

angular: directivas estructurales

- PPT 3.7 + 3.8 + 3.9 + 4.1 + 4.2 + 4.3
- lógica en capa de presentación (plantilla)
- se reconocen por tener * en el nombre
- no es necesario fichero HTML
- no es un componente

ngIf

- elemento que queremos ocultar o mostrar
- se le asigna alguna expresión que retorne un booleano
- angular crea o descrea el elemento del DOM según marque la variable:

```
<div *ngIf="mostrar" class="card text-white bg-primary mb-3" style="max-width: 100%; ">
```

- **mostrar** al estar vinculado a la directiva hace referencia a un atributo de la clase, si no, iría entre `{{ }}`
- binding de eventos:

```
<button (click)="mostrar = !mostrar" class="btn btn-outline-primary btn-block">
```

ngFor

- repite el elemento:

```
<li *ngFor="let color of colores" class="list-group-item">{{color}}</li>
```

- usamos un array **colores** definido en la clase del componente:

```
export class BodyComponent {  
  mostrar = false;  
  colores: string[] = ['rojo', 'verde', 'azul'];  
  ...  
}
```

- se pueden adicionar cosas:

```
<li *ngFor="let color of colores; let i=index" class="list-group-item">{{i}}  
  {{color}}</li>
```

ngIf-else + ngTemplate

```
<div *ngIf="numero % 2 == 0; else impar">  
  <h1>Número {{ numero }} - Es par</h1>  
</div>
```

```
<ng-template #impar>
  <h1>Número {{ numero }} - Es impar</h1>
</ng-template>
```

```
<div *ngIf="numero % 2 == 0; then par else impar"></div>
```

```
<ng-template #par>
  <h1>Número {{ numero }} - Es par</h1>
</ng-template>
```

```
<ng-template #impar>
  <h1>Número {{ numero }} - Es impar</h1>
</ng-template>
```

ngStyle

- cambio de estilo de elementos HTML usando atributos de los componentes
- scope de actuación del CSS → propio del componente, no afecta a otros
- añadir/quitar clases CSS según nuestro internés
- directivas personalizadas

- `ng c[reate] c[omponent] components/<component> -it -is --spec=false`

- directorio **components** podría variar a nuestra conveniencia
- `-it:?`
- `-is:?`
- `-spec=false`: no crear ficheros de pruebas unitarias

```
...
template: `
```

```
template: `
```

```
template: `
```

ámbito CSS

- para evitar colisiones de estilos, angular se encarga de añadir atributos para que cada componente tenga sus propios estilos
- si queremos estilos a nivel global, deberemos tocar **style.css**

ngClass

- más usada y más flexibilidad que **ngStyle**
- sintaxis:

- `<div [ngClass] ='first second'>...</div>`

- `<div [ngClass] =["first', 'second']">...</div>`

- `<div [ngClass] ="{'first': true, 'second': true}">...</div>`

- `<div [ngClass] ="stringExp | arrayExp | objExp">...</div>`

- `<div [ngClass] ="{'class1 class2 class3':true}">...</div>`

procesos asíncronos (4.4)

ngSwitch

4.5 Directivas personalizadas

- `ng g[enerate] d[irective] directivas/destacado --spec=false`

From:

<https://miguelangel.torresegea.es/wiki/> - miguel angel torres egea

Permanent link:

<https://miguelangel.torresegea.es/wiki/development:angular:directivas?rev=1580138862>

Last update: **27/01/2020 07:27**

