

# DIRECTIVES

ULT – DÉPARTEMENT GÉNIE INFORMATIQUE



Module: Projet Framework et Développement

Public cible: 5° Génie Logiciel

Houcem Hedhly

[houssem.hedhli@ult-tunisie.com](mailto:houssem.hedhli@ult-tunisie.com)

2021/2022

# PLAN

- I. Introduction
- II. Les Directives
- III. ngIf Directive
- IV. ngSwitch Directive
- V. ngFor Directive

# INTRODUCTION

- Les directives sont des instructions dans le DOM.
- Les directives dans Angular sont utilisées pour manipuler le DOM.
- En utilisant des directives, vous pouvez modifier l'apparence, le comportement ou la présentation d'un élément DOM.
- Les directives dans Angular peuvent être classées en 2 catégories en fonction de leur comportement:
  - Structural Directives
  - Attribute Directives

## LES DIRECTIVES

- **Structural Directives:** Les directives structurelles commencent par un signe \*.
- Ces directives sont utilisées pour manipuler et changer la structure des éléments DOM.
  - **\*ngIf** : permet d'ajouter ou de supprimer un élément DOM selon l'évaluation d'une condition.
  - **\*ngSwitch** : permet d'afficher un élément DOM parmi une liste de possibilités selon l'évaluation d'une condition.
  - **\*ngFor** : utilisée pour répéter une partie du Template HTML une fois pour chaque élément d'une liste itérable (Collection).

# LES DIRECTIVES

- **Attribute Directives:**
- Utilisées pour détecter et modifier le comportement des éléments HTML, attributs, propriétés et composants.
  - **NgClass** : utilisée pour ajouter ou supprimer un ensemble de classes CSS.
  - **NgStyle** : utilisée pour ajouter ou supprimer une définition de style.
  - **NgModel** : utilisée pour ajouter le two-ways binding.

# NGIF DIRECTIVE

```
<p *ngIf="serverCreated">Server was created! server name is {{ serverName }}</p>
```

\* : Structural directive

ngIf : Directive

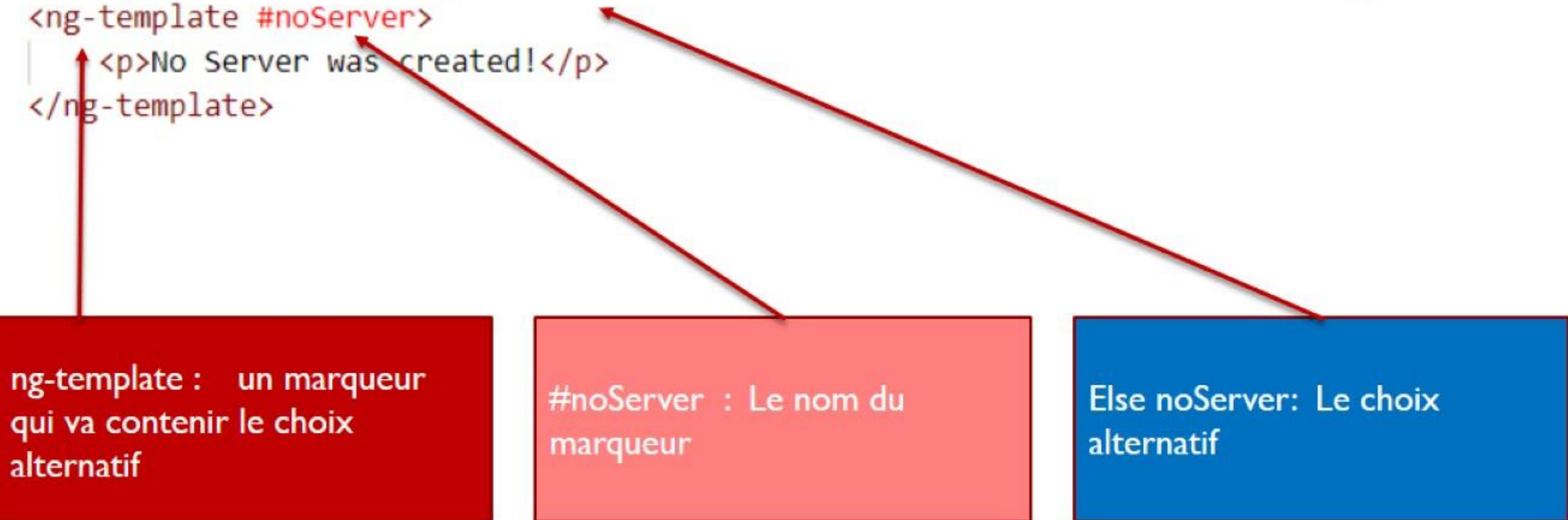
**serverCreated** : ça doit être une valeur booléenne.  
true : l'élément sera affiché  
false: l'élément ne sera pas affiché



# NGIF ELSE

- On peut ajouter un choix alternatif à la directive ngIf

```
<p *ngIf="serverCreated; else noServer">Server was created! server name is {{ serverName }}</p>  
<ng-template #noServer>  
| <p>No Server was created!</p>  
</ng-template>
```



ng-template : un marqueur  
qui va contenir le choix  
alternatif

#noServer : Le nom du  
marqueur

Else noServer: Le choix  
alternatif

# NGIF THEN ELSE

- Une autre notation

```
<div *ngIf="displayName; then thenBlock; else elseBlock"></div>
<ng-template #thenBlock>
  <h2>
    Angular
  </h2>
</ng-template>
<ng-template #elseBlock>
  <h2>
    Name is hidden
  </h2>
</ng-template>
```



# NGSWITCH DIRECTIVE

```
public color = "red";
```

```
<div [ngSwitch]="color" [style.color]="color">  
  <div *ngSwitchCase="'red'">You Picked red color</div>  
  <div *ngSwitchCase="'blue'">You Picked blue color</div>  
  <div *ngSwitchCase="'green'">You Picked green color</div>  
  <div *ngSwitchDefault>Pick again</div>  
</div>
```

Structural directive

Si la variable « color » = 'red' alors l'élément correspondant sera affiché

Choix par défaut

# NGFOR DIRECTIVE

```
public colors = ["red", "blue", "green", "yellow"];
```

colors : une collection (un tableau)  
d'éléments

```
<div *ngFor="let color of colors">  
  <h2>{{ color }}</h2>  
</div>
```

\* : Structural directive  
ngFor: Directive  
let color of colors: on parcourt la  
collection « colors » et pour chaque  
élément on affiche l'élément <h2>  
correspondant.

**red**

**blue**

**green**

**yellow**

# NGFOR INDEX

- On peut aussi accéder à l'index du tableau

```
public colors = ["red", "blue", "green", "yellow"];
```

colors : une collection (un tableau)  
d'éléments

```
<div *ngFor="let c of colors; index as i">  
  <h2>{{ i }} {{ c }}</h2>  
</div>
```

\* : Structural directive  
**ngFor**: Directive  
let color of colors: on parcourt la  
collection « colors » et pour chaque  
élément on affiche l'élément <h2>  
correspondant.

**0 red**

**1 blue**

**2 green**

**3 yellow**

# NGFOR DIRECTIVE

- Le même exemple mais en ajoutant un style Binding

```
public colors = ["red", "blue", "green", "yellow"];
```

colors : une collection (un tableau)  
d'éléments

```
<div *ngFor="let c of colors">  
  <h2 [style.color]="c">{{ c }}</h2>  
</div>
```

Style Binding

red

blue

green

yellow