

04 Overzicht van kleinste landen in Europa

Inleiding

We gaan verder met het overzicht dat we in de vorige les hebben gemaakt.

We gaan nu onze eigen query's toe voegen en maken een gesorteerd overzicht van alle kleine Europese landen.

Wat leren we?

- we leren hoe je met Yii op verschillende manieren data uit de database kan halen
- we leren hoe je zelf met Yii de gegevens gesorteerd kan afdrukken.
- we leren hoe we via een query een selectie kunnen maken.

Sorteren en selecteren

Via de Yii Gridview widget hebben we geleerd dat we eenvoudig kunnen sorteren en selecteren. De Yii Gridview heeft ook beperkingen. Je kunt bijvoorbeeld niet alle landen zoeken met *meer dan X* inwoners. Soms wil je ook dat de gebruiker zelf niet kan selecteren. Hij krijgt dan niet alle regels uit de tabel te zien, maar bijvoorbeeld alleen de regels met de datum van vandaag.

We gaan het sorteren en selecteren via de controller coderen.

Kijk nog eens naar de *CountryController* bij de *actionOverzicht*, daar staat:

```
$countries=Country::find()->all();
```

Dit statement kijkt naar het object *Country* dat in het model is gemaakt en zoekt dan naar alle regels uit dit object. Dit object wordt in het model gekoppeld aan de tabel *country*.

Yii vertaald dit naar de query

```
SELECT * FROM country
```

We kunnen nog veel meer, kijk maar eens naar het volgende uitgebreide voorbeeld:

```
$countries = country::find()->where(['Continent' => 'Europe']->orderBy(['Name' => SORT_ASC])->all();
```

Hier worden alle *countries* geselecteerd met *Continent=Europe*, gesorteerd op *Name*.

Dit is een soort Yii query; een query in Yii-formaat.

Het SQL-statement zou er zo uit zien:

```
SELECT * FROM country WHERE Continent = 'Europe' ORDER BY Name ASC
```

Vind je een query eenvoudiger? OK dat kan ook.

```
$sql=SELECT * FROM country WHERE Continent = 'Europe' ORDER BY Name ASC";
```

```
$result = Yii::$app->db->createCommand($sql)->queryOne();
```

```
// Dit is hetzelfde als
```

```
$countries = country::find()->where(['Continent' => 'Europe']->orderBy(['Name' => SORT_ASC])->all();
```

Op deze manier kun je dus via de code in de controller een selectie maken en de sortering aanpassen. In de volgende opdracht komt dit terug.

Opdracht 4a

- - Maar een nieuw menu-item en noem dit *Overzicht Europe*.
 - Maak een nieuwe function *actionOverzichtEurope*, zet deze in de juiste controller en koppel deze aan het menu-item.

(let op de vertaling naar de routing; in de routing staat nooit een hoofdletter en woorden worden gescheiden door een streepje "-", [zie les 2 over routing](#)).

- Laat in dit overzicht alle landen van Europa zien. Gebruik hiervoor het stukje code dat hierboven staat (\$countries = country::find()->where.....).
- Druk de volgende kolommen af:

Name en Surface Area

- - Zet de twee kolommen netjes onder elkaar
 - Formateer de Surface Area netjes met spatie tussen de duizendtallen en lijn ze recht uit.
 - Maak een header die **bold** is.
 - Sorteer het overzicht op Surface Area van grootste land naar kleinste.

Het overzicht komt er dus zo uit te zien:

Naam	Oppervlakte
Russian Federation	17 075 400
Ukraine	603 700
France	551 500
Spain	505 992
Sweden	449 964

Je hebt geleerd dat je een overzicht kan maken via een (Yii-)query. Met de query kan je de selectie en sortering aanpassingen. De query wordt geplaatst in de *controller* en maakt gebruik van het model. In dit voorbeeld het model van *country*.

Inleveren

- Jouw aangepaste en werkende view/country/overzicht.php in tabel vorm, en
- Een schermafdruck yii-04a-jouw-naam van jouw gehele browser waarin je je overzicht laat zien.

Opdracht 4b

Pas het overzicht van de vorige opdracht aan zodat je alleen alle kleine Europese landen ziet.

Een land is klein als `SurfaceArea < 100000`

Pas hiervoor de controller aan.

Je hebt dus te maken met **twee** zoekcriteria.

Zoek zelf op internet uit hoe je in een 'Yii-query' twee zoekcriteria kan combineren met *andWhere()* of gebruik de query methode om met Yii data uit de database te halen.

Inleveren

- Jouw aangepaste en werkende countryController.php

--

Revision #4

Created 22 June 2022 19:48:52 by Max

Updated 15 July 2022 09:37:12 by Max