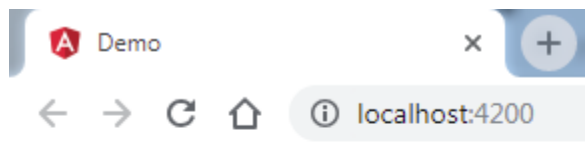


1. Create Angular application which creates two custom directive as CompSuccess and CompFailure. In case of success when move mouse cursor over data its color become green and in case of failure text color become red. When we remove mouse its color become black.

```
<> app.component.html x TS comp-failuer.directive.ts TS comp-success.directive.ts
src ▶ app ▶ <> app.component.html ▶ ...
1 <!--The content below is only a placeholder and can be replaced.-->
2 <div>
3   <h3> Assignment 14 Q1 ::</h3>
4 </div>
5 <div>
6   <h4 appCompSuccess > Use Success Directive </h4>
7   <h4 appCompFailuer > Use Failuer Directive </h4>
8 </div>
```

```
<> app.component.html TS comp-failuer.directive.ts x TS comp-success.directive.ts
src ▶ app ▶ TS comp-failuer.directive.ts ▶ ...
1 import { Directive, ElementRef, HostListener } from '@angular/core';
2
3 @Directive({
4   selector: '[appCompFailuer]'
5 })
6
7 export class CompFailuerDirective {
8
9   constructor(private ele:ElementRef) { }
10
11   @HostListener('mouseenter') onmouseenter()
12   {
13     this.ele.nativeElement.style.color = 'red';
14   }
15   @HostListener('mouseleave') onmouseleave()
16   {
17     this.ele.nativeElement.style.color = 'black';
18   }
19 }
```

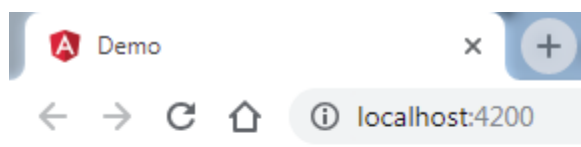
```
app.component.html TS comp-failuer.directive.ts TS comp-success.directive.ts x
src > app > TS comp-success.directive.ts > ...
1 import { Directive, ElementRef, HostListener } from '@angular/core';
2
3 @Directive({
4   selector: '[appCompSuccess]'
5 })
6
7 export class CompSuccessDirective {
8
9   constructor(private ele:ElementRef ) { }
10
11   @HostListener('mouseover') onMouseOver()
12   {
13     this.ele.nativeElement.style.color = 'green';
14   }
15   @HostListener('mouseout') onMouseOut()
16   {
17     this.ele.nativeElement.style.color = 'black';
18   }
19 }
```



**Assignment 14 Q1 ::**

**Use Success Directive**

**Use Failuer Directive**



**Assignment 14 Q1 ::**

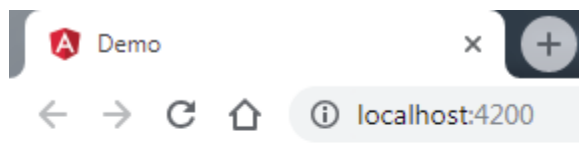
**Use Success Directive**

**Use Failuer Directive**

2. Create Angular application which creates one custom directive as CustomStyle. In this case set background color of text as yellow and text type should be bold. No need to handle mouse listener.

```
<> app.component.html x TS custom-style.directive.ts
src > app > <> app.component.html > ...
1  <!--The content below is only a placeholder and can be replaced.-->
2  <div>
3    <h3> Assignment 14 Q2 ::</h3>
4  </div>
5  <div>
6    <span appCustomStyle> Use Custom style here </span>
7  </div>
```

```
<> app.component.html TS custom-style.directive.ts x
src > app > TS custom-style.directive.ts > ...
1  import { Directive, ElementRef } from '@angular/core';
2
3  @Directive({
4    selector: '[appCustomStyle]'
5  })
6  export class CustomStyleDirective {
7
8    constructor(private ele: ElementRef) {
9
10     this.ele.nativeElement.style.background = 'yellow';
11     this.ele.nativeElement.style.fontWeight = 'bold';
12   }
13 }
```



**Assignment 14 Q2 ::**

**Use Custom style here**