

Angular

TypeScript

- Superset JavaScript
- Types toegevoegd
- .ts extensie
- Wordt gecompileerd naar JavaScript

Node.js

- Server side JavaScript framework

Angular CLI

- A command line interface for Angular
- Functionaliteit
 - Webserver starten voor testing (ng serve)
 - Angular componenten, services etc aanmaken
 - ng g c/s naam
 - Installeren: npm i -g @angular/cli

Wat is Angular?

- Basisidee: webpagina bestaat uit componenten (tags)

Component

- Bouwsteen voor een webpagina
- Bestaat uit
 - Klasse: functionaliteit
 - HTML-template (sjabloon): presentatie
 - Stylesheet: opmaak

Component - interpolation binding

HTML-template

```
<div class="weer">
  <div> {{datum}} </div>
  ...
</div>
```

Klasse: attributen - methodes

```
@Component({
  selector: 'app-weerinfo',
  templateUrl: '....html',
  styleUrls: ['....css']
})
export class WeerinfoComponent {
  datum = '26/10/17';
  ...
}
```

Component registreren

In app.module.ts bij declarations de component toevoegen

Structural directives

- *ngIf
- *ngFor
- OnInit moet geïmporteerd worden
- ngOnInit() moet gedefinieerd worden

One Way Property Binding

- Koppeling tussen eigenschap klasse en attribuut element
- 3 mogelijkheden:
 - `img src="{{afbeelding}}"`
 - `img [src]="afbeelding"`
 - `img bind-src="afbeelding"`

Two Way Property Binding

- Koppeling tussen attribuuttemplate en instantievariabele klasse

```
<select [(ngModel)]="selectedLand">
  <option *ngFor="let land of landen" [ngValue]="land">
    {{land.naam}}
  </option>
</select>
```

```
export class LandenComponent {
  selectedLand: Land;
  private _landen: Land[];
  get landen(): Land[] {
    return this._landen;
  }
}
```

- [(ngModel)]
 - Tweewegsbinding
 - Property/Field <-> waarde HTML-component
- Behoort tot de module FormsModule

Component - events

```
<select (change)="veranderdLand()">
  ...
</select>
```

```
export class LandenComponent {
  veranderdLand() {
    ...
  }
}
```

Component: @Input() - attributen toevoegen

Tag-component

```
<app-landen landen="..."> </app-landen>
```

Klasse-component

```
export class LandenComponent implements OnChanges{
  @Input()
  landen: Land[] = [];
  ngOnChanges(changes: SimpleChanges): void {
    if (this.landen !== [] && this.selectedLand == undefined ){
      this.selectedLand = this.landen[0];
    }
  }
}
```

Component: @Output() - event toevoegen

Tag-component

```
<app-landen (landChanged)="..."> </app-landen>
```

Klasse-component

```
export class LandenComponent {  
  selectedLand: Land;  
  @Output() landChanged = new EventEmitter<Land>();  
  veranderdLand() {  
    this.landChanged.emit(this.selectedLand);  
  }  
}
```

Service

- @Injectable() importeren
- in app.module.ts registreren bij providers

HTTP-service

- constructor(private http: HttpClient) { }
- Interface REST-resultaat aanmaken
- .pipe(tap(doe iets),map(items => items.map(item => new Interface(item.parameters))));

Routing

- in aparte module
- href vervangen door routerLink
- router-outlet toevoegen

Routing met parameters

- In module met routerconfiguratie

```
{  
  path: 'detail/:id',  
  component: Component  
}
```