. Lowercase jquery is used because AMD module
names are
	// derived from file names, and jQuery is normally delivered in a
lowercase
	// file name. Do this after creating the global so that if an AMD module
wants
	// to call noConflict to hide this version of jQuery, it will work.
	if ( typeof define === "function" && define.amd ) {
		define( "jquery", [], function () { return jQuery; } );
	}
}

})( window );
```

Proses selanjutnya kita pembuatan tampilan pada web tersebut dibawah ini gambar 3.4 pembuatan difolder view

cache	2/16/2015 10:38 PM	File folder	
config	2/16/2015 10:38 PM	File folder	
controllers	2/3/2015 8:30 PM	File folder	
core	2/3/2015 8:30 PM	File folder	
errors	2/3/2015 8:30 PM	File folder	
helpers	2/3/2015 8:30 PM	File folder	
hooks	2/3/2015 8:30 PM	File folder	
language	2/3/2015 8:30 PM	File folder	
libraries	2/3/2015 8:30 PM	File folder	
logs	2/3/2015 8:30 PM	File folder	
models	2/11/2015 6:34 PM	File folder	
third_party	2/3/2015 8:30 PM	File folder	
views	2/16/2015 12:40 AM	File folder	
.htaccess	2/16/2015 3:37 PM	HTACCESS File	1 KB
index	7/29/2013 3:54 PM	HTML Document	1 KB

Gambar 4.4

Nama File hader.php file tersebut pembuatan tampilan menu

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>*Mimi'Cake*</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"
/>
<link href="<?php echo base_url('assets/css/style.css');?>" rel="stylesheet"
type="text/css" />
<link href="<?php echo base_url('assets/css/menu.css');?>"
rel="stylesheet" type="text/css" />
<link rel="stylesheet" href="<?php echo base_url('assets/css/jquery-
ui.css');?>">
  <script src="<?php echo base_url('assets/js/jquery-
1.10.2.js');?>"></script>
  <script src="<?php echo base_url('assets/js/jquery-ui.js');?>"></script>
  <link rel="stylesheet" href="/resources/demos/style.css">
```

```
  <script>
  $(function() {
     $("#datepicker").datepicker({ dateFormat: "yy-mm-dd" }).val()
     $("#datepicker1").datepicker({ dateFormat: "yy-mm-dd" }).val()

  });
    function biaya_kirim(){
       var  biaya=+document.getElementById("cmb-biaya").value;
        document.getElementById("txt_biaya").value=biaya;

      var total_harga=+document.getElementById("total_harga").value;
      var   jum = total_harga+biaya;
        document.getElementById("txt_total").value=jum;
    }
</script>
</head>
<body>
<div id="container">
  <div id="top">
  <center>
     <MARQUEE align="center" direction="left" height="20" scrollamount="4"
width="68%" behavior="alternate"style="margin-left: 12%;">
    <h1>Toko Roti Online Mimi'Cake</h1>
        <center> <h1>Aneka Roti</h1></center>
    </MARQUEE>
        </center>
  </div>
  <div id="menuh-container">
    <div id="menuh">
      <ul>
        <li><a href="<?php echo base_url(); ?>">BERANDA</a>
        </li>
      </ul>
      <ul>
        <li><a href="<?php echo base_url(); ?>index.php/roti/produk"
class="top_parent"> PRODUK</a>
        </li>
      </ul>
```

```
    <ul>
      <li><a href="<?php echo base_url(); ?>index.php/cart/"
>KERANJANG BELANJA</a> </li>
    </ul>
    <ul>
      <li><a href="<?php echo base_url();
?>index.php/konfirmasi_pembayaran/" > KONFIRMASI</a> </li>
    </ul>
    <ul>
      <li><a href="<?php echo base_url(); ?>index.php/cara_pesan/">
CARA PESAN</a></li>
    </ul>

    <ul>
      <li><a href="<?php echo base_url();
?>index.php/about_us/">TENTANG KAMI</a> </li>
    </ul>
   </div>
  </div>
  <div class="clear:both;"></div>
```

Dibawah ini Output Tampilan Pertama menunya

Gambar 1.6

Pembuatannya fother dan hadernya sebgai berikut ini sorce codenya sebagai berikut ini:

File footer.php

```
<div id="footer"><center> @toko roti mimi'cake
*085729674656*</center></div>
<div id="footer"><center>kemasan widodomartani ngemplak sleman
yogyakarta 55571</center></div>
</div>
</div>
</body>
</html>
```

Tampilan outputnya sebagai berikut ini:

Gambar 1.7

Pembuatan hadernya sorce codenya sebagai berikut ini:

```
<?php include('header.php');?>
        <?php echo $status; ?>

<?php include('footer.php');?>
```

Pembuatan Tampilan branda sorce codenya sebagau berikut ini:

Dengan nama file welcome_message.php

```
<?php include('header.php');?>
<?php include('nav-left.php');?>

        <div id="content">
                <div class="content-center">
                        <div class="sambutan">
                                <h3>selamat datang di Mimi cake</h3>
                                <p>Hallo, selamat datang di mimi cake toko roti
```

```
online yang menyediakan berbagai produk roti kami siap melayani pesanan
partai kecil dan partai besar baik acara pesta, meeting, seminar dll.</p>

                    </div>
                    <?php foreach ($roti as $row) { ?>
                    <div class='content-home'>
                          <div class="nama-produk">
                                <a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"> <?php
echo $row->nama_produk; ?></a>
                          </div>
                          <div class='img-home'>
                                <a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"><img
src="<?php echo base_url('assets/images/roti/'.$row->gambar);?>"></a>
                                <div class="img-promo"><img src="<?php
echo base_url('assets/images/promo.png');?>"></div>
                          </div>
                          <div class="harga">Rp. <?php echo
number_format($row ->harga_produk,2); ?> </div>
                          <div class="stok">stok <?php echo $row->stok;
?></div>
                          <div class="kadaluarsa">Kadaluarsa <?php echo
$row->kadaluwarsa; ?> Hari </div>
                          <a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"><div
class="btn-detail">Detail</div></a>
                          <form
action="<?=base_url();?>index.php/cart/add" method="post">
                                <input type="hidden" name="id"
value="<?php echo $row->id_produk; ?>">
                                <input type="hidden" name="name"
value="<?php echo $row->nama_produk; ?>">
                                <input type="hidden" name="price"
value="<?php echo $row->harga_produk; ?>">
                                <div id="btn-beli"><input type="submit"
name="beli" value="Beli" /></div>
                          </form>
```

```
                    </div>
                    <?php } ?>
            </div>
            <div class="list-pagination">
            </div>
        </div>
<?php include('footer.php');?>
```

Dibawah ini Tampian Output brandanya

Gambar 1.8

Berikut ini pembuatan produknya dan katagori serta detailnya untuk saling berhubungan yang pertama kita membuatnya sebuah controllernya untuk menghubungakan controller ke view sebagai berikut ini kita pembuatan controllernya dan modelnya kita membuat modelnya agar dapat memanggil sebuah data basenya yang dari admin unggah data produknya berikut ini sorce codenya:

File roti.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Roti extends CI_Controller {
        function __construct()
        {
          parent::__construct();
```

```
            $this->load->model('admin/model_roti');
        }

        public function detail_roti($id)
        {
        $data['detailroti'] = $this->model_roti->get_detail_roti($id)->row();
        $data['katagori'] = $this->model_roti->get_katagori()->result();
        $this->load->view('halaman_detail_roti',$data);
        }

        public function katagori_roti($id)
        {
        $data['katagori_roti'] = $this->model_roti->get_katagori_roti($id)-
>result();
        $data['katagori'] = $this->model_roti->get_katagori()->result();
        $this->load->view('halaman_katagori_roti',$data);
        }
        public function produk()
        {
        $this->load->library('pagination');
            $config['base_url'] = site_url('roti/produk');
            $config['total_rows'] = $this->model_roti->get_num_rows();
            $config['per_page']=$per_page=12;
            $config['uri_segment'] = 3;
            $this->pagination->initialize($config);
            $data['paging']=$this->pagination->create_links();
            $page=($this->uri->segment(3)) ? $this->uri->segment(3):0;

            $data['roti'] = $this->model_roti->get_roti($per_page,$page)-
>result();
            $data['katagori'] = $this->model_roti->get_katagori()-
>result();
            $this->load->view('halaman_produk',$data);
        }


}
```

Proses selanjutnya file diview berikut ini sorce code halaman_produk.php

```
<?php include('header.php');?>
<?php include('nav-left.php');?>

	<div id="content">
		<div class="content-center">

			<?php foreach ($roti as $row) { ?>
			<div class='content-home'>
				<div class="nama-produk">
					<a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"> <?php
echo $row->nama_produk; ?></a>
				</div>
				<div class='img-home'>
					<a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"><img
src="<?php echo base_url('assets/images/roti/'.$row->gambar);?>"></a>
				</div>
				<div class="harga">Rp. <?php echo
number_format($row ->harga_produk,2); ?> </div>
				<div class="stok">stok <?php echo $row->stok;
?></div>
				<div class="kadaluarsa">Kadaluarsa <?php echo
$row->kadaluwarsa; ?> <spasi>  Hari</div>
				<a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"><div
class="btn-detail">Detail</div></a>
				<form
action="<?=base_url();?>index.php/cart/add" method="post">
					<input type="hidden" name="id"
value="<?php echo $row->id_produk; ?>">
					<input type="hidden" name="name"
value="<?php echo $row->nama_produk; ?>">
					<input type="hidden" name="price"
value="<?php echo $row->harga_produk; ?>">
					<div id="btn-beli"><input type="submit"
```

```
name="beli" value="Beli" /></div>
                        </form>
                </div>
                <?php } ?>
        </div>
        <div class="list-pagination">
        <?php echo $paging; ?>
        </div>
    </div>
<?php include('footer.php');?>
```

Proses seanjunya kita pembuatan katagorinya sebagai berikut ini:

```
<?php include('header.php');?>
<?php include('nav-left.php');?>

    <div id="content">
        <div class="content-center">
            <?php foreach ($katagori_roti as $row) { ?>
            <div class='content-home'>
                <div class="nama-produk">
                    <a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"> <?php
echo $row->nama_produk; ?></a>
                </div>
                <div class='img-home'>
                    <a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"><img
src="<?php echo base_url('assets/images/roti/'.$row->gambar);?>"></a>
                </div>
                <div class="harga">Rp. <?php echo
number_format($row ->harga_produk,2); ?> </div>
                <div class="stok">stok <?php echo $row->stok;
?></div>
                <div class="kadaluarsa">Kadaluarsa <?php echo
$row->kadaluwarsa; ?> Hari </div>
                <a href="<?php echo base_url();
?>index.php/roti/detail_roti/<?php echo $row->id_produk; ?>"><div
```

```
class="btn-detail">Detail</div></a>
                            <form
action="<?=base_url();?>index.php/cart/add" method="post">
                                    <input type="hidden" name="id"
value="<?php echo $row->id_produk; ?>">
                                    <input type="hidden" name="name"
value="<?php echo $row->nama_produk; ?>">
                                    <input type="hidden" name="price"
value="<?php echo $row->harga_produk; ?>">
                                    <div id="btn-beli"><input type="submit"
name="beli" value="Beli" /></div>
                            </form>
                    </div>
                    <?php } ?>
            </div>
            <div class="list-pagination">
            </div>
        </div>
<?php include('footer.php');?>
```

Proses selanjunya pembuatan detailnya degan nama file

detail_roti.php

```
<?php include('header.php');?>
<?php include('nav-left.php');?>

        <div id="content">
            <div class='content-detail'>
                    <div class="nama-produk-detail">
                            <?php echo $detailroti ->nama_produk; ?>
                    </div>
                    <div class='img-detail'>
                            <center><img src="<?php echo
base_url('assets/images/roti/'.$detailroti->gambar);?>"></center>
                    </div>
                    <div class="harga">Rp. <?php echo
number_format($detailroti ->harga_produk,2); ?> </div>
```

```
                    <div class="stok">stok <?php echo $detailroti->stok;
?></div>
                    <div class="kadaluarsa">Kadaluarsa <?php echo
$detailroti->kadaluwarsa; ?> Hari</div>
                    <div class="keterangan"><?php echo $detailroti -
>keterangan; ?></div>
                    <form action="<?=base_url();?>index.php/cart/add"
method="post">
                        <input type="hidden" name="id" value="<?php
echo $detailroti->id_produk; ?>">
                        <input type="hidden" name="name" value="<?php
echo $detailroti->nama_produk; ?>">
                        <input type="hidden" name="price" value="<?php
echo $detailroti->harga_produk; ?>">
                        <center><input type="submit" name="login"
value="Tambahkan kepesanan" /></center>
                    </form>
            </div>
        </div>
<?php include('footer.php');?>
```

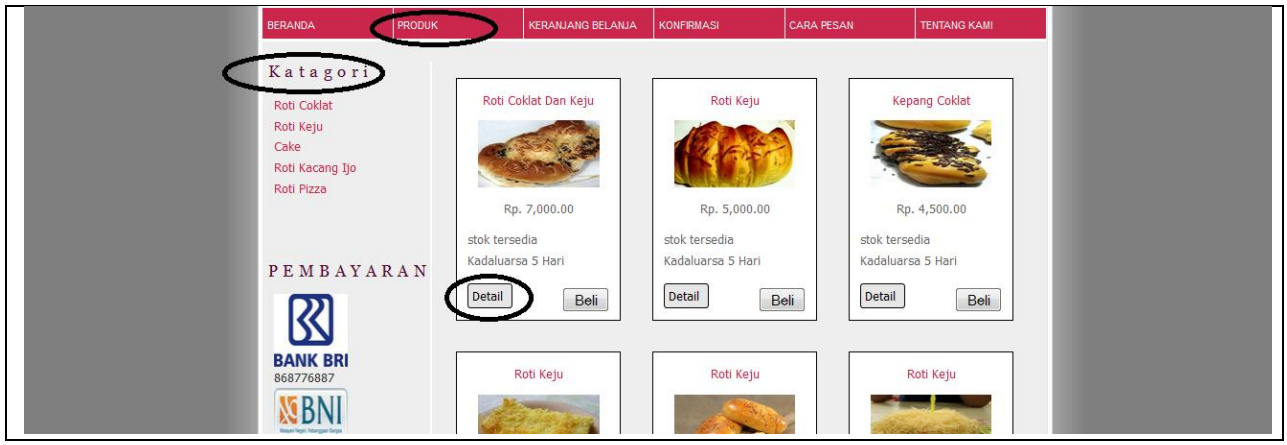
File modelnya sebagai berikut ini:

Dengan nama file Model_roti.php

```
<?php  if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_roti extends CI_Model
{
public function get_roti($limit,$start){
        return $this->db->get('produk_roti',$limit,$start);
}
public function get_roti_promo($limit,$start){
        $this->db->where("promo","promo");
        return $this->db->get('produk_roti',$limit,$start);
}
public function get_katagori(){
return $this->db->get('katagori');
```

```
}
public function get_num_rows()
{
return $this->db->get('produk_roti')->num_rows();
}
public function get_detail_roti($id){
return $this->db->get_where('produk_roti', array('id_produk' => $id));
}

public function update_roti($data,$id){
$this->db->update("produk_roti",$data,$id);
}
public function delete_roti($id){
$this->db->where('id_produk', $id);
$this->db->delete('produk_roti');
}

public function insert_roti($data){
$this->db->insert('produk_roti', $data);
}

public function get_katagori_roti($id){
return $this->db->get_where('produk_roti', array('id_katagori' => $id));
}

}
```

Berikut ini tampilan produk roti, serta tampilan detail dan katagorinya:



Gambar 1.9

Tampilan detailnya

Gambar 1.10

Pembuatan pada tampilan pembayarannya sorce codenya sebagai berikut ini:

```
<div id="leftnav">
   <h2>Katagori</h2>
   <div id="nav">
     <ul id="navlist">
       <?php foreach ($katagori as $row) { ?>
       <li><a href="<?php echo base_url();
?>index.php/roti/katagori_roti/<?php echo $row->id_katagori; ?>"> <?php
echo $row->jenis_katagori; ?></a></li>
       <?php } ?>
     </ul>
```

```
    </div>

    <div id="pembayaran-left">
        <h2>PEMBAYARAN</h2>
        <ul>
                <li><img src="<?php echo
base_url('assets/images/icon/bri.png');?>"><br>868776887</br></li>
                <li><img src="<?php echo
base_url('assets/images/icon/bni.png');?>"><br>098988989</br></li>
        </ul>
    </div>

</div>
```

Tampilan outputnya sebagai berikut ini

Gambar 1.10

Proses selanjutnya pembuatan dalam keranjang belanja sebagai berikut ini cara pembuatan keranjang belanja sebagai berikut ini cara pembuatan kereanjang belanja yang pertama kita bikin sebuah controllernya sebagai berikut ini pembuatannya controllernya:

Dengan nama file controllernya cart.php

```
<?php if (!defined('BASEPATH')) exit('No direct script access allowed');
```

```
class Cart extends CI_Controller {

	public function __construct()
	{
		parent::__construct();
		$this->load->model('Cart_model'); // memanggil model dengan
nama file cart_model
	}
// program untuk menampilkan pesan keranjang masih kosong
	public function index()
	{
		$this->data['title'] = 'Shopping Carts'; // nama deklarasi
shopping cart

		if (!$this->cart->contents()){  //Fungsi ini mengembalikan array
pada keranjang belanja
			$this->data['message'] = '<p>keranjang belanja masih
kosong</p>'; // proses untuk mengirikan keranjang belanja kosong
		}else{
			$this->data['message'] = $this->session-
>flashdata('message');  // proses pengiriman pesan
		}

		$this->load->view('cart', $this->data); // proses pemanggilan
pada folder view dengan nama file cart
	}

	public function add() //proses fungsi untu membuat penambahan
	{
		$this->load->model('Cart_model');

		$insert_room = array(
			'id' => $this->input->post('id'),
			'name' => $this->input->post('name'),
			'price' => $this->input->post('price'),
			'qty' => 1
		);
		//pada baris 29 s/d 32, untuk menjadikan inputan tersebut
```

```
kedalam bentuk array jika bernilai 1.

            $this->cart->insert($insert_room); //proses untuk penambahan
pada keranjang belanja

            redirect('cart'); // skrip untuk mengarahkan (redirect) kepada
posisi keranjang belanja.
        }

        function remove($rowid){
        if ($rowid == "all"){
        $this->cart->destroy();
        }else{
        $data = array(// variabel data untuk pemanggilan array menggunakan
array
        'rowid' => $rowid,
        'qty' => 0
        );
        $this->cart->update($data);
        }
        redirect('cart');
        }
// program untuk update
        function update_cart(){
            foreach($_POST['cart'] as $id => $cart) //membuat variabel cart
yang akan
            {
                $price = $cart['price']; // harga = akan ditampung pada
filed kernajang belanja dan membuat variabel harga.
                $amount = $price * $cart['qty']; // perkalian jumlah =
harga akan dikali jumlah yang akan dipesan maka akan diporses dikeranjang
belanja.

                $this->Cart_model->update_cart($cart['rowid'],
$cart['qty'], $price, $amount); // fungsi akan dikembalikan pada cart_model
            }

            redirect('cart'); //untuk kembali ke cart
```

```
	}
}
```

Proses selanjuntya pembuatan modelnya dengan namanya cart_model.php

```
<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

class Cart_model extends CI_Model {

	public function __construct()
	{
		//$this->load->database();
	}

	function update_cart($rowid, $qty, $price, $amount) {
		$data = array(
			'rowid'   => $rowid,
			'qty'     => $qty,
			'price'   => $price,
			'amount'   => $amount
		);
// untuk menggunkan fungsi update  yang sudah dibuat
		$this->cart->update($data);
	}

}
```

Proses selanjutnya pembuatan viewnya cart sebagai berikut ini

```
<?php include('header.php');?>
<script>
function clear_cart() {
	var result = confirm('Hapus semua pesanan?');

	if(result) {
		window.location = "<?php echo base_url();
?>index.php/cart/remove/all";
	}else{
```

```
                return false; // cancel button
        }
}
</script>

<div >
        <div style="padding-bottom:10px">
                <h1 align="center">Keranjang Belanja</h1>
                <input type="button" value="Belanja Lagi"
onclick="window.location='<?=base_url();?>'" />
        </div>
        <div style="color:#F00"><?php echo $message?></div>
        <table border="0" cellpadding="5px" cellspacing="1px" style="font-
family:Verdana, Geneva, sans-serif; font-size:11px; background-
color:#E1E1E1" width="100%">
                <?php if ($cart = $this->cart->contents()): ?>
                <tr bgcolor="#FFFFFF" style="font-weight:bold">
                        <td>No</td>
                        <td>Nama Produk</td>
                        <td>Harga</td>
                        <td>Jumlah</td>
                        <td>Total</td>
                        <td>Options</td>
                </tr>
                <?php
                echo form_open('cart/update_cart');
                $grand_total = 0; $i = 1;

                foreach ($cart as $item):
                        echo form_hidden('cart['. $item['id'] .'][id]', $item['id']);
                        echo form_hidden('cart['. $item['id'] .'][rowid]',
$item['rowid']);
                        echo form_hidden('cart['. $item['id'] .'][name]',
$item['name']);
                        echo form_hidden('cart['. $item['id'] .'][price]',
$item['price']);
                        echo form_hidden('cart['. $item['id'] .'][qty]',
$item['qty']);
```

```
            ?>
            <tr bgcolor="#FFFFFF">
                <td>
                    <?php echo $i++; ?>
                </td>
                <td>
                    <?php echo $item['name']; ?>
                </td>
                <td>
                    Rp. <?php echo number_format($item['price'],2);
?>
                </td>
                <td>
                    <?php echo form_input('cart['. $item['id'] .'][qty]',
$item['qty'], 'maxlength="3" size="1" style="text-align: right"'); ?>
                </td>
                <?php $grand_total = $grand_total + $item['subtotal'];
?>
                <td>
                    Rp. <?php echo number_format($item['subtotal'],2)
?>
                </td>
                <td>
                    <?php echo
anchor('cart/remove/'.$item['rowid'],'Cancel'); ?>
                </td>
                <?php endforeach; ?>
            </tr>
            <tr>
                <td><b>Total pesan: Rp. <?php echo
number_format($grand_total,2); ?></b></td>
                <td colspan="5" align="right"><input type="button"
value="Hapus semua" onclick="clear_cart()">
                        <input type="submit" value="Update Cart">
                        <?php echo form_close(); ?>
                        <input type="button" value="Selesai"
onclick="window.location='billing'"></td>
            </tr>
```

```
            <?php endif; ?>
        </table>
</div>
<?php include('footer.php');?>
```

Tampilan Outputnya sebagai berikut ini:

Gambar 1.11

Pembuatan info pembayaran yang pertama kita pembuatan controllernya sebagai berikut ini setelah itu kita bikin view dan modelnya sebagai berikut ini.

File controllernya dengan nama sebagai berikut ini:

File billing.php

```
<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

class Billing extends CI_Controller {

        public function __construct()
        {
                parent::__construct();
```

```
		$this->load->model('Billing_model');
	}

	public function index()
	{

		$data['biaya'] = $this->Billing_model->get_biaya()->result();
		$this->load->view('billing', $data);
	}

	public function save_order()
	{
		$this->form_validation->set_rules('name','name','required');
		$this->form_validation-
>set_rules('email','email','required|email');
		$this->form_validation-
>set_rules('address','address','required');
		$this->form_validation-
>set_rules('phone','phone','required|numeric');
		$this->form_validation-
>set_rules('alamat_pengiriman','alamat_pengiriman','required');
		$this->form_validation-
>set_rules('biaya_pengiriman','biaya_pengiriman','required');
		if($this->form_validation->run()==false){
			$data['biaya']	=	$this->Billing_model->get_biaya()-
>result();
			$this->load->view('billing', $data);
		}
		else{
			$pelanggan = array(
				'nama_pelanggan'		=>		$this->input-
>post('name'),
				'email'	=> $this->input->post('email'),
				'alamat'	=> $this->input->post('address'),
				'no_telephone'	=> $this->input->post('phone')
			);
```

```
                    $id_pelanggan              =              $this->Billing_model-
>insert_pelanggan($pelanggan);

                    $order = array(
                          'tgl_pesan'                => date('Y-m-d'),
                          'id_pelanggan'     => $id_pelanggan,
                          'total_harga' => $this->input->post('total_harga'),
                          'metode_pembayaran'          =>          $this->input-
>post('metode_pembayaran'),
                          'alamat_pengiriman'           =>          $this->input-
>post('alamat_pengiriman'),
                          'biaya_pengiriman'            =>          $this->input-
>post('biaya_pengiriman'),
                          'tgl_penerimaan'              =>          $this->input-
>post('tgl_penerimaan'),
                          'status_pembayaran' => "belum lunas",
                    );

                    $ord_id = $this->Billing_model->insert_order($order);

                    if ($cart = $this->cart->contents()):
                          foreach ($cart as $item):
                                $order_detail = array(
                                      'no_pesan'          => $ord_id,
                                      'kd_roti'     => $item['id'],
                                      'jumlah_roti'               => $item['qty'],
                                      'harga'               => $item['price'],
                                      'total_harga'
=>$item['qty']*$item['price'],
                                );

                                $id_pelanggan      =      $this->Billing_model-
>insert_order_detail($order_detail);
                          endforeach;
                    endif;
                    $nopesan=$ord_id;
                    $data['detailpesan']           =           $this->Billing_model-
>get_detail_pesan($nopesan)->row();
```

```
                    $data['pesan']=$ord_id;

                    $this->load->view('detail_pesan',$data);
            }
        }
}
```

Proses selanjutnya pembuatan modelnya

Dengan nama file billing_model.php

```
<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

class Billing_model extends CI_Model {

        public function __construct()
        {
                //$this->load->database();
        }

        public function insert_pelanggan($data)
        {
                $this->db->insert('pelanggan', $data);

                $id = $this->db->insert_id();

                return (isset($id)) ? $id : FALSE;
        }

        public function insert_order($data)
        {
                $this->db->insert('pesan', $data);

                $id = $this->db->insert_id();

                return (isset($id)) ? $id : FALSE;
        }
```

```
        public function insert_order_detail($data)
        {
                $this->db->insert('detail_pesan', $data);
        }
        public function get_biaya(){
                return $this->db->get('biaya');
        }
        public function get_detail_pesan($id){
                return    $this->db->get_where('pesan',    array('no_pesan'    =>
$id));
        }
}
```

Proses selanjutnya pembuatan pada viewnya sebagai berikut ini:

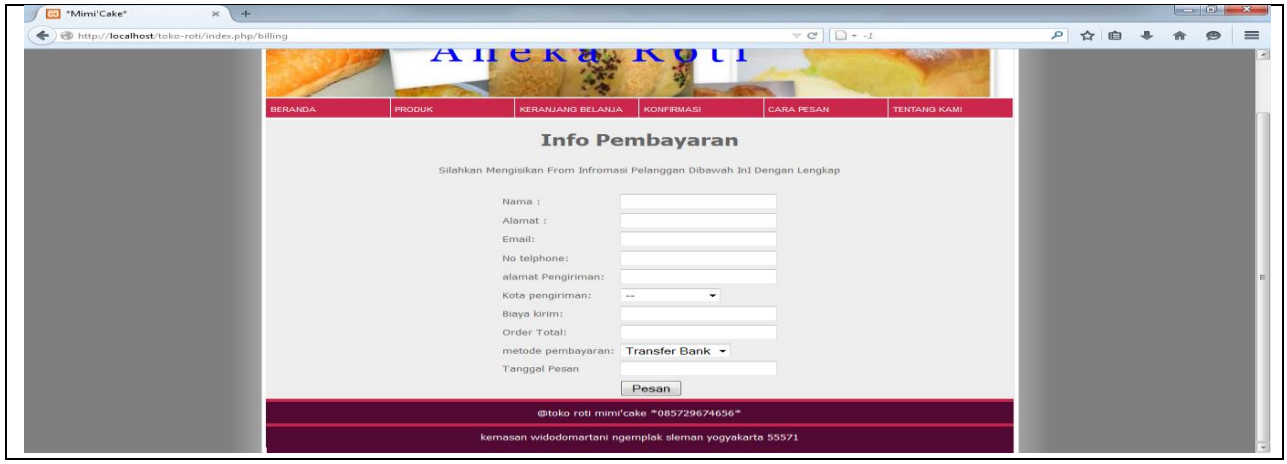
Dengan nama filenya billing.php

```
<?php
$grand_total = 0;

if ($cart = $this->cart->contents()):
        foreach ($cart as $item):
                $grand_total = $grand_total + $item['subtotal'];
        endforeach;
endif;
?>

<?php include('header.php');?>

<form        name="billing"        method="post"        action="<?php        echo
base_url().'index.php/billing/save_order' ?>" >
    <input type="hidden" name="command" />
        <div align="center">
        <h1 align="center">Info Pembayaran</h1>
        <center>
        <p>Silahkan  Mengisikan  From  Infromasi  Pelanggan  Dibawah  InI
Dengan Lengkap</p>
        </center>
```

```
      <table border="0" cellpadding="2px">
       <input        type="hidden"        id="total_harga"name="total_harga"
value="<?php echo $grand_total; ?>" readonly="readonly"/>
         <tr>
            <td>Nama :</td>
            <td>
               <input type="text" name="name" />
               <?php                echo                form_error('name','<div
class="error">','</div>');?>
            </td>
         </tr>
         <tr><td>Alamat  :</td><td><input  type="text"  name="address"
/><?php                      echo                      form_error('address','<div
class="error">','</div>');?></td></tr>
         <tr><td>Email:</td><td><input     type="text"     name="email"
<?php                        echo                        form_error('email','<div
class="error">','</div>');?></td></tr>
         <tr><td>No          telphone:</td><td><input          type="text"
name="phone"/><?php              echo              form_error('phone','<div
class="error">','</div>');?></td></tr>
         <tr><td>alamat      Pengiriman:</td><td><input      type="text"
name="alamat_pengiriman"                    /><?php                    echo
form_error('alamat_pengiriman','<div
class="error">','</div>');?></td></tr>
         <tr><td>Kota                          pengiriman:</td><td><select
name="biaya_pengiriman" id="cmb-biaya"onchange="biaya_kirim()">
         <option value="">--</option>
           <?php foreach ($biaya as $row) { ?>
         <option value="<?php echo $row->biaya; ?>"><?php echo $row-
>kota; ?></option>
         <?php } ?>
         </select></td><?php    echo    form_error('kota_pengiriman','<div
class="error">','</div>');?></tr>
         <tr><td>Biaya          kirim:</td><td><input          type="text"
name="biayakirim"     id="txt_biaya"     value=""     readonly="readonly"
/></td></tr>
         <tr><td>Order          Total:</td><td><input          type="text"
```

```
name="total_harga"                    id="txt_total"                    value=""
readonly="readonly"/></td></tr>
        <tr><td>metode                    pembayaran:</td><td><select
name="metode_pembayaran">
            <option value="transfer">Transfer Bank</option>
            <option value="cod">COD</option>
        </select></td></tr>
        <tr>
            <td>Tanggal Pesan</td>
            <td><input              type="text"              id="datepicker1"
name="tgl_penerimaan"><?php      echo      form_error('tgl_transfer','<div
class="error">','</div>');?></td>
        </tr>

        <tr><td> </td><td><input   type="submit"   value="Pesan"
/></td></tr>
      </table>
      </div>
</form>

<?php include('footer.php');?>
```

Maka Tampilannya sebagai berikut ini:



Gambar 1.12

Setelah semua di isikan maka proses selanjutnya akan ditampilkan ke produk detailnya sebagai berikut yang berada dalam folder viewnya

Sorce codenya sebagai berikut ini

```
<?php include('header.php');?>
<center>detail Pesan</center>
terima kasih telah pesan roti di toko kami,berikut detail pesanan anda<br>
Waktu pengiriman disesuaikan dengan tanggal dan waktu yang ditentukan pelanggan pada saat memesan.<br>
Dengan batas waktu dari pukul 07.00 pagi 17.00 sore jika pemesanan lebih dari jam kerja maka pengiriman akan dilaksanankan pada hari berikutnya.<br>
No transaksi anda  <?php echo $pesan; ?> dengan detail barang sebagai berikut:
<table border="0" cellpadding="5px" cellspacing="1px" style="font-family:Verdana, Geneva, sans-serif; font-size:11px; background-color:#E1E1E1" width="100%">
        <?php if ($cart = $this->cart->contents()): ?>
        <tr bgcolor="#FFFFFF" style="font-weight:bold">
            <td>No</td>
            <td>Nama Produk</td>
            <td>Harga</td>
            <td>Jumlah</td>
            <td>Total</td>
        </tr>
        <?php
        echo form_open('cart/update_cart');
        $grand_total = 0; $i = 1;

        foreach ($cart as $item):
            echo form_hidden('cart['. $item['id'] .'][id]', $item['id']);
            echo form_hidden('cart['. $item['id'] .'][rowid]', $item['rowid']);
            echo form_hidden('cart['. $item['id'] .'][name]', $item['name']);
            echo form_hidden('cart['. $item['id'] .'][price]', $item['price']);
            echo form_hidden('cart['. $item['id'] .'][qty]', $item['qty']);
        ?>
        <tr bgcolor="#FFFFFF">
            <td>
```

```
            <?php echo $i++; ?>
         </td>
         <td>
            <?php echo $item['name']; ?>
         </td>
         <td>
            Rp. <?php echo number_format($item['price'],2); ?>
         </td>
         <td>
            <?php echo form_input('cart['. $item['id'] .'][qty]', $item['qty'],
'maxlength="3" size="1" style="text-align: right"'); ?>
         </td>
         <?php $grand_total = $grand_total + $item['subtotal']; ?>
         <td>
            Rp. <?php echo number_format($item['subtotal'],2) ?>
         </td>

         <?php endforeach; ?>
      </tr>

      <?php endif; ?>
   </table>
<div style="margin-left: 443px;">
<table>
   <tr>
      <td style="width: 116px;">biaya pengiriman</td>
      <td>Rp. <?php echo number_format($detailpesan ->biaya_pengiriman,2); ?></td>
   </tr>
   <tr>
      <td> <b>Total Biaya</b></td>
      <td><b>Rp. <?php echo number_format($detailpesan ->total_harga,2); ?></b></td>
   </tr>
</table>
</div>
   no transaksi di gunakan untuk konfirmasi pembayaran
<?php include('footer.php');?>
```

Maka tampilan detailnya sebagai berikut ini:

Gambar 1.13

Pembuatan pada konfirmasi_pembayaran sebagai beikut ini :

Kita bikin dulu sebuah controllernya sebagai berikut ini source codenya:

Controllernya dengan nama filenya konfirmasi_pembayaran.php

```
<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

class Konfirmasi_pembayaran extends CI_Controller {

	public function __construct()
	{
		parent::__construct();
		$this->load->model('konfirmasi_model');
	}

	public function index()
	{
		$this->load->view('halaman_konfirmasi_pembayaran');
	}
	public function save_konfirmasi()
	{
		$this->form_validation-
>set_rules('no_pesan','no_pesan','required');
		$this->form_validation-
```

```
>set_rules('email','email','required|email');
            $this->form_validation-
>set_rules('atas_nama','atas_nama','required');
            $this->form_validation-
>set_rules('jumlah_transfer','jumlah_transfer','required|numeric');
            $this->form_validation-
>set_rules('tgl_transfer','tgl_transfer','required');
            $this->form_validation-
>set_rules('no_rekening','no_rekening','required|numeric');
            $this->form_validation->set_rules('catatan','catatan','required');
            if($this->form_validation->run()==false){
                  $this->load->view('halaman_konfirmasi_pembayaran');
            }
            else{
                  $data = array(
                        'no_pesan' => $this->input->post('no_pesan') ,
                        'email' => $this->input->post('email'),
                        'atas_nama' => $this->input->post('atas_nama'),
                        'jumlah_transfer'          =>          $this->input-
>post('jumlah_transfer'),
                        'tgl_transfer' => $this->input->post('tgl_transfer'),
                        'no_rekening' => $this->input->post('no_rekening'),
                        'no_rekening_mimi'          =>          $this->input-
>post('no_rekening_mimi'),
                        'catatan' => $this->input->post('catatan'),
                  );
                  $cek_pesan          =          $this->konfirmasi_model-
>cek_pesan($data)->result();

                  if ($cek_pesan==true) {
                        $cek_pelanggan      =      $this->konfirmasi_model-
>cek_pelanggan($data)->row();

                        if ($cek_pelanggan==true) {
                              $data_insert = array(
                                    'no_pesan'        =>        $this->input-
>post('no_pesan') ,
                                    'id_pelanggan'    =>    $cek_pelanggan-
```

```
>id_pelanggan,
                                        'atas_nama'        =>        $this->input-
>post('atas_nama'),
                                        'jumlah_transfer'     =>     $this->input-
>post('jumlah_transfer'),
                                        'tgl_transfer'        =>        $this->input-
>post('tgl_transfer'),
                                        'no_rekening'        =>        $this->input-
>post('no_rekening'),
                                        'no_rekening_mimi'    =>    $this->input-
>post('no_rekening_mimi'),
                                        'catatan'           =>           $this->input-
>post('catatan'),
                                    );
                                    $simpan_konfirmasi            =            $this-
>konfirmasi_model->insert_konfirmasi($data_insert);
                                    $id['no_pesan']           =           $this->input-
>post('no_pesan');
                                    $data_update    =    array('status_pembayaran'
=> "lunas"
                                    );
                                    $this->konfirmasi_model-
>updatepembayaran($data_update,$id);
                                    if ($simpan_konfirmasi) {
                                        $data['status']="transaksi    pembayaran
berhasil, Barang akan segera dikirim";
                                        $this->load-
>view('info_transaksi',$data);
                                    }
                                    else{
                                        $data['status']="transaksi    pembayaran
gagal, harap isi data yang benar";
                                        $this->load-
>view('info_transaksi',$data);
                                    }

                                }
                                else{
```

```
                                $data['status']="pelanggan tidak ada";
                                $this->load->view('info_transaksi',$data);
                        }
                }
                else{
                        $data['status']="order tidak ada";
                        $this->load->view('info_transaksi',$data);
                }
        }
    }

}
```

File modelnya sebgai berikut ini

Pembuatan modelnya sebagai berikut ini:

Dengan nama file konfirmasi_model.php

```
<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

class Konfirmasi_model extends CI_Model {

    public function __construct()
    {
        //$this->load->database();
    }

    public function cek_pesan($data){
        $ada=   $this->db->get_where('pesan',   array('no_pesan'   =>
$data['no_pesan']));
        return $ada;
    }

    public function cek_pelanggan($data){
        return   $this->db->get_where('pelanggan',   array('email'   =>
$data['email']));

    }
```

```
	public function insert_konfirmasi($data){
		return $this->db->insert('bayar', $data);
	}
	public function updatepembayaran($data,$id){
		$this->db->update("pesan",$data,$id);
	}
}
```

Proses selanjutnya pembuatan viewnya sebagai berikut ini:

Dengan nama file viewnya sebagai berikut ini:

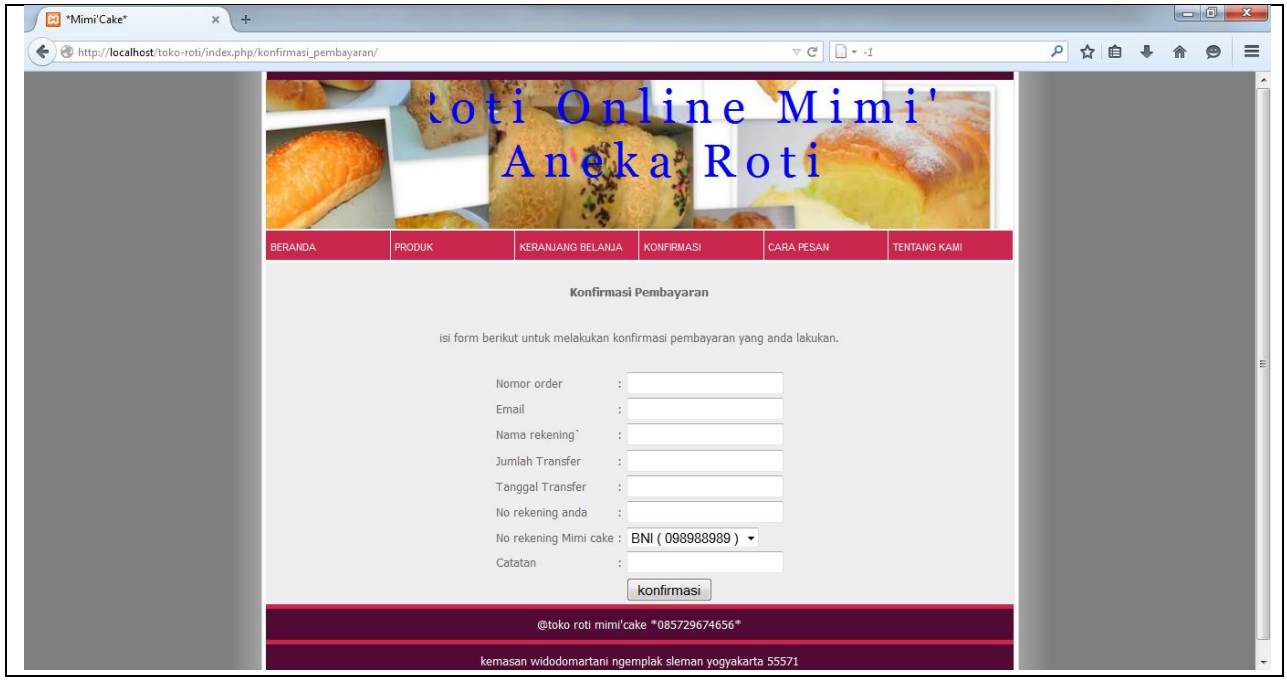
Konfirmasi_pembayaran.php

```
<?php include('header.php');?>
	<center><h4>Konfirmasi Pembayaran</h4></center>
	<center><p>isi form berikut untuk melakukan konfirmasi pembayaran yang anda lakukan.</p></center>
	<form action="<?=base_url();?>index.php/konfirmasi_pembayaran/save_konfirmasi"method="post">
	<center>
		<table>
			<tr>
				<td>Nomor order</td>
				<td>:</td>
				<td><input type="text" name="no_pesan"><?php echo form_error('no_pesan','<div class="error">','</div>');?></td>
			</tr>
			<tr>
				<td>Email</td>
				<td>:</td>
				<td><input type="text" name="email"><?php echo form_error('email','<div class="error">','</div>');?></td>
			</tr>
			<tr>
				<td>Nama rekening`</td>
				<td>:</td>
```

```
                        <td><input type="text" name="atas_nama"><?php
echo form_error('atas_nama','<div class="error">','</div>');?></td>
                    </tr>
                    <tr>
                        <td>Jumlah Transfer</td>
                        <td>:</td>
                        <td><input                                type="text"
name="jumlah_transfer"><?php   echo   form_error('jumlah_transfer','<div
class="error">','</div>');?></td>
                    </tr>
                    <tr>
                        <td>Tanggal Transfer</td>
                        <td>:</td>
                        <td><input        type="text"        id="datepicker"
name="tgl_transfer"><?php        echo        form_error('tgl_transfer','<div
class="error">','</div>');?></td>
                    </tr>
                    <tr>
                        <td>No rekening anda</td>
                        <td>:</td>
                        <td><input                                type="text"
name="no_rekening"><?php       echo       form_error('no_rekening','<div
class="error">','</div>');?></td>
                    </tr>
                    <tr>
                        <td>No rekening Mimi cake</td>
                        <td>:</td>
                        <td><select name="no_rekening_mimi">
                            <option   value="bni">   BNI   (   098988989
)</option>
                            <option   value="bri">   BRI   (   868776887
)</option>
                        </select></td>
                    </tr>
                    <tr>
                        <td>Catatan</td>
                        <td>:</td>
                        <td><input   type="text"   name="catatan"><?php
```

```
echo form_error('catatan','<div class="error">','</div>');?></td>
                    </tr>
                    <tr>
                        <td></td>
                        <td></td>
                        <td><input type="submit" name="kirim"
value="konfirmasi"></td>
                    </tr>
                </table>
        </center>
        </form>
<?php include('footer.php');?>
```

Tampilan outputnya sebagai berikut ini:

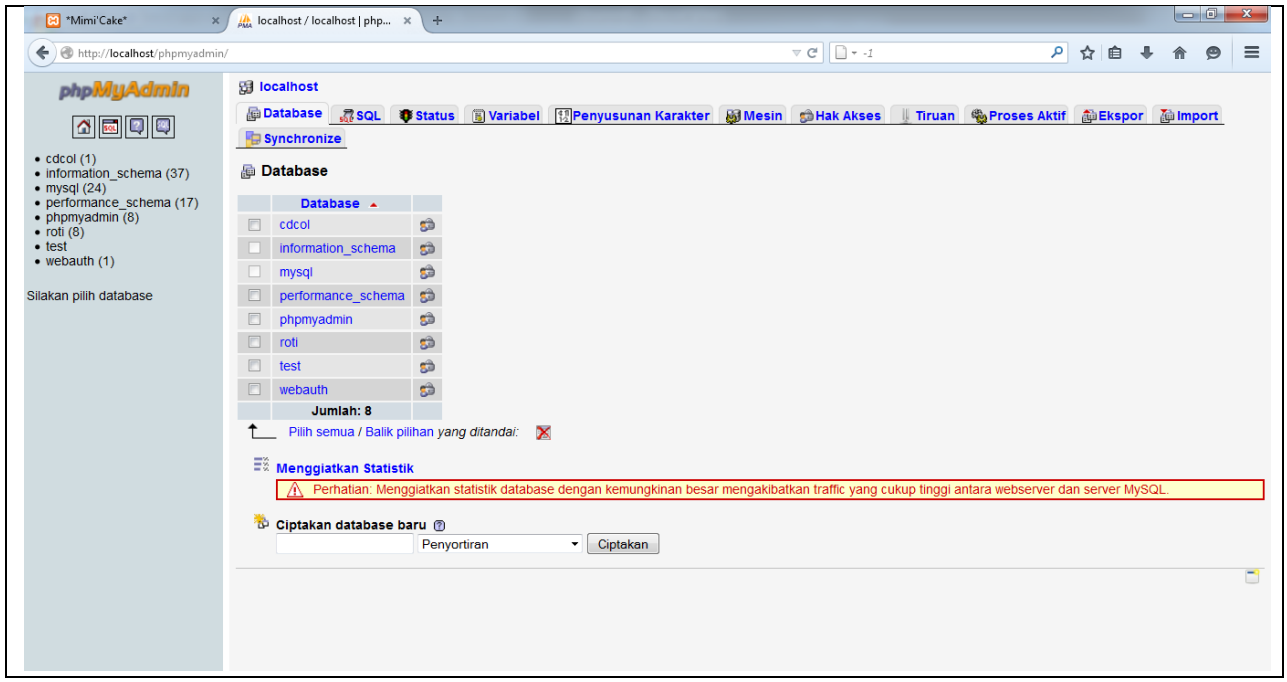


Gambar 1.14

PROSES PEMBUATAN DATA BASENYA

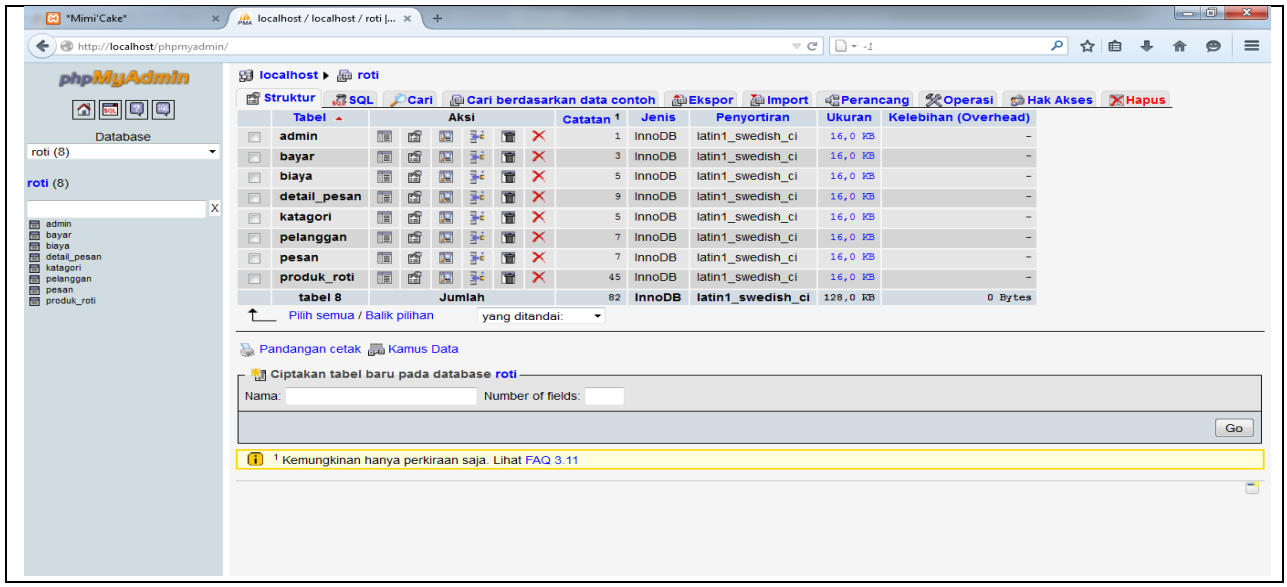
Proses selanjutnya Untuk pembuaan sebuah data basenya sebagai berikut ini gambar data basenya.

Dengan Nama Data Basenya Roti.sql



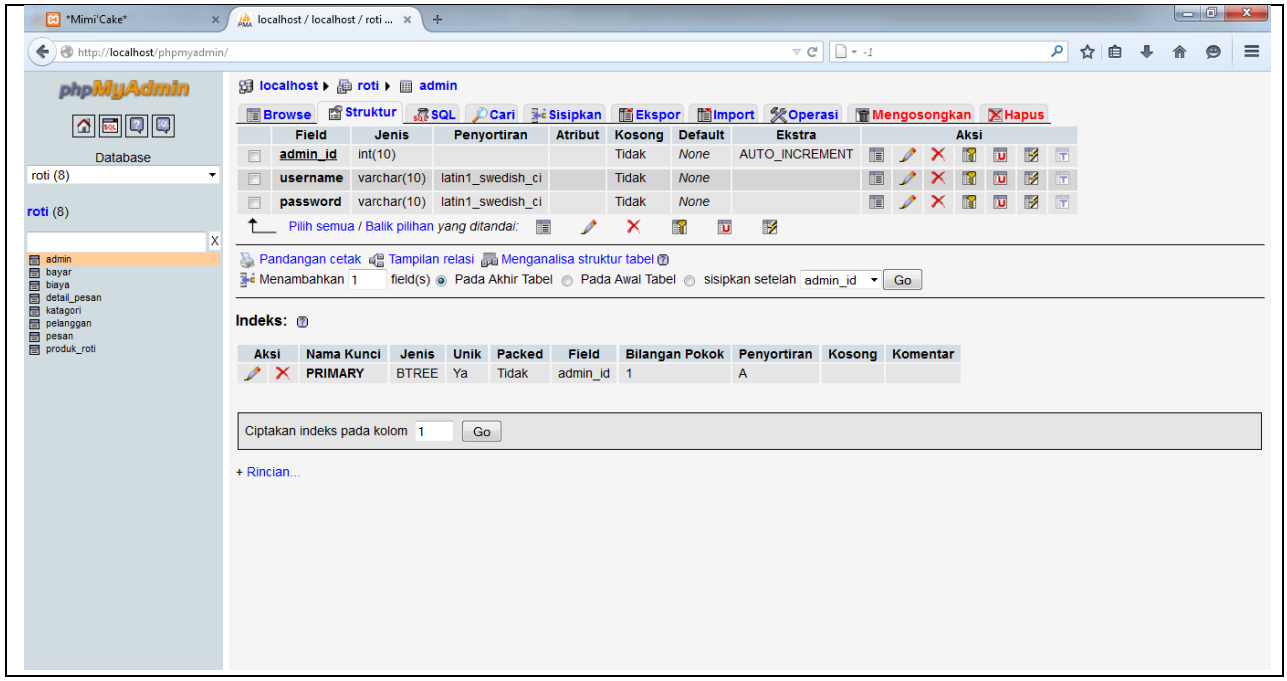
Gambar 1.15

Berikut ini terdapat 8 tabelnya sebagai berikut ini



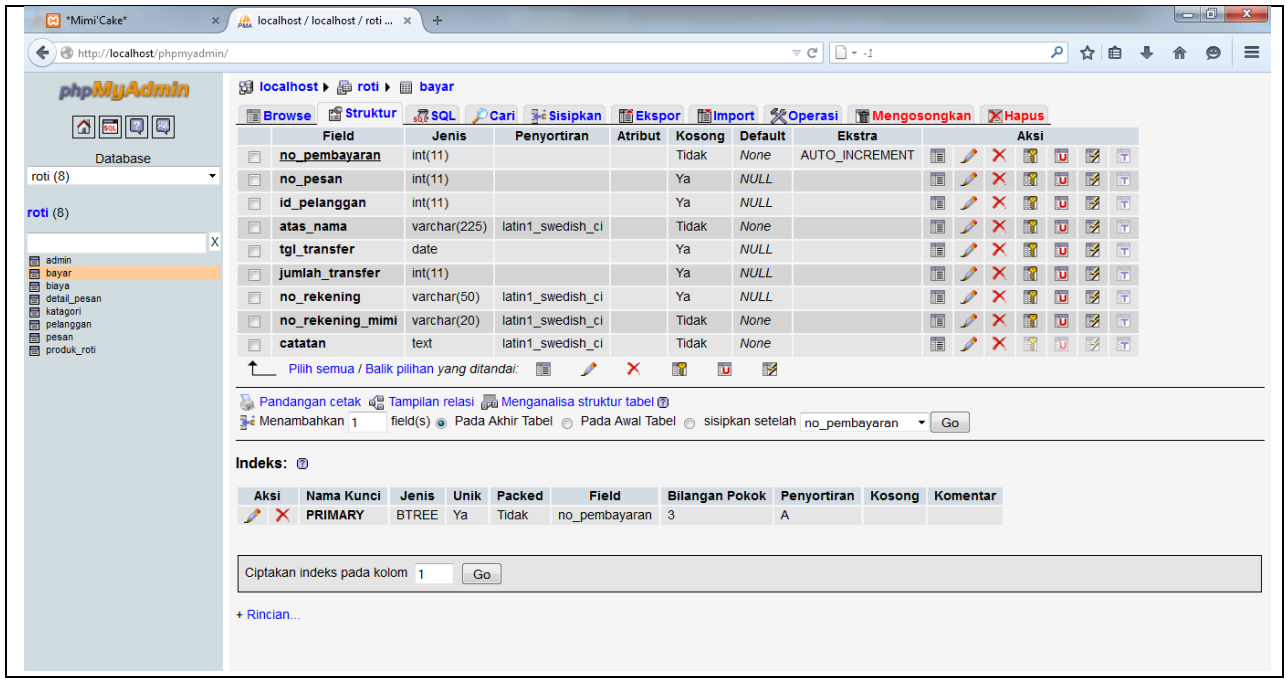
Gambar 1.16

Berikut ini ditable admin terdapat Filednya sebagai berikut ini:



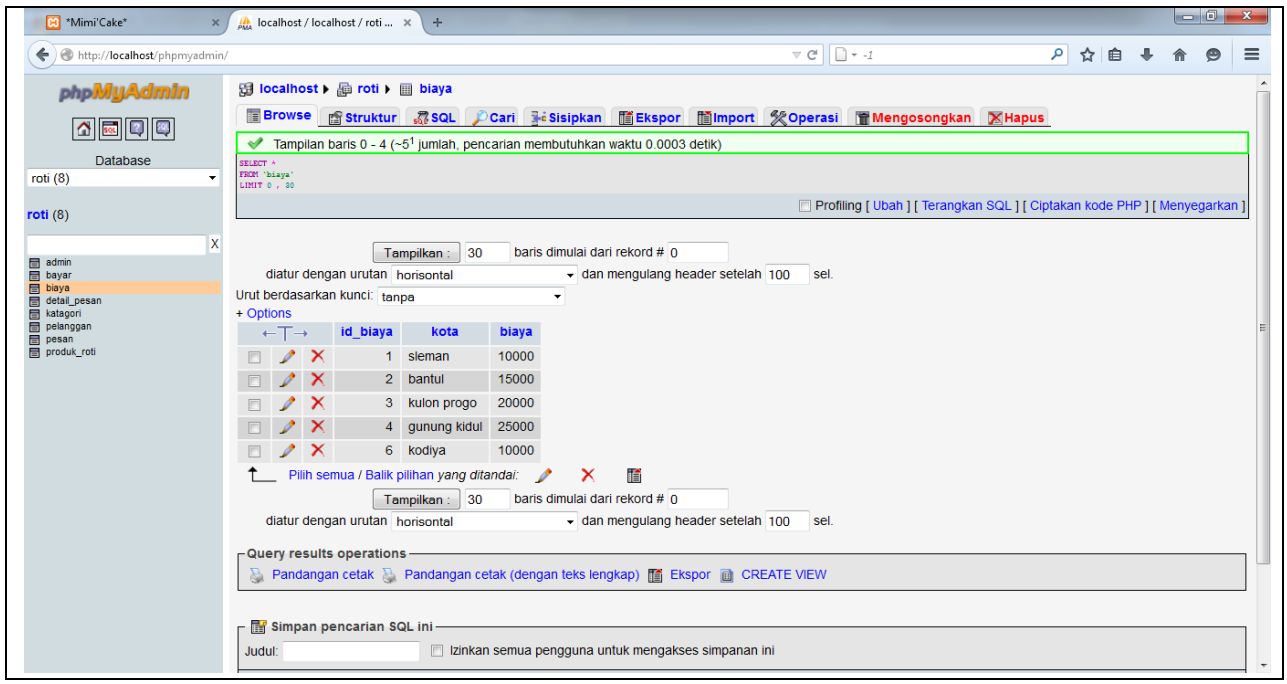
Gambar 1.17

Berikut ini ditable bayar terdapat Filednya sebagai berikut ini:



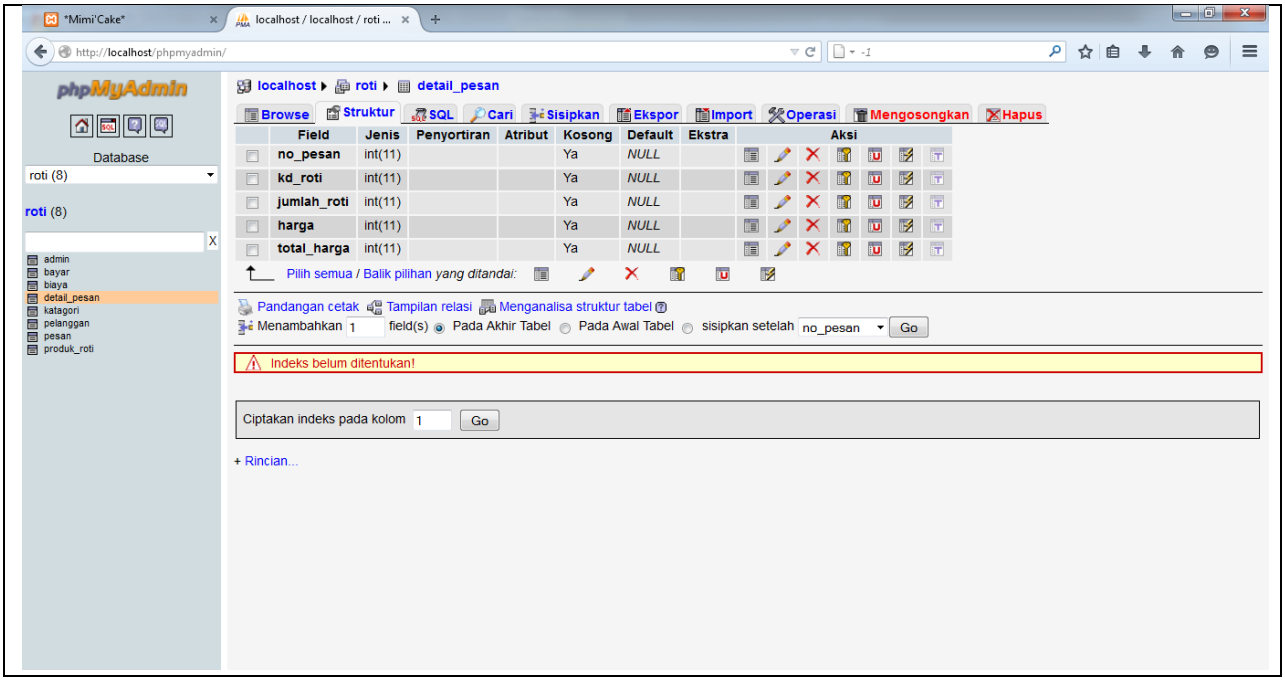
Gambar 1.18

Berikut ini ditable biaya terdapat Filednya sebagai berikut ini:



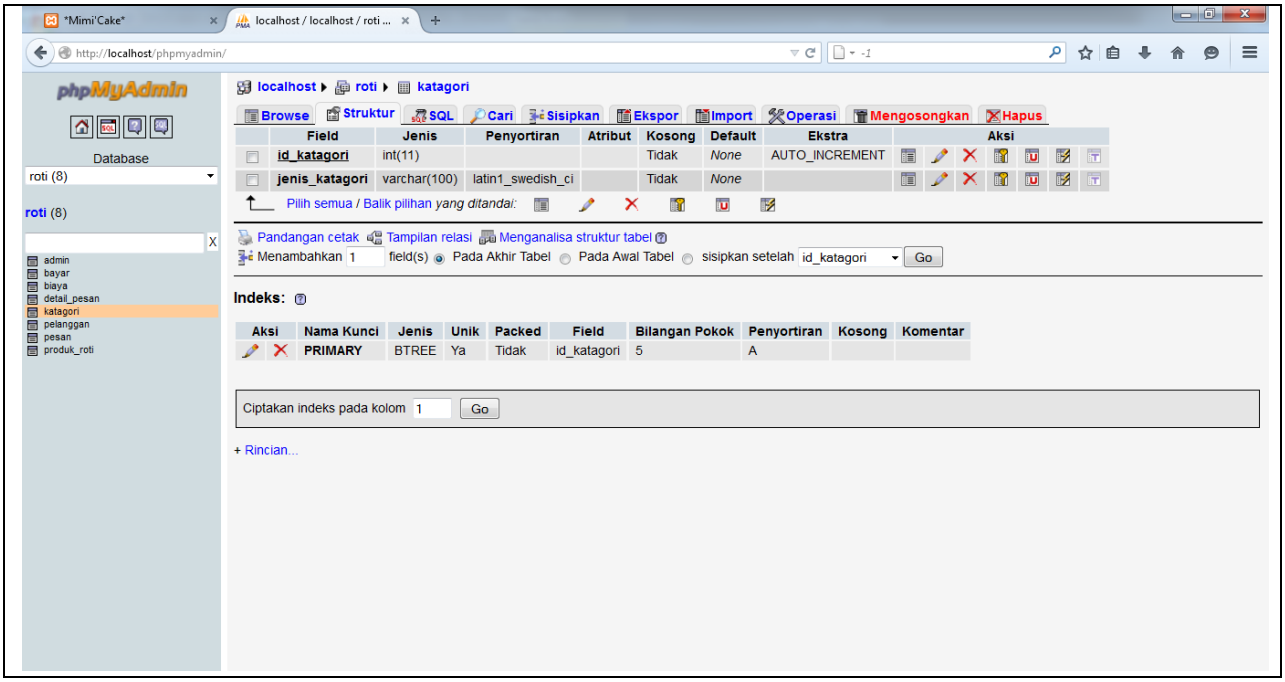
Gambar 1.19

Berikut ini ditable detail_pesan terdapat Filednya sebagai berikut ini:



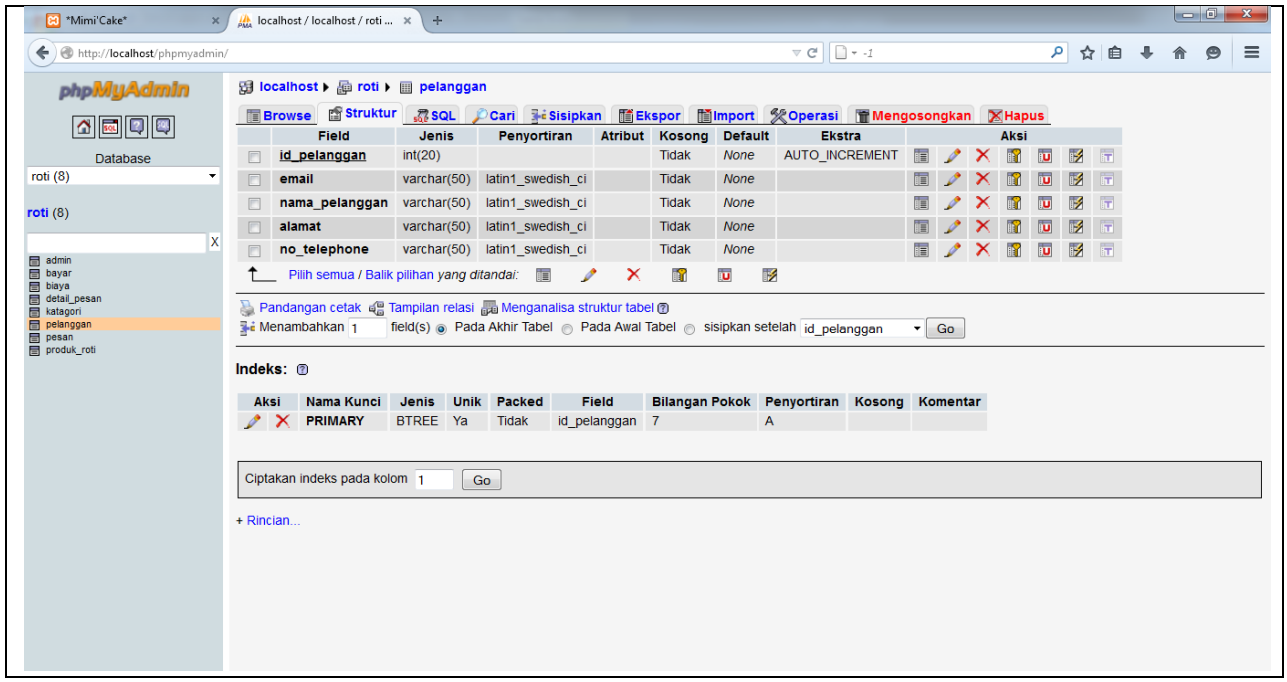
Gambar 1.20

Berikut ini ditable katagori terdapat Filednya sebagai berikut:



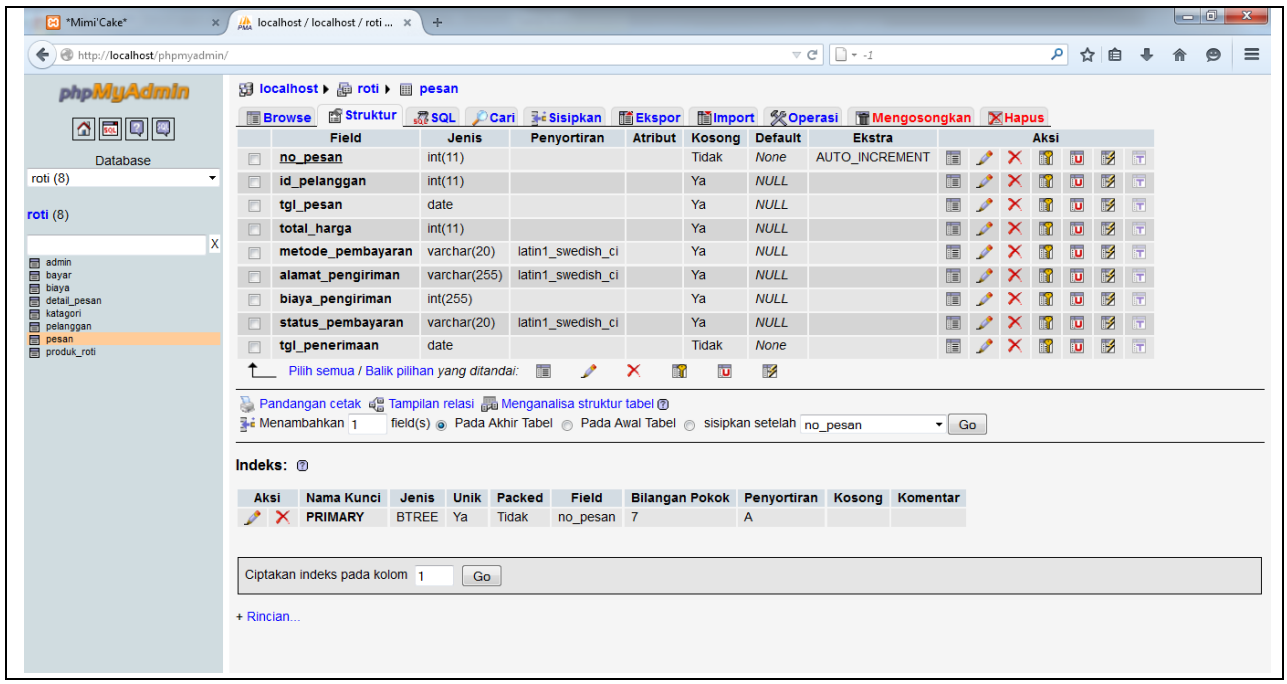
Gambar 1.21

Berikut ini ditable pelanggan terdapat Filednya sebagai berikut:



Gambar 1.22

Berikut ini ditable pelanggan terdapat Filednya sebagai berikut:



Gambar 1.23

Berikut ini ditable produk_roti terdapat Filednya sebagai berikut: