



Angular 4

directives d'attribut

vertika.org by Jean Garutti

-- Angular 4 --

03/09/2017 -- 1

info@vertika.org

Les directives d'attribut



- Vous êtes en plein dans le *LANGAGE OBJETS*
- Angular l'améliore encore

by Jean Garutti

Jean Garutti – vertika.org

-- Angular 4 --

info@vertika.org

des attributs légers...



by Jean Garutti

-- Angular 4 --

Directives d'attributs

3

ATTRIBUTE DIRECTIVE

Changes the appearance or behavior of an element

```

import { Directive, ElementRef, HostListener } from '@angular/core';
@Directive({
  selector: '[appBoldText]'
})
export class BoldTextDirective {
  constructor(private elementRef: ElementRef) { }
  @HostListener('mouseenter') onMouseEnter() {
    this.elementRef.nativeElement.style.fontWeight = 'bold';
  }
  @HostListener('mouseleave') onMouseLeave() {
    this.elementRef.nativeElement.style.fontWeight = null;
  }
}

```

Importing Directive, ElementRef & HostListener

Directive Metadata

Injecting ElementRef to access the DOM elements

Bold the text on cursor hover

Un-bold the text

by Jean Garutti

-- Angular 4 --

-- 4

Directives d'attributs

- Vous pouvez créer
- vos propres directives d'attributs
- ou vous pouvez utiliser des directives Angular intégrées.

by Jean Garutti

-- Angular 4 --

-- 5

Directives d'attributs exemples

- <https://embed.plnkr.co/?show=preview>

by Jean Garutti

-- Angular 4 --

-- 6

Directives d'attributs

- La directive d'attribut requiert au minimum
- la construction d'une classe de contrôleur
 - annotée avec @Directive,
- qui spécifie le sélecteur identifiant l'attribut.
- La classe de contrôleur implémente le comportement de la directive désirée.

Directives d'attributs

- voici la construction d'une directive d'attributs simple
myHighlight
- pour définir la couleur d'arrière-plan d'un élément lorsque l'utilisateur survole sur cet élément.

Directives d'attributs

src/app/app.component.html

<p myHighlight>Highlight me!</p>

le code de la directive

- Suivez les instructions de configuration pour créer un nouveau projet local appelé **attribute-directives**.
- Créez le fichier source suivant :

le code de la directive

```
src/app/highlight.directive.ts
1. import { Directive, ElementRef, Input } from
   '@angular/core';

2. @Directive({ selector: '[myHighlight]' })
3. export class HighlightDirective {
4.     constructor(el: ElementRef) {
5.         el.nativeElement.style.backgroundColor
6.         = 'yellow';
7.     }
8. }
```

le code de la directive

- La déclaration **import** spécifie les symboles du noyau :
- 1. **directive** fournit la fonctionnalité du décorateur **@Directive**.
- 2. **ElementRef** s'injecte dans le constructeur de la directive afin que le code puisse accéder à l'élément DOM.
- 3. **Input** permet aux données de circuler à partir de l'expression de liaison dans la directive.

le code de la directive

- Après les métadonnées **@Directive**, la classe de contrôleur de la directive,
 - appelée **HighlightDirective**,
- contient la logique de la directive.
- L'exportation de **HighlightDirective** la rend accessible à d'autres composants.

le code de la directive

- Après les métadonnées **@Directive**, la classe de contrôleur de la directive,
 - appelée **HighlightDirective**,
- contient la logique de la directive.

le code de la directive

- L'exportation de **HighlightDirective** la rend accessible à d'autres composants.

Directives d'attributs

- Une illustration assez simple d'une directive d'attributs de base,
ngClass
- est ajouté à un élément composant
- et sa valeur est donnée en **classe**.

Directives d'attributs

1. `import {Component, CORE_DIRECTIVES}`
`from 'angular2/angular2'`

Directives d'attributs

3. `@Component({`
4. `selector: 'my-app',`
5. `directives:`
`[CORE_DIRECTIVES],`

Directives d'attributs

```
6.   template: `
7.     <div>
8.       <h2>{{title}}</h2>
9.       <div ngClass="gray-border purple">string of
      classes</div>
10.      <div [ngClass]="'gray-border purple'">string of
      classes</div>
11.      <div [ngClass]="['gray-border', 'green']">array of
      classes</div>
12.      <div [ngClass]="{'gray-border': true, 'blue':
      true}">object/map of classes</div>
13.      <button [ngClass]="{active: isActive}"
      (click)="isActive = !isActive">Click me</button>
14.    </div>
```

by Jean Garutti

-- Angular 4 --

-- 19

Directives d'attributs

```
16. styles: [`
17.   .gray-border { border: 1px solid gray;
18.   }
19.   .blue { color: blue; }
20.   .green { color: green; }
21.   .purple { color: purple; }
22.   .active { background-color: #f55; }
23. `]
```

by Jean Garutti

-- Angular 4 --

-- 20

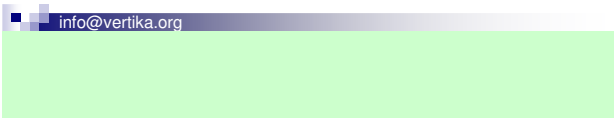
Directives d'attributs

```
24. export class App {
25.   title = "Angular 2 - NgClass";
26.   isActive = false;
27. }
```

by Jean Garutti

-- Angular 4 --

-- 21



Merci !

Thanks !
