

## 2 - Afficher une liste de sélection (III)

### ETAPE 1

1 - Remplacer la variable hero par heroes. Il s'agit d'un tableau contenant 10 héros (voir le code qui suit):

```
heroes = [  
  { id: 11, name: 'Dr Nice' },  
  { id: 12, name: 'Narco' },  
  { id: 13, name: 'Bombasto' },  
  { id: 14, name: 'Celeritas' },  
  { id: 15, name: 'Magneta' },  
  { id: 16, name: 'RubberMan' },  
  { id: 17, name: 'Dynamia' },  
  { id: 18, name: 'Dr IQ' },  
  { id: 19, name: 'Magma' },  
  { id: 20, name: 'Tornado' }  
];
```

### 2 - Afficher la liste des héros

Ouvrez le fichier HeroesComponent et apportez les modifications suivantes:

- Ajoutez un <h2> en haut,
- En dessous, ajoutez une liste HTML non ordonnée ( <ul>)
- Insérez un <li> dans le <ul> qui affiche les propriétés d'un héro.

A l'aide de la directive \*ngFor, répétez l'élément <li> pour chaque élément de la liste heroes s'affiche

### 3 - Faire apparaître les détails correspondants à l'héro cliqué

- Ajouter une liaison d'événement de clic
- Ajouter une section de détails

*Astuce: Le composant ne doit afficher les détails du héros sélectionné que si celui-ci existe. Enveloppez le code HTML des détails du héros dans un fichier <div>. Ajoutez la directive \*ngIf.*

#### 4 - Ajoutez un style personnalisé pour les héros sélectionnés

- Dans `heroes.component.css`, ajouter le code css suivant

```
/* HeroesComponent's private CSS styles */
```

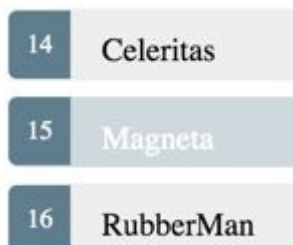
```
.heroes {  
  margin: 0 0 2em 0;  
  list-style-type: none;  
  padding: 0;  
  width: 15em;  
}  
.heroes li {  
  cursor: pointer;  
  position: relative;  
  left: 0;  
  background-color: #EEE;  
  margin: .5em;  
  padding: .3em 0;  
  height: 1.6em;  
  border-radius: 4px;  
}  
.heroes li:hover {  
  color: #607D8B;  
  background-color: #DDD;  
  left: .1em;  
}  
.heroes li.selected {  
  background-color: #CFD8DC;  
  color: white;  
}  
.heroes li.selected:hover {  
  background-color: #BBD8DC;  
  color: white;  
}  
.heroes .badge {  
  display: inline-block;  
  font-size: small;  
  color: white;  
  padding: 0.8em 0.7em 0 0.7em;
```

```
background-color:#405061;
line-height: 1em;
position: relative;
left: -1px;
top: -4px;
height: 1.8em;
margin-right: .8em;
border-radius: 4px 0 0 4px;
}
```

## ETAPE 2

1 - Il est difficile d'identifier le héros sélectionné dans la liste lorsque tous les éléments `<li>` se ressemblent.

Si l'utilisateur clique sur "Magneta", ce héros doit être rendu avec une couleur d'arrière-plan qui suit :



Cette coloration de héros sélectionnée est le travail de la classe CSS `.selected` dans le code que vous avez ajoutés précédemment.

Il vous suffit d'appliquer la classe `.selected` au `<li>` lorsque l'utilisateur clique dessus.

La liaison de classe facilite l'ajout et la suppression d'une classe CSS de manière conditionnelle. Ajoutez simplement `[class.selected]="ici_une_condition"` à l'élément que vous souhaitez styliser.

Ajoutez la liaison `[class.selected]` suivante au `<li>` dans le `HeroesComponent` : exemple:

```
[class.selected]="hero === selectedHero"
```

Lorsque le héros de la ligne actuelle est le même que le `selectedHero`, Angular ajoute la classe CSS `selected`. Lorsque les deux héros sont différents, Angular supprime la classe.

2 - Quand on clique sur un nom, les détails du héros sélectionné s'affiche en bas

Résultat :

## Tour of Heroes

### My Heroes

11	Dr Nice
12	Narco
13	Bombasto
14	Celeritas
15	Magneta
16	RubberMan
17	Dynama
18	Dr IQ
19	Magma
20	Tornado

### BOMBASTO Details

id: 13

name: