CODE:

```
TS app-module.ts
app.html
                TS app.ts
                                                    heroes.htn
                            ×
my-angular-app ♭ src ♭ app ♭ TS app.ts ♭ ...
       import { Component, signal } from '@angular/core';
       import {Heroes} from './heroes/heroes'
       import {CommonModule} from '@angular/common'
       @Component({
         selector: 'app-root',
        templateUrl: './app.html',
        standalone: true,
        styleUrl: './app.css',
         imports: [CommonModule, Heroes]
 10
 11
       })
       export class App {
 12
       title = 'my-angular-app';
 13
 14
 15
```

```
app.html
               TS app.ts
                              TS app-module.ts × ♦ heroes.html
                                                                  TS heroes.ts
my-angular-app ▷ src ▷ app ▷ TS app-module.ts ▷ 🍖 AppModule
       import { NgModule, provideBrowserGlobalErrorListeners } from '@angular/core';
       import { BrowserModule } from '@angular/platform-browser';
       import { FormsModule } from '@angular/forms';
       import { AppRoutingModule } from './app-routing-module';
       import { App } from './app';
       import { Heroes } from './heroes/heroes';
      @NgModule({
         declarations: [
        ],
        imports: [
         BrowserModule,
 13
         ],
         providers: [
         provideBrowserGlobalErrorListeners()
         bootstrap: [App]
       export class AppModule { }
                             TS app-module.ts
                                                ♦ heroes.html × TS heroes.ts
app.html
              TS app.ts
my-angular-app ♭ src ♭ app ♭ heroes ♭ ↔ heroes.html ♭ 🗭 div ♭ 🗭 div
       <h2>My Heroes</h2>
      *ngFor = "let hero of heroes">
              <button type = "button"</pre>
               [class.selected] ="hero === selectedHero" (click)="onSelect(hero)">
              <span class = "badge">{{hero.id}} </span>
              <span class = "name">{{hero.name}} </span>
           </button>
           <div *ngIf = "selectedHero">
           <h2>{{selectedHero.name | uppercase}} Details</h2>
           <div>id: {{selectedHero.id}}</div>
 14
                   <label for="hero-name"> Hero Name:</label>
                   <input type="text" id="hero-name"</pre>
                   [(ngModel)] = "selectedHero.name" placeholder = "name">
       </div>
```

```
app.html
             TS app.ts
                           TS app-module.ts
my-angular-app ▷ src ▷ app ▷ heroes ▷ TS heroes.ts ▷ ...
  import { Hero } from '../hero';
      import { FormsModule } from '@angular/forms'
      import { HEROES } from '../mock-heroes';
      import {
      NgFor, NgIf, UpperCasePipe
       /**... */} from '@angular/common';
  9
 selector: 'app-heroes',
 11
       standalone: true,
 12
       templateUrl: './heroes.html',
 13
       styleUrl: './heroes.css',
 14
 15 E
       imports: [
 16
        NgFor, NgIf, FormsModule, UpperCasePipe
 17
       1
 18
      })
 heroes = HEROES;
 20
 21
       selectedHero?:Hero;
       onSelect(hero:Hero): void{
 22 ⊟
        this.selectedHero = hero;
 23
 24
        }
 25
```

```
my-angular-app > src > app > TS hero.ts > •• Hero

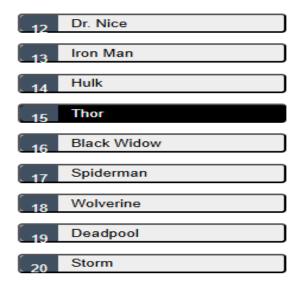
1 export interface Hero {
2 id: number,
3 name: string
4 }
```

```
app.html
               TS app.ts
                                TS app-module.ts
my-angular-app ♭ src ♭ app ♭ TS mock-heroes.ts ♭ 🝽 HEROES
       import {Hero} from './hero';
       export const HEROES: Hero[] = [
           {id:12 , name: 'Dr. Nice'},
           {id:13 , name: 'Iron Man'},
           {id:14 , name: 'Hulk'},
           {id:15 , name: 'Thor'},
           {id:16 , name: 'Black Widow'},
           {id:17 , name: 'Spiderman'},
           {id:18 , name: 'Wolverine'},
 10
           {id:19 , name: 'Deadpool'},
 11
           {id:20 , name: 'Storm'},
 12
 13
       1
```

OUTPUT:

my-angular-app

My Heroes



THOR Details

id: 15 Hero Name: Thor