

Osnove izrade PHP aplikacija

Yii (4)

Izradio: Davor Cihlar

Plan

- ▶ Što ćemo naučiti
 - ▶ AJAX
 - ▶ TinyMCE

Yii i jQuery

- ▶ Yii je temeljen na jQueryju
- ▶ Mnoge su funkcionalnosti jQueryja ugrađene
- ▶ Korišćenje jQueryja je u mnogim slučajevima transparentno
 - ▶ Nije potrebno znati jQuery
 - ▶ Nije potrebno znati čak niti JavaScript

Provjera ispravnosti putem AJAX-a (1)

- ▶ CRUD generiran Gii-jem u akcijama "create" i "update" ima komentiranu liniju:
 - ▶ `$this->performAjaxValidation($model);`
 - ▶ Metoda `performAjaxValidation` implementirana je u kontroleru i nalazi se skroz na kraju datoteke kontrolera
- ▶ Obrasci generirani Gii-jem također imaju komentiran dio koda za provjeru ispravnosti putem AJAX-a
 - ▶ Isječak je gotovo identičan metodi `performAjaxValidation`
- ▶ Takve isječke potrebno je otkomentirati ukoliko je potrebna provjera ispravnosti putem AJAX-a!

Provjera ispravnosti putem AJAX-a (2)

- ▶ Osim izmjena u kontroleru, potrebno je i omogućiti AJAX provjeru u prikazu u samom *widgetu* obrasca
 - ▶ Postavka `enableAjaxValidation`
- ▶ Primjer:
 - ▶

```
$form = $this->beginWidget('CActiveForm',  
    array(  
        'id'=>'korisnik-form',  
        'enableAjaxValidation'=>true  
    )  
);
```

Zadatak (1)



- ▶ Omogućiti provjeru ispravnosti podataka prilikom dodavanja korisnika!

Ručni AJAX pozivi

- ▶ Pomoću jQueryja
 - ▶ URL je akcija kontrolera
- ▶ Potrebno je priključiti jQuery
 - ▶ `Yii::app()->clientScript->registerCoreScript('jquery');`
- ▶ U akciji kontrolera potrebno je odrediti zaglavlje Content-Type
 - ▶ `header("Content-Type: application/json");`
- ▶ Pomoću statičke klase CJSON moguće je kodirati podatke u JSON:
 - ▶ `echo CJSON::encode(date('r'));`
- ▶ Više na:
 - ▶ <http://www.yiiframework.com/doc/api/1.1/CJSON>

Primjer (1)



▶ Java Script:

```
▶ function dobio(d) {  
    alert(d);  
}  
function dohvatiAjaxom() {  
    $.get('?r=kontroler/ajaxjq', [], dobio);  
}
```

▶ Kontroler:

```
▶ public function actionAjaxJQ() {  
    header('Content-Type: application/json');  
    echo CJSON::encode(date('r'));  
    Yii::app()->end();  
}
```

CHtml

- ▶ CHtml je statična klasa s raznim metodama koje pomažu pri stvaranju prikaza
- ▶ Primjer:
 - ▶ `echo CHtml::submitButton('Pošalji');`
- ▶ Više na:
 - ▶ <http://www.yiiframework.com/doc/api/1.1/CHtml>

Izmjena HTML sadržaja AJAX-om

- ▶ Pomoću metode `ajaxLink()`
- ▶ Link AJAX-om poziva zadanu akciju te ispis akcije ubacuje na mjesto zadano jQuery selektorom
- ▶ Akcija koju poziva AJAX mora završiti s:
 - ▶ `Yii::app()->end();`
- ▶ Primjer:
 - ▶

```
echo CHtml::ajaxLink(
    'Tekst linka',
    array('kontroler/akcija'),
    array('update'=>'selektor')
);
```

Primjer (2)



▶ Prikaz:

▶ `<?php`

```
echo CHtml::ajaxLink(  
    'Izmijeni tekst',  
    array( 'kontroler/procajax', 'prikazi'=>42 ),  
    array( 'update'=>'#tekst' )  
);  
?>  
<div id="tekst">Lorem ipsum</div>
```

▶ AJAX akcija modela:

```
public function actionProcAjax() {  
    echo 'Hello world ' . $_GET['prikazi'];  
    Yii::app()->end();  
}
```

Primjer (3.1)



▶ Glavni prikaz:

▶ `<?php`

```
echo CHtml::ajaxLink(  
    'Izmijeni broj',  
    array( 'kontroler/procajax2', 'broj'=>42 ),  
    array( 'update'=>'#tekst' )  
);  
?>
```

```
<div id="tekst"><?php  
$this->renderPartial( '_ajax2', array( 'b'=>4 ) );  
?></div>
```

▶ Podprikaz (`_ajax2.php`):

▶ `Broj:`

```
<?php echo $b; ?>
```

Primjer (3.2)



- ▶ AJAX akcija modela:

- ▶

```
public function actionProcAjax2() {  
    $this->renderPartial('_ajax2',  
                        array('b' =>  
                            $_GET['broj']));  
    Yii::app()->end();  
}
```

Zadatok (2)



TinyMCE (1)

- ▶ Podrška za TinyMCE izvedena je kao proširenje sustava
 - ▶ <http://www.yiiframework.com/extension/tinymce/>
- ▶ Uključivanje proširenja:
 - ▶ Dohvatiti arhivu
 - ▶ Otpakirati u `protected/extensions`
 - ▶ Konfiguraciju nije potrebno uređivati

TinyMCE (2)

- ▶ Jednostavni TinyMCE moguće je ugraditi u prikazu kao *widget*:
 - ▶

```
$this->widget('application.extensions.tinymce.ETinyMce',  
              array('name'=>'html'));
```
 - ▶ Postavka name je ime podatka koji će se poslati obrascem
- ▶ Napredni TinyMCE ima samo dodatnu postavku editorTemplate:
 - ▶

```
$this->widget('application.extensions.tinymce.ETinyMce',  
              array('name'=>'html',  
                    'editorTemplate'=>'full'));
```

Zadatak (3)

