# Angular:❤️Day2

Now →to create project (ng new filename)

→to Run project (ng serve -o)

"Must your director be the new folder created && will create .angular folder you can delete it if you want also can delete .vscode and any folder for testing(.spec)"

In tsconfig.json file→"strict" :true (can do it false if you want it loosely coupled)

In tsconfig.app.json file →convert all typescript files to javascript

```
"scripts": {
   "ng": "ng",
   "start": "ng serve",
   "build": "ng build",
   "watch": "ng build --watch --configuration development",
   "test": "ng test"
 },
```

In package.json file→contain all dependencies

We can run the project with another command but with diff port (npm start)"replace it by ng serve"

→if you want to open directly in your browser and with diff port do this

```
"scripts": {
   "ng": "ng",
   "start": "ng serve -o --port 5200",
   "build": "ng build",
   "watch": "ng build --watch --configuration development",
   "test": "ng test"
 },
```

Some commands for (build , watch or test your project)

- npm run build
- npm run watch
- npm run test

(Package.json+package-lock.json+node modules مرتبطين ببعض)

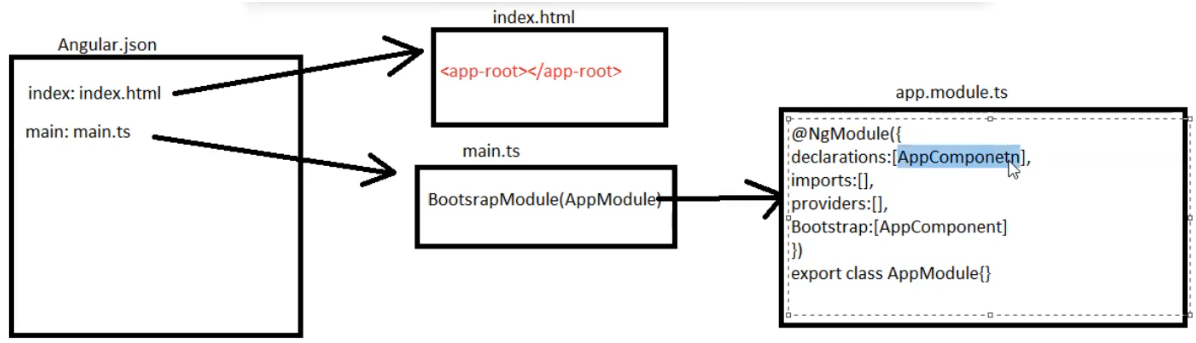
In angular.json→Contains

- Version, Projectroot called"projects"
  - ★ Contain sourceRoot->src
  - ★ Architect
    - ➢ build(index->start point in src folder),serve,test,extract-i18n

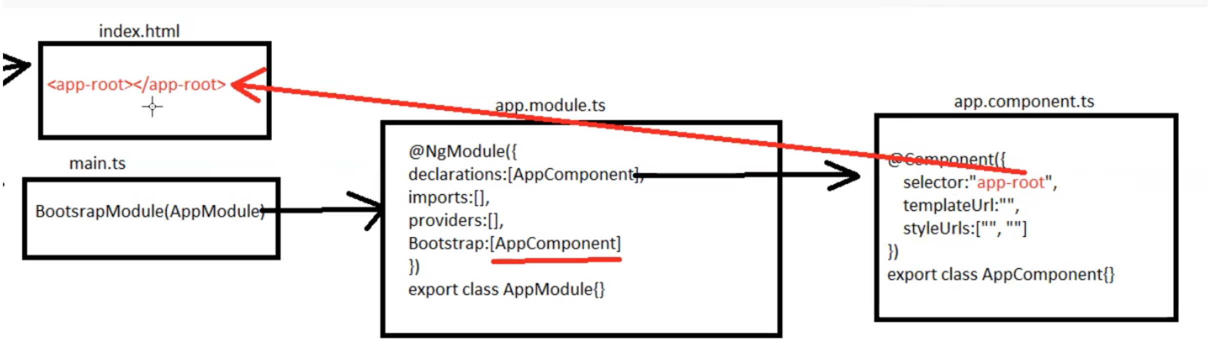
Start point → src folder

->index.html"contain new tag for app-root"

-> main.ts"import app module"



```
@NgModule({
declarations:[AppComponetn],
imports:[],
providers:[],
Bootstrap:[AppComponent]
})
export class AppModule{}
```



```
@NgModule({
declarations:[AppComponent],
imports:[],
providers:[],
Bootstrap:[AppComponent]
})
export class AppModule{}
```

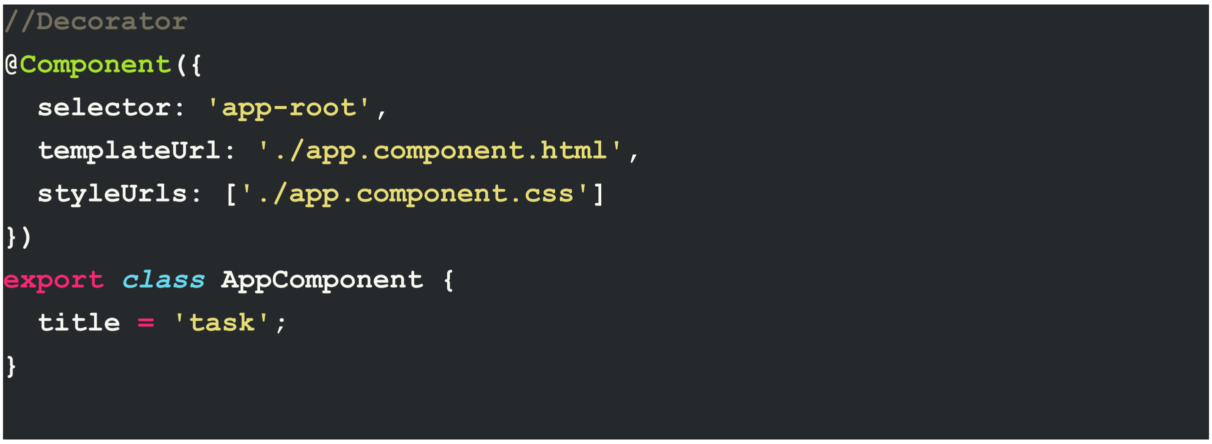
```
@Component({
  selector:"app-root",
  templateUrl:"",
  styleUrls:["", ""]
})
export class AppComponent{}
```

Any decorator import from @angular/core

In src->app -> app.modules.ts contains decorator ”contain metadata about the main module in our project”

```
@NgModule({
  declarations: [
    AppComponent
    /*
    //here we declare any
    1-component
    2-pipes
    3-directives
    */
  ],
  imports: [
    BrowserModule,
    /*
    here we put
    4-external module
     [httpclient module,
       router module,
       forms modules-....]
    */
    AppRoutingModule
  ],
  providers: [
    /**
     * 5-put on it services
     */
  ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

In src->app -> app.component.ts contains another decorator



```
//Decorator
@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'task';
}
```

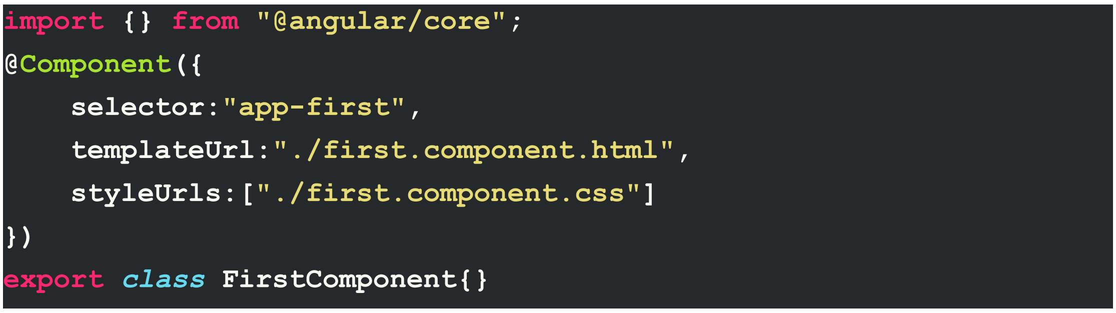
To create new component

-> in src -> app ->create folder components contain componentone contains ( html , css, ts)

اللى يربط صفحة ال html و ال css و ال logic هيكون ال ts

Steps:

1. Create 3 files (html,css,ts)
2. In ts file create decorator contain (selector,template,style)



```
import {} from "@angular/core";
@Component({
    selector:"app-first",
    templateUrl:"./first.component.html",
    styleUrls:["./first.component.css"]
})
export class FirstComponent{}
```

3. Do any tag in html file
4. Add component you create it in app.modules in declaration part
5. Add component tag in app.component.html

بدل كل القرف دا فى command يقدر يعملك ال Component

Step1:

Right click on component folder and open integrated terminal

Step2:

ng g c componentname

علشان نعمل global style حطة فى ال styles.css

Binding → get data from typescript and display it in html

```
<!--Binding Ways one way binding-->
<!--Way 1-->
<p>{{name}}</p>

<!--Way 2-->
<input type="text" value="{{name}}" />

<!--Way 3-->
<input type="text" [value]="name"/>
```

```
<!--Event Binding-->

<input type="button" value="Change" (click)="change()">
```

## Twoway Binding

1. Our way

```
  typing(e:any)
    {
        this.word=e.target.value;

    }
```

```
<input placeholder="Enter any word" type="text"
(input)="typing($event)">
<p>{{word}}</p>
```

2. Angular Way (without events and target)

```
  imports: [
    BrowserModule,
    FormsModule,//to 2way work
```

```
<input placeholder="Enter any word" type="text"
[(ngModel)="lname"]>
<p>{{lname}}</p>
```