



Beginnen met AngularJS

Luc Doumen

**DE HOGESCHOOL
MET HET NETWERK**

Hogeschool PXL – Elfde-Liniestraat 24 – B-3500 Hasselt
www.pxl.be - www.pxl.be/facebook



In dit hoofdstuk

- Een AngularJS-paginasjabloon instellen
- “Hello World !!!”
- De directive ng-repeat
- Gegevens filteren



Een AngularJS-paginasjabloon instellen

- Inleiding

```
<!DOCTYPE html>
<html>
<head>
  <title>Template AngularJS</title>
  <link href="../../css/bootstrap.css" rel="stylesheet" />
</head>
<body>
  <div class="container">
    <h1>{{ 2 + 2 }}</h1>
  </div>
  <script src="../../js/vendor/angular/angular.js"></script>
</body>
</html>
```



Een AngularJS-paginasjabloon instellen

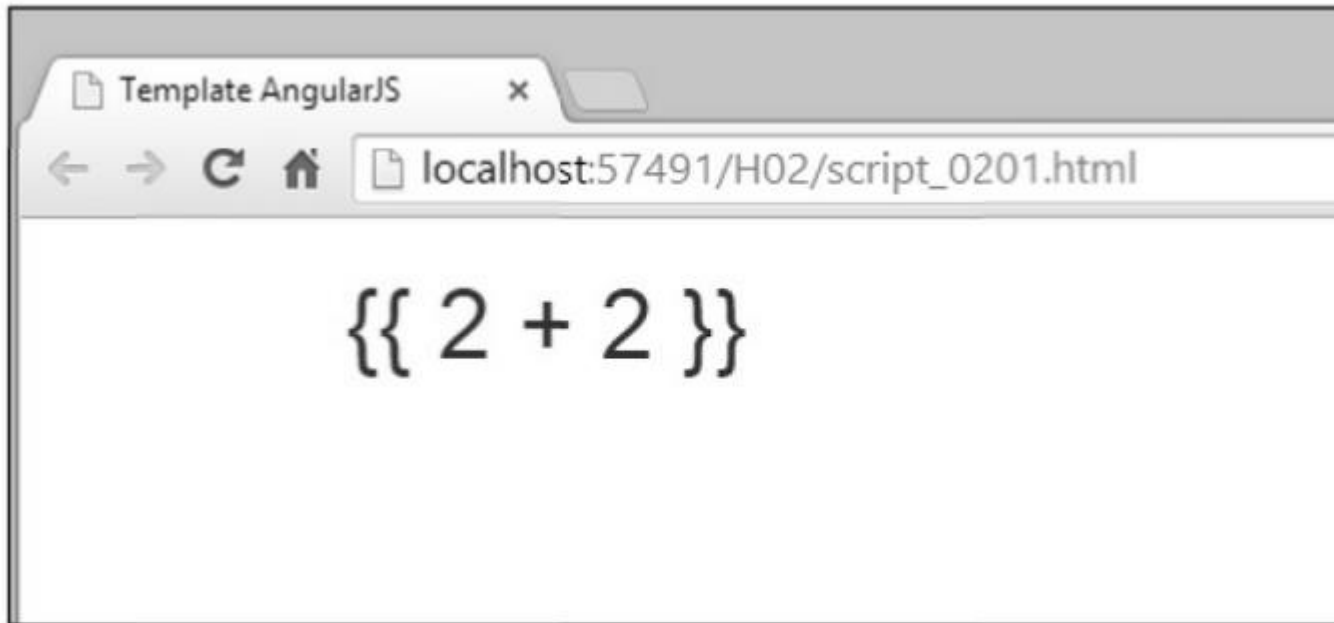
- **Inleiding**

- Voor eenvoudige CSS-opmaak is Twitter Bootstrap toegevoegd aan de header.
- In de HTML van de pagina wordt de expressie $2 + 2$ uitgerekend (bedenk: binnen `{{ ... }}` mag een JavaScript-expressie staan, dus ook een eenvoudige rekensom).
- Aan het einde wordt het Angular-framework geladen.



Een AngularJS-paginasjabloon instellen

- Inleiding



De som wordt niet uitgerekend. AngularJS lijkt niet goed te werken. Wat is er fout ?

Een AngularJS-paginasjabloon instellen

- Directive ng-app
 - In de inleiding werkt het voorbeeld niet, omdat Angular niet weet welk deel van de pagina onder handen wordt genomen. Dit wordt aangegeven door een “*directive*”.
 - De directive ng-app geeft aan welk onderdeel tot het angular-domein behoort. Alles daarbinnen wordt geëvalueerd, alles erbuiten niet.



Een AngularJS-paginasjabloon instellen

- Directive ng-app
 - Het is gebruikelijk om de ng-app toe te voegen aan de tag <HTML> of aan de <BODY> van de pagina

```
<html ng-app>  
...  
</html>
```

```
<body ng-app>  
...  
</body>
```

Een AngularJS-paginasjabloon instellen

- Directive ng-app



Na het toevoegen van de directive ng-app werkt de applicatie correct ...

Een AngularJS-paginasjabloon instellen

- Notatiewijzen voor directives

Directives die met AngularJS worden meegeleverd, beginnen allemaal met `ng-`. Zo kunt u standaarddirectives herkennen.

- Directives worden toegevoegd als attribuut van een element.
- Om HTML-validators tevreden te stellen, is het toegestaan de directive-naam vooraf te laten gaan door `data-ng-*` (voor HTML-validators) of door `x-ng-*` (voor XML-validators). Het eventuele voorvoegsel wordt door Angular gestript.
- Deze schrijfwijzen betekenen dus allemaal hetzelfde:
 - `ng-app`
 - `data-ng-app`
 - `x-ng-app`
- In dit boek gebruiken we de korte notatiewijze, dus zonder `data-`.



Een AngularJS-paginasjabloon instellen

- Notatiewijzen voor directives
-



Modulenaam voor ng-app

Zodra de app wat uitgebreider wordt, is het gebruikelijk om hem een naam te geven. De naam geeft u op als waarde voor het attribuut `ng-app`, bijvoorbeeld als `ng-app="myApp"`. Zodra u dit doet, bent u echter ook *verplicht* JavaScript-code te schrijven waarin de module `myApp` wordt gedefinieerd. Dat doen we vanaf het volgende hoofdstuk. Voor nu mag u eenvoudig `ng-app` schrijven, of `ng-app=""`.

Hello World !!!

- Directive ng-model
 - Alles tussen {{ ... }} wordt door AngularJS geëvalueerd en direct in het DOM geplaatst
 - Een data binding-expressie, die gelijk is aan een waarde in het model, wordt bijgewerkt zodra die waarde wijzigt in het model

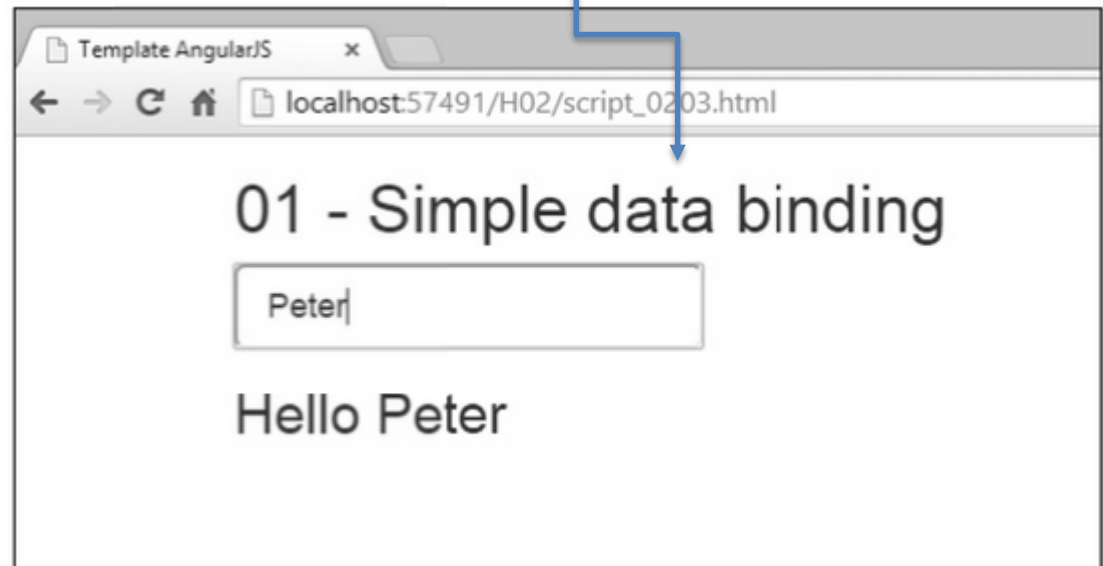


Hello World !!!

- Directive ng-model
 - Een eenvoudig HTML-formulier met een tekstveld

```
<input type="text" ng-model="firstName"
      class="input-lg" placeholder="typ je naam..." />
<h2>Hello {{ firstName }} </h2>
```

H02/Script_0203.html



Hello World !!!

- Meer over de directive ng-model
 - De directive is beschikbaar voor alle typen HTML-invoervelden
 - Het is de motor in veel data-driven Angular-apps

```
<input type="checkbox"
      value="yellow"
      ng-model="color" />Geel
<p>Gechecked : {{ color }}</p>
```

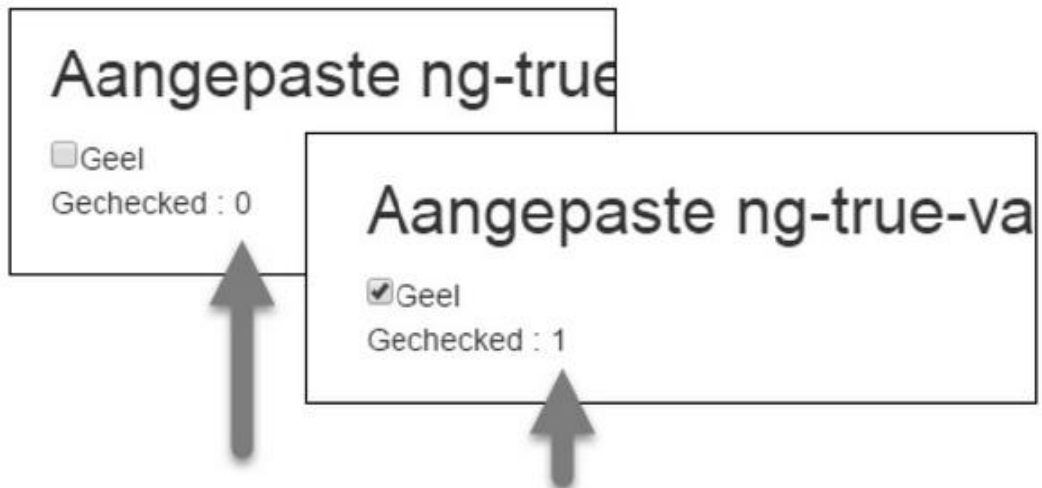


Data binding voor een selectievakje ...

Hello World !!!

- ng-true-value en ng-false-value

```
<input type="checkbox"
  value="yellow"
  ng-model="colorWithValue"
  ng-true-value="1"
  ng-false-value="0" />Geel
<p>Gechecked : {{ colorWithValue }}</p>
```



De waarden voor True en False zijn met directives zelf in te stellen ...

Hello World !!!

- ng-true-value en ng-false-value

```
<input type="checkbox"
  value="yellow"
  ng-model="colorWithValue"
  ng-true-value="'Geel is geselecteerd'"
  ng-false-value="'Geel is niet geselecteerd'" />Geel
<p>Gechecked : {{ colorWithValue }}</p>
```



Een aangepaste string-waarde voor ng-true-value en ng-false-value

Hello World !!!

- Samenvatting: directives & data binding-expressies



De directive ng-repeat

- In applicaties worden vaak **for-** en **while-lussen** gebruikt (iteratie)
- Angular kent de directive ng-repeat om over een collectie in het DOM te itereren



De directive ng-repeat

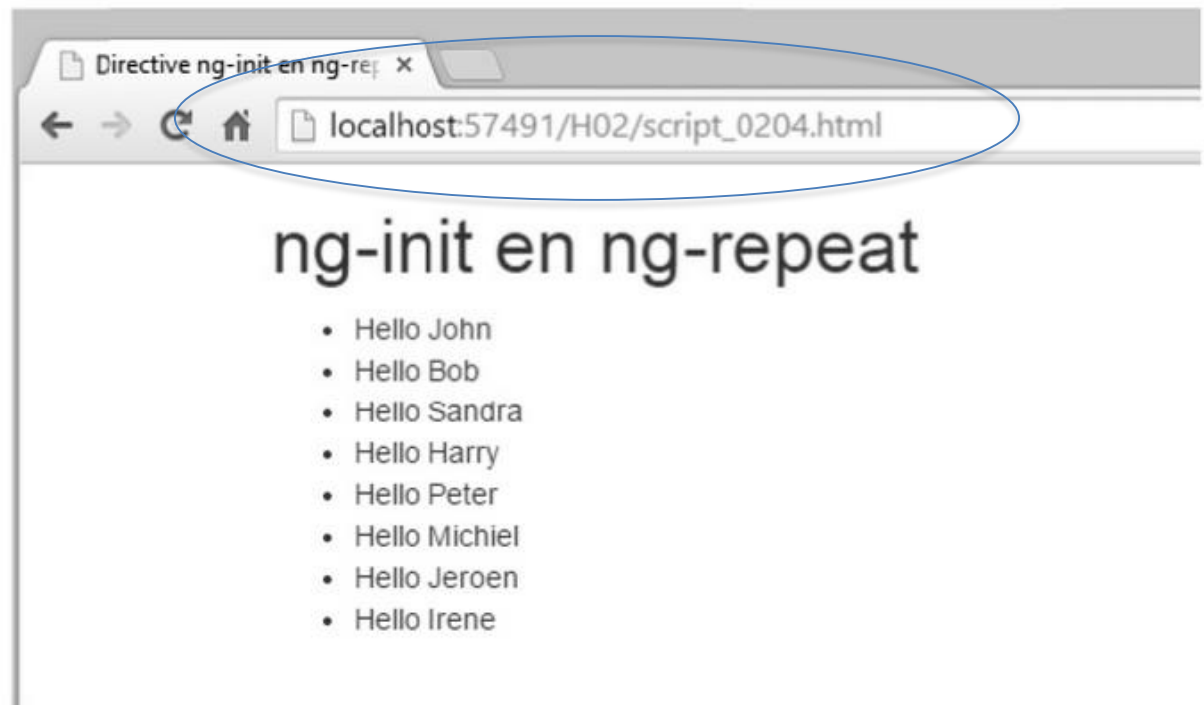
- Voorbeeld
 - Een array met namen die in een HTML-lijst (``) wordt getoond

```
<div class="container"
  ng-init="names=['John', 'Bob', 'Sandra', 'Harry', ...]">
  <h1>ng-init en ng-repeat</h1>
  <!-- directive ng-repeat voor itereren over een datacollectie -->
  <ul>
    <li ng-repeat="firstName in names">Hello {{ firstName }}</li>
  </ul>
</div>
```



De directive ng-repeat

- Voorbeeld
 - Een array met namen die in een HTML-lijst (``) wordt getoond



De directive ng-repeat

- Voorbeeld
 - Een array met namen wordt in een HTML-lijst getoond
-



Directive ng-init

We gebruiken in dit codevoorbeeld `ng-init` om even snel, *hardcoded* de beginwaarde voor de array names op te geven. In echte applicaties zullen deze namen uiteraard uit een database of van elders komen. `Ng-init` wordt in live-apps niet gebruikt. Maar voor demo-doeleinden is deze directive goed geschikt.

De directive ng-repeat

- Notatie van ng-repeat

De notatie voor ng-repeat is `ng-repeat="firstName in names"`.

- Hierbij is `firstName` de variabele die ter plekke wordt gemaakt.
- Bij elke lusdoorgang wordt `firstName` gevuld met het volgende element uit de array `names`.
- Deze array werd bij `ng-init` geïnitieerd.

Zo kunt u een array van eenvoudige elementen op het scherm tonen. In meer complexe apps krijgt u echter ook al snel te maken met arrays met objecten.



De directive ng-repeat

- ng-repeat bij objecten
 - Een array met objecten wordt in een HTML-lijst getoond

```
<div class="container"
  ng-init="persons=[{name : 'John', lastname:'Visser'}, {name : 'Bob', lastname:'Tasseur'}, ...]">
  <h1>ng-repeat en objecten</h1>
  <ul>
    <li ng-repeat="person in persons">
      {{ person.name }} - {{ person.lastname }}
    </li>
  </ul>
</div>
```



De directive ng-repeat

- ng-repeat bij objecten
 - Een array met objecten wordt in een HTML-lijst getoond

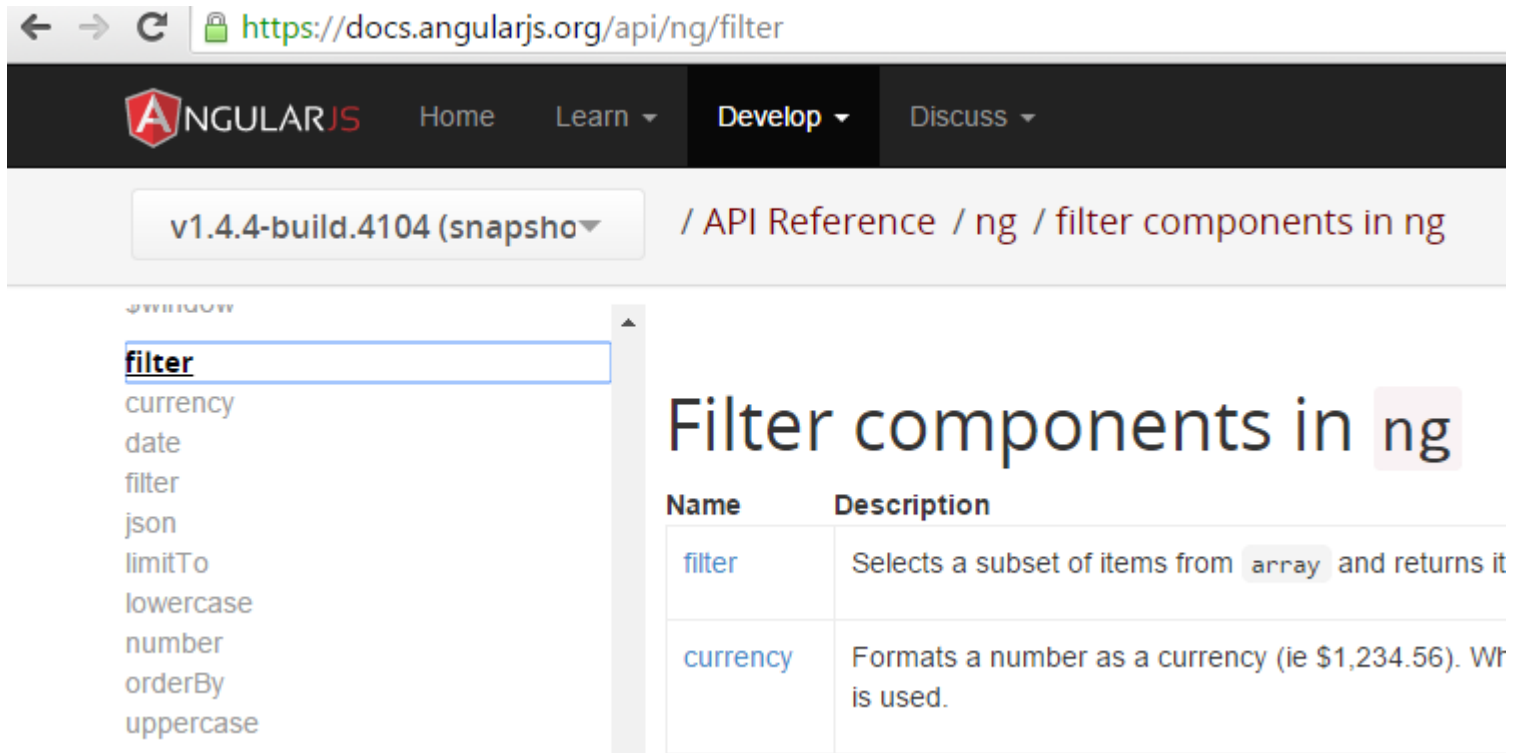


De directive ng-repeat

- ng-repeat bij objecten
 - Een array met objecten wordt in een HTML-lijst getoond
 - In de ng-repeat wordt in elke lusdoorgang een variabele `person` gemaakt op basis van het volgende object uit de array `persons`. We benaderen de eigenschappen van elke `person` via de dot-notatie `person.name` en `person.lastname`.
 - In dit geval had de data binding-expressie ook als één statement kunnen worden geschreven: `{{ person.name + ' - ' + person.lastname }}`.



Gegevens Filteren



The screenshot shows the AngularJS documentation page for filter components. The browser address bar displays <https://docs.angularjs.org/api/ng/filter>. The navigation bar includes the AngularJS logo, links for Home, Learn, Develop, and Discuss, and a version selector set to v1.4.4-build.4104 (snapshot). The breadcrumb trail indicates the path: / API Reference / ng / filter components in ng. On the left, a sidebar lists various filters, with 'filter' selected. The main content area is titled 'Filter components in ng' and contains a table with two entries: 'filter' and 'currency'.

Name	Description
filter	Selects a subset of items from <code>array</code> and returns it
currency	Formats a number as a currency (ie \$1,234.56). Wr is used.

Angular heeft 9 ingebouwde filters, aan te vullen met eigen geschreven (custom) filters



Gegevens Filteren

- Door een filter toe te passen op de gegevens is sneller een zoekresultaat zichtbaar
- Filters worden in Angular aangegeven door een “|”
- Wordt toegepast in data binding-expressies en in directives



Gegevens Filteren

- Uppercase en lowercase
 - Het resultaat van de data binding wordt geconverteerd naar hoofdletters of kleine letters

```
<li ng-repeat="person in persons">  
  {{ person.name | uppercase }} - {{ person.lastname | lowercase }}  
</li>
```



Gegevens Filteren

- orderBy
 - Een verzameling gegevens wordt gesorteerd op een bepaalde sleutel. Werkt het beste voor arrays

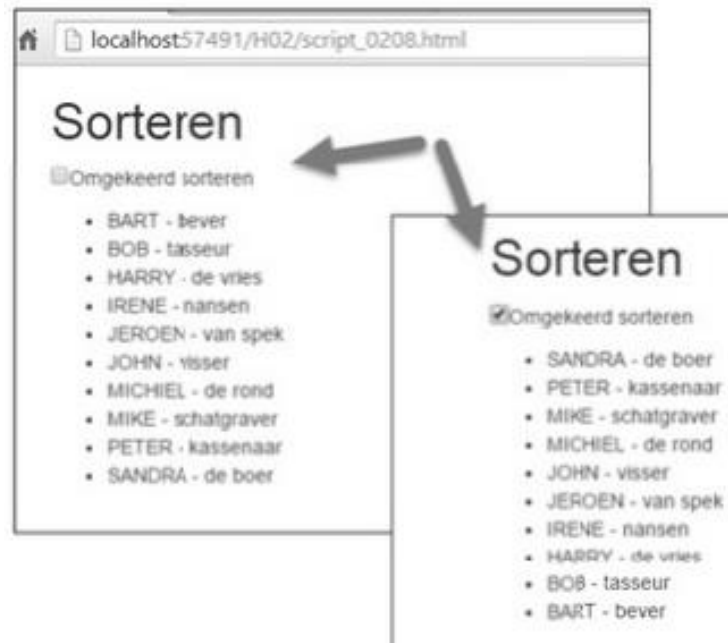
```
<li ng-repeat="person in persons | orderBy : 'name' ">  
  {{ person.name | uppercase }} - {{ person.lastname | lowercase }}  
</li>
```



Gegevens Filteren

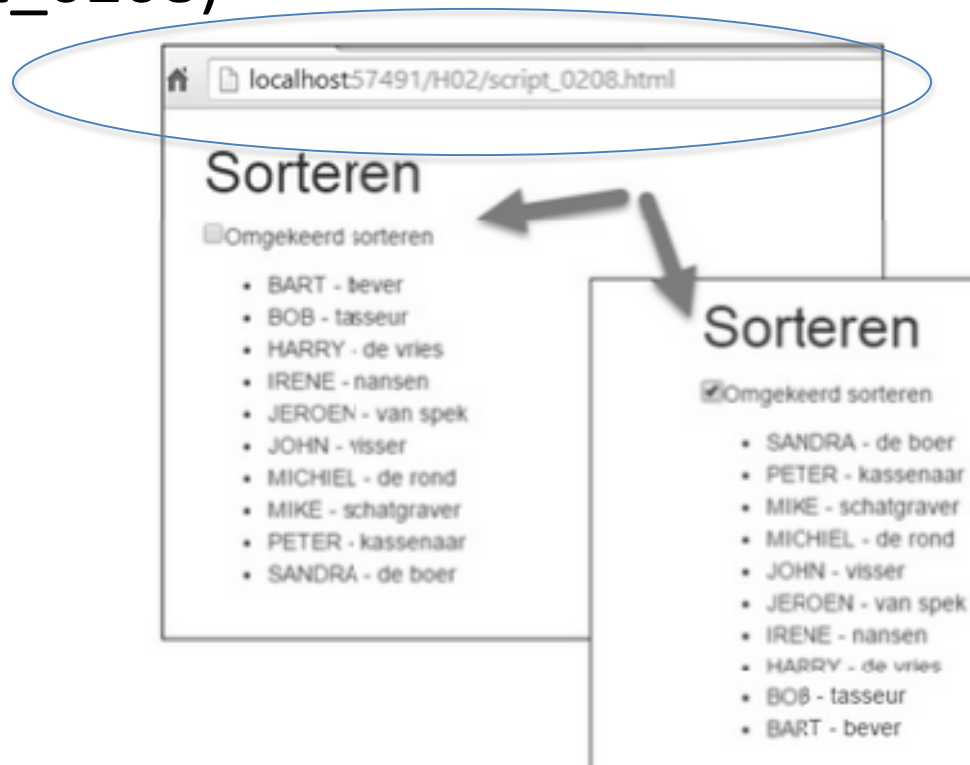
- Omgekeerd sorteren
 - OrderBy heeft een tweede parameter, 'reverse'.
True-waarde sorteert in omgekeerde volgorde.
Default waarde : 'false'

```
<li ng-repeat="person in persons | orderBy : 'name' : true ">
```



Gegevens Filteren

- Omgekeerd sorteren
 - Voorbeeld ‘omgekeerd sorteren’, variabele gebonden aan een invoerveld via ng-model (script_0208)



Gegevens Filteren

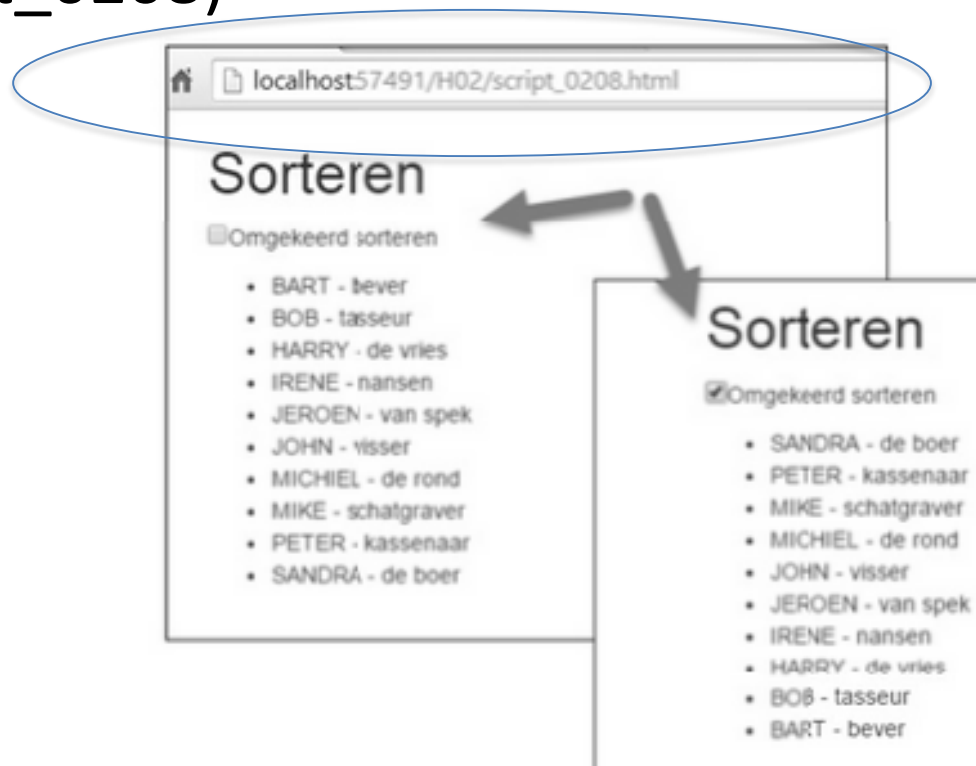
- Omgekeerd sorteren
 - Voorbeeld 'omgekeerd sorteren', variabele gebonden aan een invoerveld via ng-model (script_0208)

```
<p>
  <input type="checkbox" ng-model="reversed" />Omgekeerd sorteren
</p>
<ul>
  <li ng-repeat="person in persons | orderBy : 'name' : reversed ">
    {{ person.name | uppercase }} - {{ person.lastname | lowercase }}
  </li>
</ul>
```



Gegevens Filteren

- Omgekeerd sorteren
 - Voorbeeld ‘omgekeerd sorteren’, variabele gebonden aan een invoerveld via ng-model (script_0208)



Gegevens Filteren

- Gegevens filteren
 - Angular filter met de naam 'filter'
 - De array bevat uiteindelijk de items waarvoor het filter de waarde true heeft (script_0209)



Gegevens Filteren

- Gegevens filteren
 - Angular filter met de naam 'filter'
 - De array bevat uiteindelijk de items waarvoor het filter de waarde true heeft (script_0209)

```
<p>  
  <input type="text" ng-model="naam" class="input-lg" placeholder="filter op naam..." />  
</p>  
<ul>  
  <li ng-repeat="person in persons | filter : naam ">  
    {{ person.name }} - {{ person.lastname }}  
  </li>  
</ul>
```



Gegevens Filteren

- Filteren op veldnaam van object
 - Filteren gebeurt altijd op het volledige object
 - Specifiek filteren op een veld in het object gebeurt door het object mee te geven in de filterclausule (script_0210)

```
<li ng-repeat="person in persons | filter : { lastname: naam }">  
  {{ person.name }} - {{ person.lastname }}  
</li>
```



Gegevens Filteren

- Resultaat niet gevonden – ng-if
 - Toon een melding als er geen resultaten worden gevonden (script_0211)
 - De gefilterde array cachen in een nieuwe variabele

```
<ul>  
  <li ng-repeat="person in persons | filter:{lastname: naam} as numResults ">  
    {{ person.name }} - {{ person.lastname }}  
  </li>  
  <li ng-if="numResults.length === 0">  
    <strong>Geen resultaten gevonden...</strong>  
  </li>  
</ul>
```



Gegevens Filteren

- Overige filters

- **currency** Toon een getal als valuta. Standaard wordt het dollarteken gebruikt, maar u kunt een eigen valutasymbool opgeven.
- **date** Toon een datumveld volgens een bepaalde formaat-string. Zo is een datum bijvoorbeeld te formatteren in het formaat d-mm-yyyy en nog veel meer.
- **json** Formateer de uitvoer als JSON-string. Handig om snel het formaat van objecten te inspecteren (maar minder vaak gebruikt in live-apps).
- **limitTo** Maak een nieuwe array op basis van een expressie waarin een limiet (onder- of bovengrens) is opgegeven.
- **number** Formateer een getal als tekst, eventueel met cijfers achter de komma. Als de invoer geen getal is, wordt een lege string teruggegeven.



Zie de website 'docs.angularJS.org/api/ng/filter' voor meer informatie en codevoorbeelden

Oefening

- <https://bb.pxl.be>
 - Leermaterialen/Week 1/oefeningen
 - Oef => H2 – Beginnen met AngularJS

