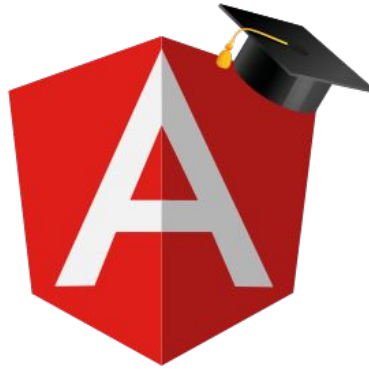


IV-Directives-2



On gère les style CSS avec Angular

en Javascript

```
document.querySelector('#myDiv').style.color = 'red';
```

```
document.querySelector('#myDiv').style.background = 'red';
```

Avec Angular

```
<div [style.background-color]=" 'red' " [style.color]=" 'white' ">My DIV to style with Angular</div>
```

Il est possible d'utiliser des variables comme valeur

LES DIRECTIVES (ngStyle et ngClass)

À la différence des directives structurelles, les directives par attribut modifient le comportement d'un objet déjà existant (ex: `ngModel`)

Dans ce cours, nous parlerons des **directives par attribut**.

ngStyle

L'utilisation de la Directive `NgStyle` permet d'assigner dynamiquement des valeurs pour changer le style d'un élément du DOM. C'est une manière très simple de modifier l'apparence de l'interface utilisateur en fonction de ces preference.

Modifier le fichier `mon-premier-component.component.html`

Ajouter ce code à `mon-premier-component.component.ts`

```
<li class="list-group-item">
  <div
    style="width: 20px; height: 20px;"
    [ngStyle]="{ background: getColor() }"
  ></div>
  <p>Etudiant : {{ persoName }} -- Status : {{ getStatus() }}</p>
</li>
```

```
getColor() {
  if (this.persoStatus === "Présent") {
    return "green";
  } else if (this.persoStatus === "Absent") {
    return "red";
  }
}
```

Résultat:

Liste



Etudiant : Emmanuelle -- Status : Présent



Etudiant : Robin -- Status : Absent



Etudiant : Dominique -- Status : Présent



Utilisateur connecté

ngClass

Pour modifier des styles directement, il peut être très utile d'ajouter des classes CSS à un élément de manière dynamique. Pour cela, nous pouvons utiliser la directive ngClass

Modifier le fichier `mon-premier-component.component.html`

```
<li
  [ngClass]="{
    'list-group-item': true,
    'list-group-item-success': persoStatus === 'Présent',
    'list-group-item-danger': persoStatus === 'Absent'
  }"
>
  <div
    style="width: 20px; height: 20px;"
    [ngStyle]="{ background: getColor() }"
  ></div>
  <p>Etudiant : {{ persoName }} -- Status : {{ getStatus() }}</p>
</li>
```

Dans cet exemple, on utilise les classes CSS de Bootstrap

Explication :

- La classe sera appliquée list-group-item
- Les classes list-group-item.success et list-group-item.danger seront appliquées si la condition qui leur a été attribuée est vraie

Résultat :

Liste



Etudiant : Emmanuelle -- Status : Présent



Etudiant : Robin -- Status : Absent



Etudiant : Dominique -- Status : Présent

☒ Utilisateur connecté