

# Web Frameworks

## Angular – deel 2

Elektronica – ICT

Sven Mariën

(sven.marien01@ap.be)

# Inhoud

## 1. Deel 1

1. Waarom Angular ?
2. Nieuw Angular project
  - Aanmaken
  - Bestandsstructuur
3. Opbouw
  - Van een Applicatie
  - Van een Component
4. Nieuwe Component Aanmaken

## 2. Deel 2

1. Interpolation
2. Pipes

## 2.1. Interpolation

- Hiermee wordt het mogelijk om **gegevens vanuit onze klasse weer te geven** in de template.
- Interpolation werkt **enkel** voor **weergave** (one-way binding)
- Met interpolation kunnen **properties** en/of **methods** worden aangesproken
- Ook eenvoudige **expressies of concatenaties** zijn mogelijk
- Gebruik steeds **dubbele accolades** voor en na de expressie of binding `{{...}}`

### Interpolation

#### Template

```
<h1>{{pageTitle}}</h1>
{{'Title: ' + pageTitle}}
{{2*20+1}}
{{'Title: ' + getTitle()}}
<h1 innerText={{pageTitle}}></h1>
```

#### Class

```
export class AppComponent {
  pageTitle: string =
    'Acme Product Management';
  getTitle(): string {...};
}
```

# Inhoud

## 1. Deel 1

1. Waarom Angular ?
2. Nieuw Angular project
  - Aanmaken
  - Bestandsstructuur
3. Opbouw
  - Van een Applicatie
  - Van een Component
4. Nieuwe Component Aanmaken

## 2. Deel 2

1. Interpolation
2. Pipes

## 2.2. Pipes

`<h2>{{datum | date : "hh:mm:ss"}}</h2>`

- De gegevens die worden weergegeven via 'Interpolation' moeten soms nog extra formattering (= opmaak) ondergaan.
- Een mogelijkheid is het gebruik van 'Pipes'.
  - Deze transformeren de data voordat ze wordt weergegeven
  - Er bestaan:
    - Ingebouwde (built-in) Pipes
    - Zelfgemaakte (custom) pipes
  - Built-in
    - Formatteren van tekst, datums, getallen, geld, json data, enz...
  - Meerdere pipes kunnen na elkaar worden gebruikt

USD45.00

€45.00

EUR45.00

€45.000

Oct 15, 2017

08:07

08:07:32

15/10/2017

15/Oct/17

Title

TITLE

title

# Pipes

- Gebruik van een pipe wordt aangegeven met het '|' teken
- **Tekst** formatteren: lowercase , uppercase pipes
  - `{{Title | uppercase}}`
- **Datum** formatteren : date pipe, deze kan met parameters worden ingesteld
  - `{{sunrise | date}}`
  - `{{sunrise | date: 'HH:mm'}}`
  - `{{sunrise | date: 'HH:mm:ss'}}`
- **Munteenheid** formatteren:
  - `{{Amount | currency}}`
  - `{{Amount | currency: 'EUR'}}`
  - `{{Amount | currency: 'EUR':true}}`
- **Getallen** formatteren:
  - `{{length | number}}`
  - `{{length | number: '2.2-5'}}`

Lijst van ingebouwde pipes:

<https://angular.io/api?type=pipe>

# Oefeningen: Interpolation & pipes

- Maak telkens een component die:
  - De huidige tijd weergeeft
    - in het formaat 'HH:mm:ss'
    - De tijd wordt uiteraard elke seconde up-to-date gebracht
  - Een teller weergeeft
    - Die elke 200ms verhoogt in waarde
    - Gebruik een get'ter om de teller op te vragen want de teller zelf is private.
  - Een willekeurige waarde weergeeft
    - tussen 1 en 100
    - Elke 2 seconden wordt een andere willekeurige waarde weergegeven
    - (kijk ook eens of de lodash module je hierbij kan helpen)
  - Een tekst weergeeft:
    - "De applicatie werd opgestart: [xx minuten geleden]"
    - Uiteraard wordt deze automatisch up-to-date gebracht
    - Gebruik **moment.js** hiervoor

# Oefeningen:

- Maak een component die:
  - Een (HTML) tabel weergeeft
    - van 3 rijen en 3 kolommen gevuld met de cijfers 1-9
    - Maak hiervoor in de klasse een array die je initialiseert met de cijfers 1-9
    - Gebruik een methode om vanuit interpolation aan de gegevens te geraken van de array
    - (dus de array zelf maak je private)



# Oefeningen:

- Maak een component die
  - Een Tabel weergeeft.
    - Maak een klasse die deze interface IWeather implementeert.
    - Maak een array met 2 weather objecten en stel de locaties in hoofdletters en de temperaturen in graden Fahrenheit in.
    - Geef de 2 objecten weer in de tabel
    - kolom 1: locatie: met Eerste letter een hoofdletter weergeven
    - kolom 2: temperatuur, zet deze met een formule om van F naar C
    - kolom 3: zonsopkomst : uiteraard enkel de tijd weergeven

```
interface IWeather
{
    location : string;
    description : string;
    temperature : number;
    sunrise : Date;
    sunset : Date;
}
```

# Oefeningen

- Maak een component die:
  - Een teller weergeeft
  - Deze teller verhoogt elke seconde met 1
  - Geef de teller weer op het scherm
    - Met 2 cijfers voor de komma indien de teller  $< 10$
    - Met 4 cijfers voor de komma indien de teller  $\geq 10$

