

PERTEMUAN KE – 2

Model Menggunakan CFormModel

A. TUJUAN

1. Mampu membuat web, menggunakan *Model-View-Controller* Framework Yii menggunakan kelas `CFormModel`.
2. Membuat Web statis dengan kelas `CFormModel`

B. TEORI SINGKAT

1. Model

Kelas Model dalam aplikasi Yii-based diletakkan dalam folder `protected/models/`. File-file model tersebut disimpan dengan nama file berekstensi `.php`. Di dalam Yii model menggunakan kelas-kelas perluasan dari kelas `CModel`, yang juga digunakan oleh sub kelas `CFormModel`, dan `CActiveRecord`. Untuk `CActiveRecord` untuk digunakan operasi rekaman dalam tabel, dan `CFormModel` hanya digunakan untuk operasi yang menghasilkan format HTML.

Contoh penulis model

```
class NamaKelas extend CFormModel {
//atribut
    public $atribut_1;
    public $atribut_2;
    public $atribut_3;
    .....
//metode
    public function rules()
    {
        .....
    }

    public function attributeLabels()
    {
        .....
    }
}
```

2. View

View adalah kombinasi HTML dan PHP yang membantu untuk menciptakan output yang diinginkan bahwa pengguna untuk melihat dalam browser nya. Model sesuatu objek yang rumit dalam desain tetapi mudah digunakan. Sebaliknya, view mudah dalam desain untuk membuat tampilan aplikasi web.

Contoh :

```
<?php
/* @var $this SiteController */
$this->pageTitle=Yii::app()->name;
?>
<h1>Welcome to <i>
<?php echo CHtml::encode(Yii::app()->name); ?></i></h1>
```

3. Controller

Bagian utama ketiga dari pendekatan desain MVC adalah controller. Controller bertindak sebagai agen, perantara yang menangani pengguna dan menjalankan aksi. Controller adalah bagian yang menghubungkan antara model dan view.

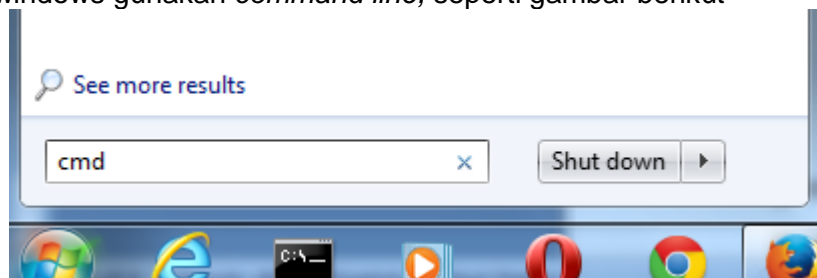
Contoh :

```
class SiteController extends Controller
{
    public function actions()
    {
        return array(...);
    }

    public function actionIndex()
    {
        $this->render('index');
    }
}
```

C. PRAKTIK

1. Buatlah project baru dengan menjalankan generator yiic
 - Klik windows gunakan *command line*, seperti gambar berikut



ketikkan seperti berikut :

```
C:\Users\lenovo>cd C:\xampp\htdocs\framework\
```

```
C:\xampp\htdocs\framework>yiic webapp ../pertemuan2
Create a Web application under 'C:\xampp\htdocs\pertemuan2'?
(yes|no) [no]: yes
```

2. Lakukan pengaturan konfigurasi dengan membuka file `./protected/config/main.php`, ubah seperti pada kode program berikut :

```
20
21     'modules'=>array(
22         // uncomment the following to enable the Gii tool
23
24         'gii'=>array(
25             'class'=>'system.gii.GiiModule',
26             'password'=>'123',
27             // If removed, Gii defaults to localhost only. Edit carefully to taste.
28             'ipFilters'=>array('127.0.0.1',':::1'),
29         ),
30
31     ),
```

kemudian simpan kembali

3. Buatlah model dengan menggunakan turunan kelas `CFormModel`, seperti pada kode program berikut :

awal file Kota.php

```
<?php
/*****
 * Kelas Kotak
 *****/
class Kotak extends CFormModel
{
    /* atribut/properti*/
    public $panjang;
    public $lebar;

    // validasi
    public function rules()
    {
        return array(
            array('panjang, lebar', 'required'),
            array('panjang, lebar', 'numerical', 'integerOnly'=>true),
        );
    }

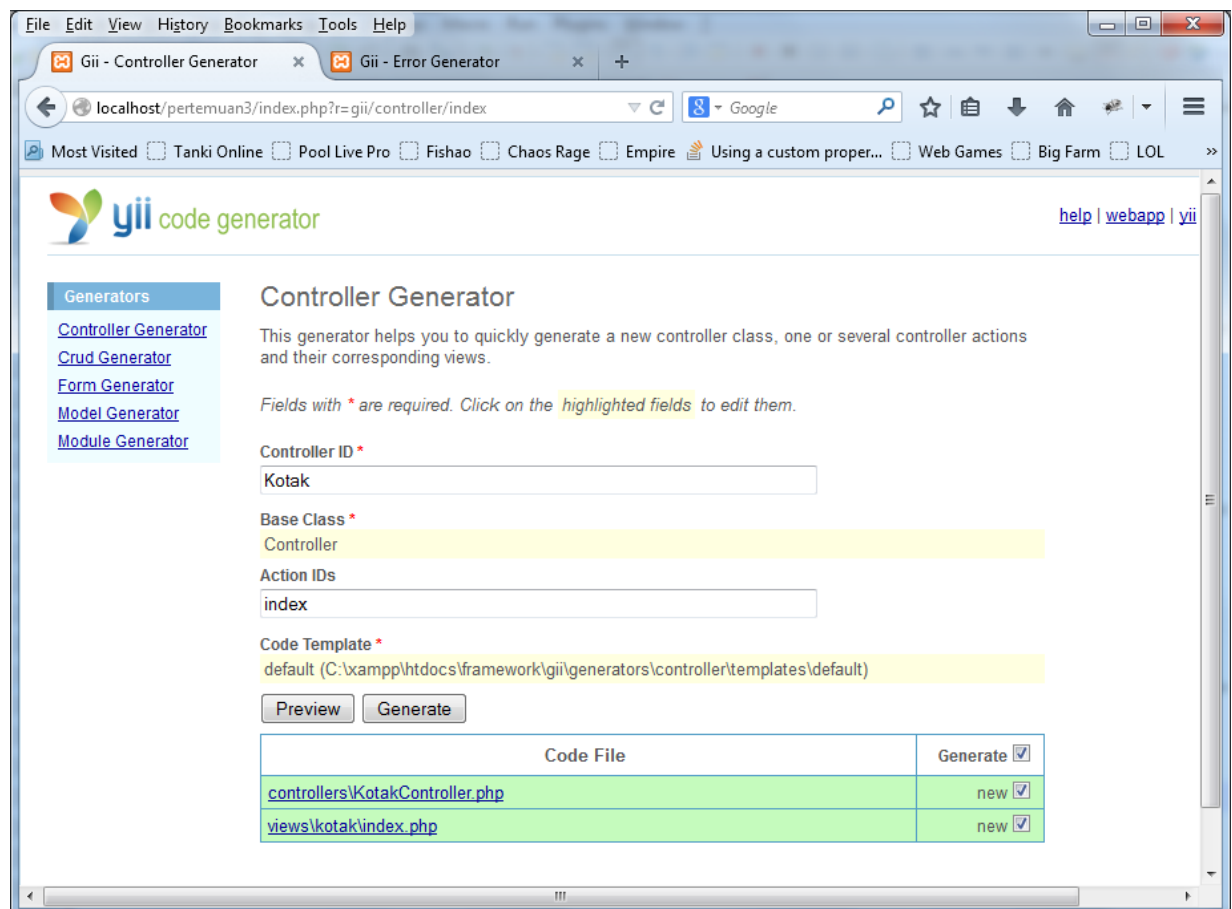
    /**
     * Deklarasi attribute label.
     */
    public function attributeLabels()
    {
        return array(
            'panjang'=>'Panjang Kotak',
            'lebar'=>'Lebar Kotak',
        );
    }

    public function luas()
    {
        return $this->panjang * $this->lebar;
    }

    public function keliling()
    {
        return 2*($this->panjang+$this->lebar);
    }
}
```

akhir file Kota.php

4. Membuat model dengan menggunakan gii generator <http://localhost/pertemuan2/index.php?r=gii>
- Klik Controller Generator
 - Isikan Controller ID : Kota (sama dengan nama kelas Kotak)
 - Secara default Action IDs terisi index
 - Klik tombol Preview
 - Secara otomatis akan membentuk file, yang ter tulis dalam folder
 `.\protected\controller\KotakController.php`
 `.\protected\views\kotak\index.php`
 - Selanjutnya klik Generate seperti gambar dibawah :



Mengubah bagian KotakController.php

awal file KotakController.php

```
<?php
class KotakController extends Controller
{
    public function actionIndex()
    {
        $this->render('index');
    }

    public function actionInput()
    {
        $model=new Kotak;
        $model->unsetAttributes(); // clear any default values

        //memanggil fungsi Ajax untuk validasi input
        $this->performAjaxValidation($model);

        if(isset($_POST['Kotak']))
        {
            $model->attributes= $_POST['Kotak'];
            $panjang      = $_POST['Kotak']['panjang'];
            $lebar        = $_POST['Kotak']['lebar'];
            $this->redirect(array('luas',
                                'panjang'=>$panjang,'lebar'=>$lebar));
        }

        $this->render('input',array(
            'model'=>$model,));
    }

    public function actionLuas()
```

```

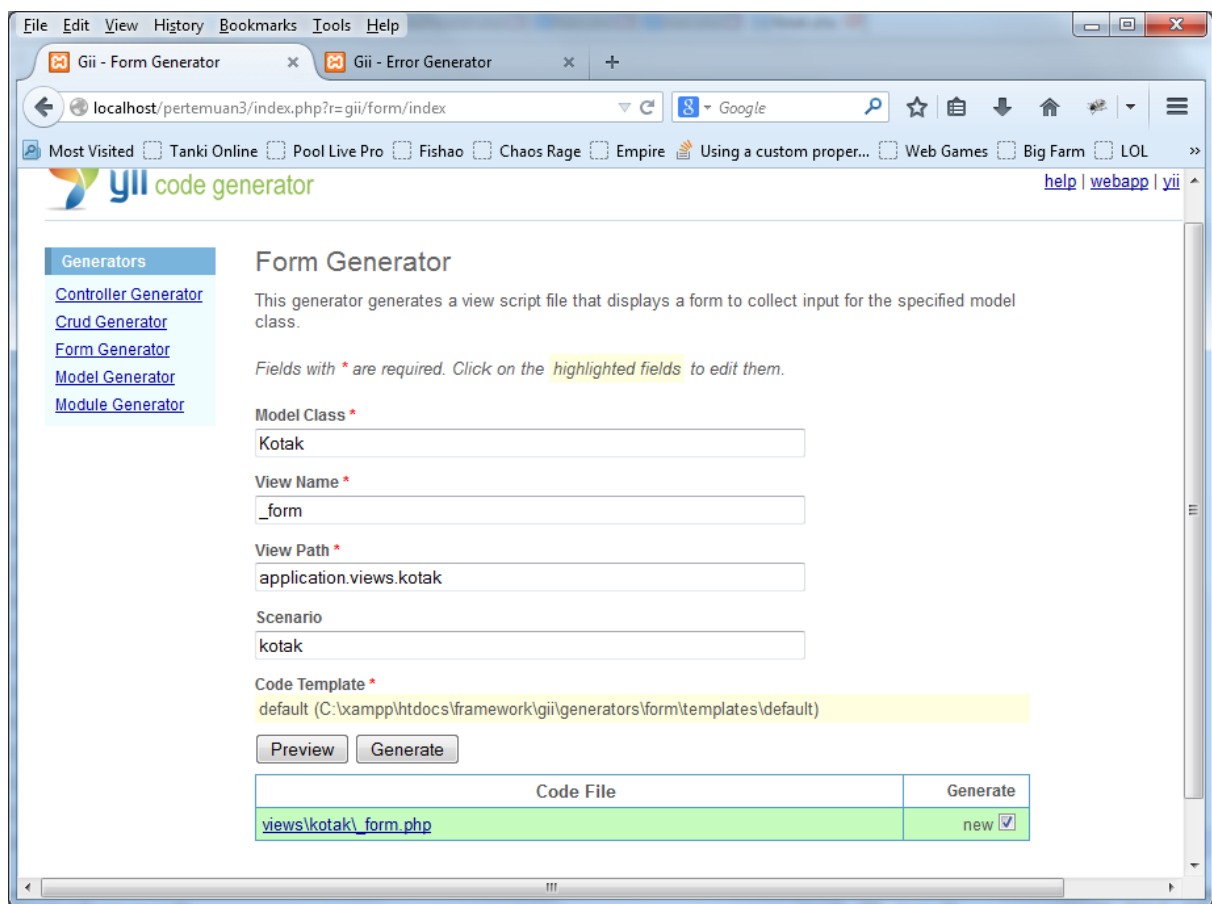
{
    $model=new Kotak;
    $model->unsetAttributes(); // clear any default values
    if(isset($_GET['Kotak']))
        $model->attributes=$_GET['Kotak'];
    $this->render('luas',array(
        'model'=>$model,
    ));
}

// ajax untuk validasi
protected function performAjaxValidation($model)
{
    if(isset($_POST['ajax']) && $_POST['ajax']==='kotak-form')
    {
        echo CActiveForm::validate($model);
        Yii::app()->end();
    }
}
}

```

akhir file KotakController.php

- Bagian View, menciptakan Form kotak
 - Klik Form Generator
 - Isikan Model Class : Kotak
 - Views Name _form, seperti pada gambar berikut :



- Buatlah File pada view dengan nama input.php

awal file input.php

```

<?php
/* @var $this KotakController */

```

```

/* @var $model Kotak */

$this->breadcrumbs=array(
    'Kotak'=>array('index'),
    'Input Data',
);
?>

<h1>Input Kotak</h1>

<?php $this->renderPartial('_form', array('model'=>$model)); ?>

```

akhir file input.php

7. Mengaktifkan validasi input menggunakan ajax

Ubah dan tambahkan dalam file _form.php

'id'=>'kotak-_form-form', diubah jadi 'id'=>'kotak-form',
tambahkan untuk cek validasi dalam widget 'CActiveForm' seperti berikut :

```

'enableAjaxValidation'=>true,
'enableClientValidation'=>true,
'clientOptions'=>array(
    'validateOnType'=>true,
    'validateOnChange' => true,),

```

sehingga secara lengkap menjadi :

awal file _form.php

```

<?php
/* @var $this KotakController */
/* @var $model Kotak */
/* @var $form CActiveForm */
?>

<div class="form">

<?php $form=$this->beginWidget('CActiveForm', array(
    'id'=>'kotak-form',
    'enableAjaxValidation'=>true,
    'enableClientValidation'=>true,
    'clientOptions'=>array(
        'validateOnType'=>true,
        'validateOnChange' => true,
    ),
)); ?>

    <p class="note">Fields with <span class="required">*</span> are
    required.</p>

    <?php echo $form->errorSummary($model); ?>

    <div class="row">
        <?php echo $form->labelEx($model,'panjang'); ?>
        <?php echo $form->textField($model,'panjang'); ?>
        <?php echo $form->error($model,'panjang'); ?>
    </div>

    <div class="row">
        <?php echo $form->labelEx($model,'lebar'); ?>
        <?php echo $form->textField($model,'lebar'); ?>
        <?php echo $form->error($model,'lebar'); ?>
    </div>

```

```

<div class="row buttons">
    <?php echo CHtml::submitButton('Submit'); ?>
</div>

<?php $this->endWidget(); ?>

</div><!-- form -->

```

akhir file _form.php

8. Buatlah view menampilkan hasil perhitungan dengan nama `luas.php`

awal file luas.php

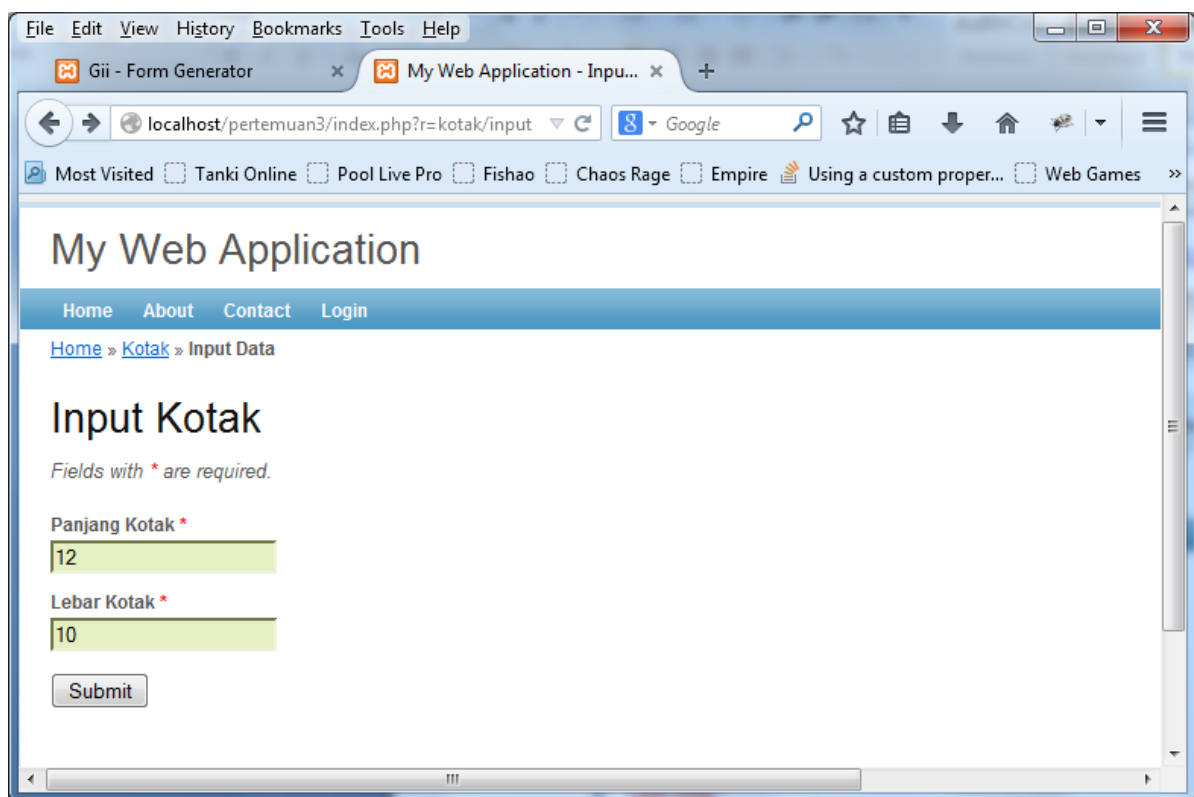
```

<?php
$model->panjang = $_GET["panjang"];
$model->lebar = $_GET["lebar"];
echo "Panjang " . $model->panjang; echo "<br>";
echo "Lebar " . $model->lebar; echo "<br>";
echo "Luas " . $model->luas(); echo "<br>";
echo "Keliling " . $model->keliling();
?>

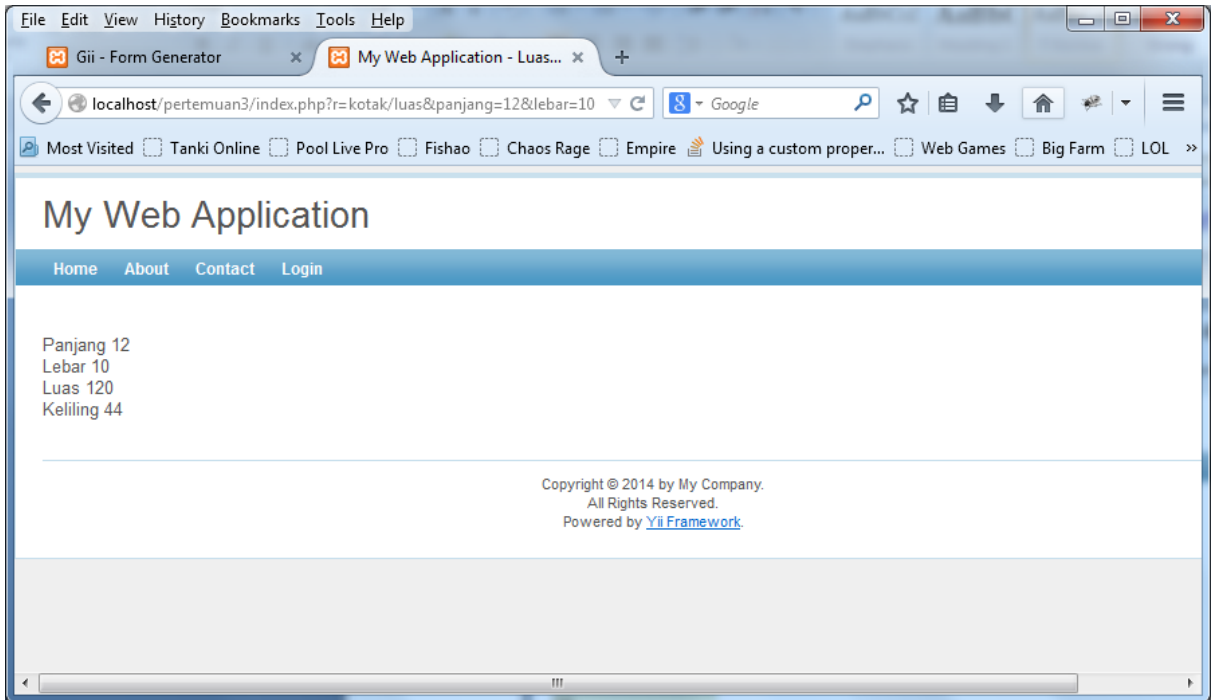
```

akhir file luas.php

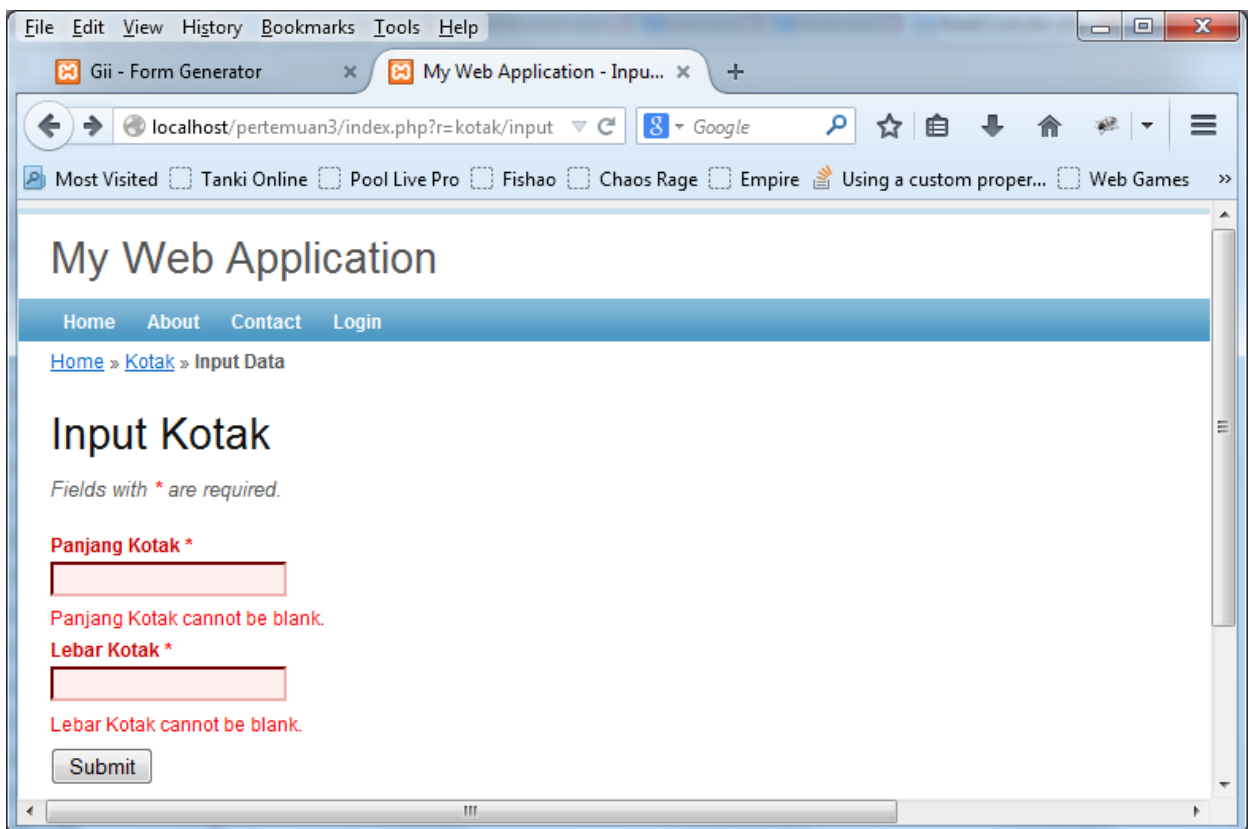
Hasil jalankan dengan URL <http://localhost/pertemuan2/index.php?r=kotak/input>



- Klik Submit



Mencoba Validasi, ulangi <http://localhost/pertemuan2/index.php?r=kotak/input>
 Untuk mencoba Panjang Kotak dikosongkan, apa yang terjadi ? diisi dengan huruf apa yang terjadi ?, seperti pada gambar berikut :



D. LATIHAN

1. Kasus di atas akan dikembangkan menjadi sebuah kubus/balok,
2. Tambahkan satu atribut untuk kelas Kotak tebal,
3. Tambah masukkan tebal,
4. Tambahkan fungsi/metode menghitung volume
5. Tampilkan hasil perhitungan

6. Buatlah dengan cara yang sama untuk menghitung lingkaran luas = $22/7 * r * r$ kemudian kembangkan, menjadi silinder,
7. buatlah untuk menghitung silinder

E. TUGAS

Buatlah analisis dan jawablah pertanyaan di bawah ini:

1. Di Controller terdapat method `render(..)` dan `renderPartial(..)`, jelaskan penggunaan apa bedanya.
2. Apa yang dimaksud dengan perintah `$model->unsetAttributes()`. ?
3. Apa yang dimaksud dengan perintah `$this->render('luas',array('model'=>$model,));` ?
4. Apa yang dimaksud dengan perintah `$model=new Kotak;`
5. Apa yang dimaksud dengan perintah `$panjang = $_POST['Kotak']['panjang'];`
6. Apa yang dimaksud dengan perintah `$model->attributes=$_GET['Kotak'];`
7. Apa yang dimaksud dengan perintah `class Kotak extends CFormModel`
8. Dalam model ada rules `array('panjang, lebar', 'required'),`
9. dan `array('panjang, lebar','numerical', 'integerOnly'=>true),`
10. dalam modal ada method `attributeLabels()` ada
`return array(`
`'panjang'=>'Panjang Kotak',`

Dalam view terdapat perintah jelaskan perintah di bawah

11. `$model->panjang = $_GET["panjang"];`
12. `echo "Panjang " . $model->panjang;`
13. `echo "Keliling " . $model->keliling();`