⑧ custom-directives in Angular-6

colorDirective.ts

```
import { Directive,ElementRef} from "@angular/core";
@Directive({
  selector:'[pcolor]'
})
export class ColorDirective{
  constructor(el:ElementRef){
    el.nativeElement.style.backgroundColor = "yellow";
  }
}
```

in view show above selector as property

```
<span pcolor>Register Here</span>
```

If you want make like events then,

```
import { Directive, ElementRef, HostListener } from '@angular/core';
@Directive({
  selector:"[pcolor]"
})
export class ColorDirective{
  constructor(private el:ElementRef){}
  @HostListener('mouseenter') onMouseEnter(){
    this.highlight("yellow");
  }
  @HostListener('mouseleave') onMouseLeave(){
    this.highlight(null);
  }
  private highlight(color:string){
    this.el.nativeElement.style.backgroundColor = color;
  }
}
Textbox:
---------
import { ElementRef, HostListener } from '@angular/core';
import { Directive, Input } from '@angular/core';
@Directive({
 selector:"[disableInput]"
})
export class DisableInputDirective{
 constructor(private element:ElementRef){}
 @Input("disableInput")
 blockedWords:string;
 @HostListener("blur")
 m1(){
  var value = this.element.nativeElement.value;
  var blockWords = this.blockedWords.split(',');
  for(var i=0;i<blockWords.length;i++){
   if(value == blockWords[i])
   {
    alert("We dont allow this value");
    this.element.nativeElement.value="";
   }


  }
 }
}
//in html
<input type="text" name="username" #uname="ngModel" class="form-control  border-none"
placeholder="Username" [(ngModel)]="user.username" disableInput="admin,subadmin" >
```

fork- join

input

output

observables → 31st July

C1 -TS

i: 10

C1 - Temp → 14th Aug 2018

{{i}}

<c2 [j] = i > </c2>

(sendData) = "m2($event)"

@Input()

j: number;