**Custom Directory and Custom Pipe**

**Custom Pipe:**

**############################################################**

**Create a pipe**

**Ng g p custompipe**

**Use in html**

**<h2>Line Started {{ "This is ex string " | custompipe:"FirstArgu":"Default" }}</h2>**

**Customepipe.ts**

**@Pipe({**

**name: 'custompipe'**

**})**

**export class custompipePipe implements PipeTransform {**

**transform(value:any, ...args): any {**

**//getting two argument from interpolation where value is "This is ex string "**

**// and args[0]="FirstArgu" ,args[1]="Default" we can return data by condition**

**if(args[0]!='FirstArgu'){**

**console.log("FirstArgu",args[0]);**

**return value ;**

**}**

**if(args[1]=='Default'){**

**console.log("SecondArgu",args[1]);**

**return args[1];**

**}**

**}**

**}**

**############################################################**

Create a custom Directly

ng g d customDir

IN Html

<p appCustomDir >this is custom Directive</p>

**IN CustomDirDirective.ts**

**import { Directive, ElementRef } from '@angular/core';**

**@Directive({**

**selector: '[appCustomDir]'**

**})**

**export class CustomDirDirective {**

**constructor(private el:ElementRef) {**

**console.log("Console CustomDirDirective",el);**

**// to console js style in chrome ------- $0.style.color**

**// to check debug process in chrome ------- debugger;**

**el.nativeElement.style.color="red";**

**el.nativeElement.style.backgroundColor="black";**

**}**

**}**