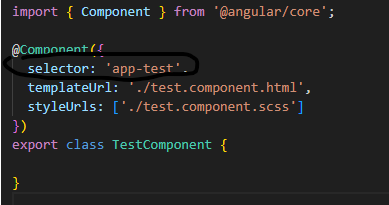
* Install nodejs
* Install vscode
* Go to vs code terminal install angular
* Npm install -g @Angular/cli
* If getting below error

# **[Angular - ng.ps1 cannot be loaded because running scripts is disabled on this system](https://stackoverflow.com/questions/68178101/angular-ng-ps1-cannot-be-loaded-because-running-scripts-is-disabled-on-this-sy)**

Then run Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy Unrestricted

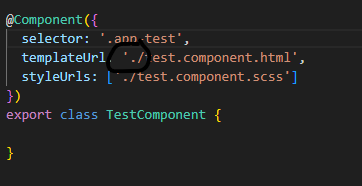
* In terminal
* Ng create -g roj name
* Npm start
* ng g c test - for creating new component
* Anytime you create a new component ur application should be aware of it. So to do that add it in appModule.ts file inside declaration array []
* For html root component is app-component.html there u will have to add ur new component html if you want to include ur component in the parent ui

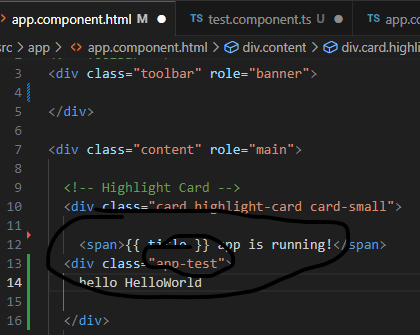
**Changing selector**

* Ways of Using selector of the component
* 
* 1.using as

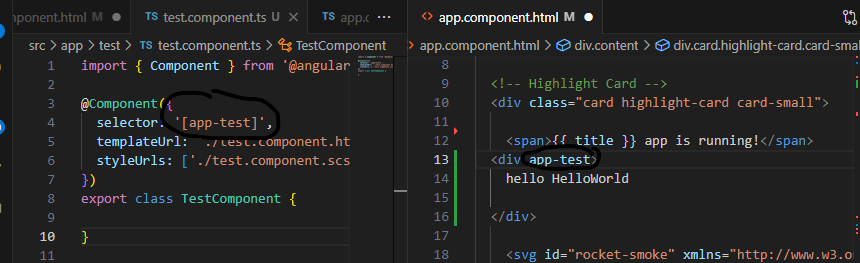
<app-test></app-test> in app component.html

* 2. usng as class add dot in frnt of selector name and then use it in appcomponent.html as aclass in a div tag



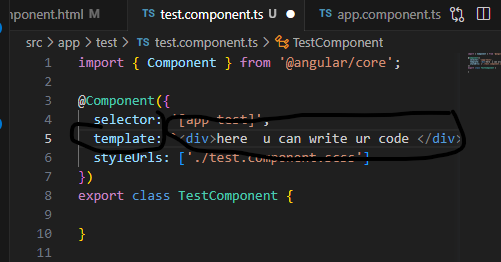


3rd way is use it as a square bracket in the selector and in html of app component use it as a attribute in the div tag



**Changing template**

Templateurl points to file which contains html and in any component its possible to specify the template inline which is in the same typescript file and for that we use **template** insteand of templateUrl



**Interpolation**

: {{name }} : -> it is called interpolation

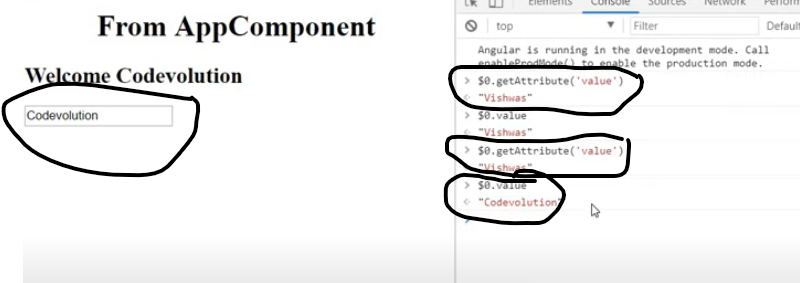


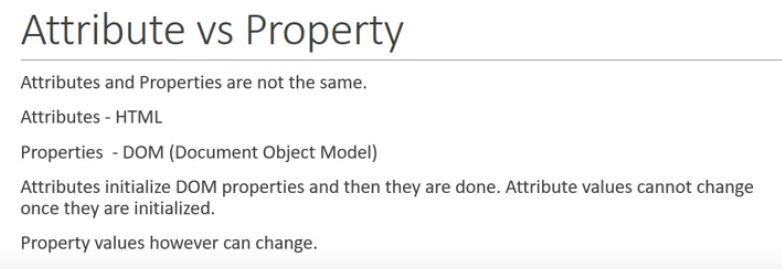
Assignment is not possible like name=mohit in interpolation

With interpolation we can bind data from class to the template. We can call method also inside {{}} bracket

**Property binding**

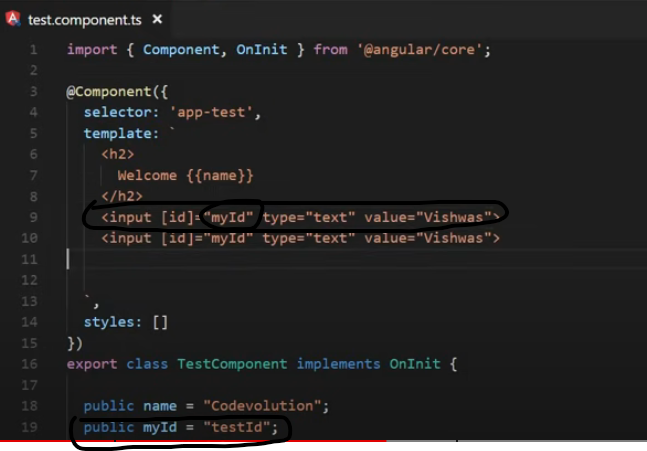
**Attribute vs property**



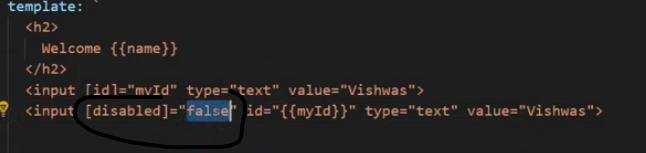


Atrribute specify the initial value and the dom value prp is current value so val attrib is always same while val is changed

**Property binding**



We use property binding because there is limitation for the interpolation, it can only work with string values. So if we need boolean value we can’t use interpolation

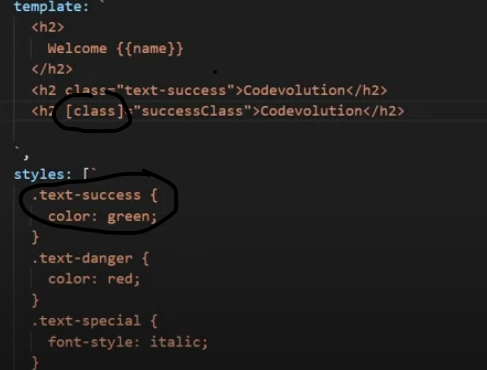


Alernate syntax for property binding is

Bind-propertyname



**Binding classes to the html elements**

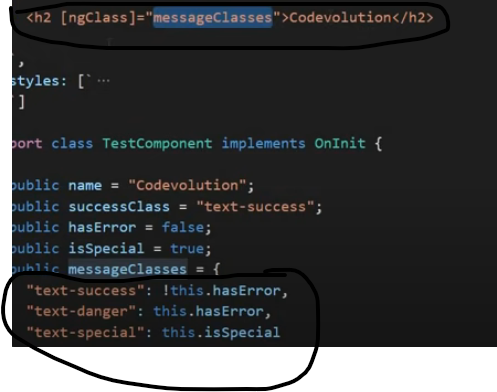




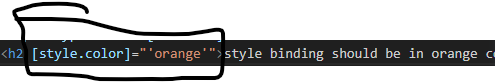
Has error is in the class as bolean variable

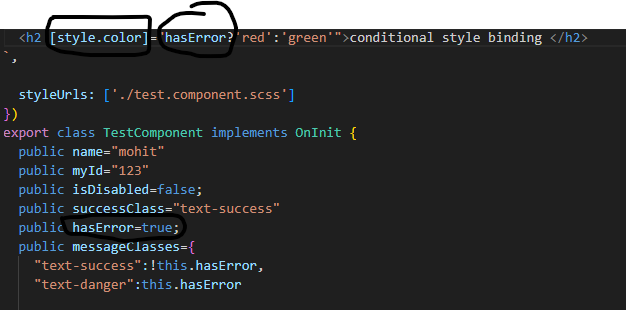
These things works fine when you conditionally apply a signle class but when you want to apply multiple class then angular provide the ng class directive

**Ngclass directive is a custom html attribute**

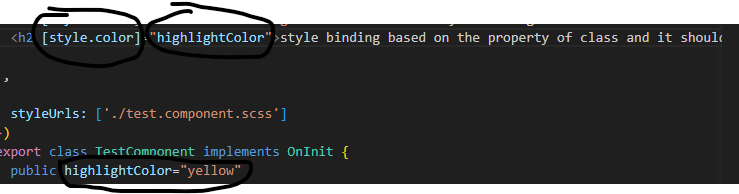


**StyleBinding**

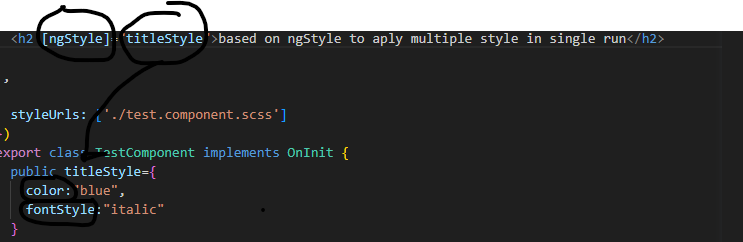
* In angular style binding used to apply inline style to html elements. And it is similar to class binding
* We use style binding as 
* Conditional style binding as below



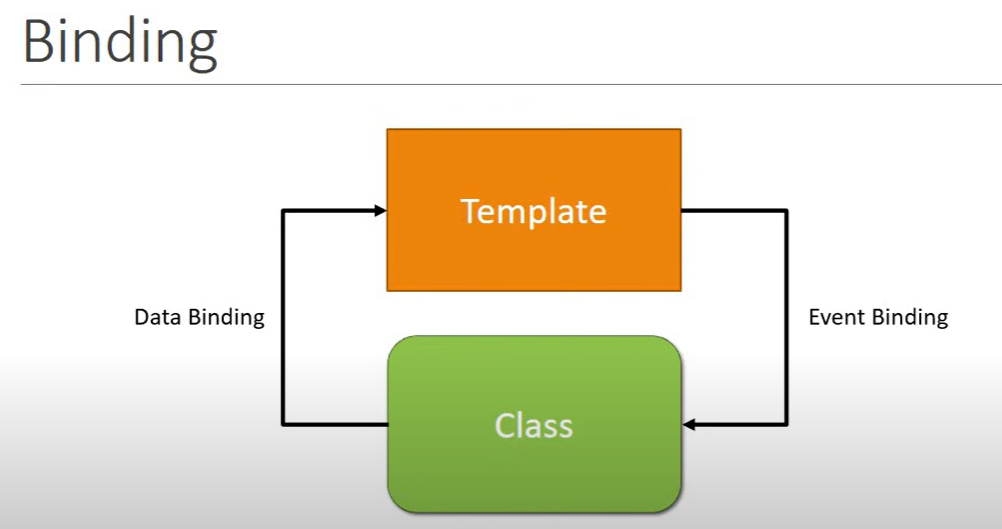
* We can also assign component class property during binding like below

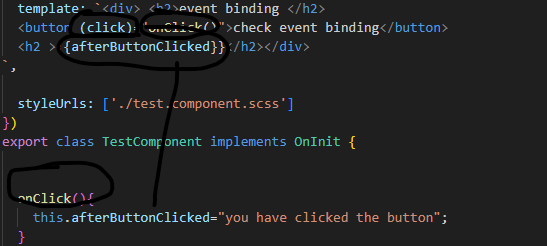
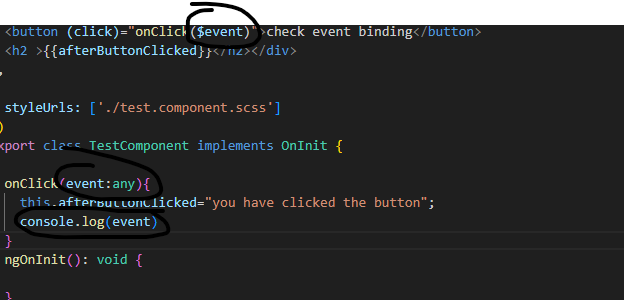
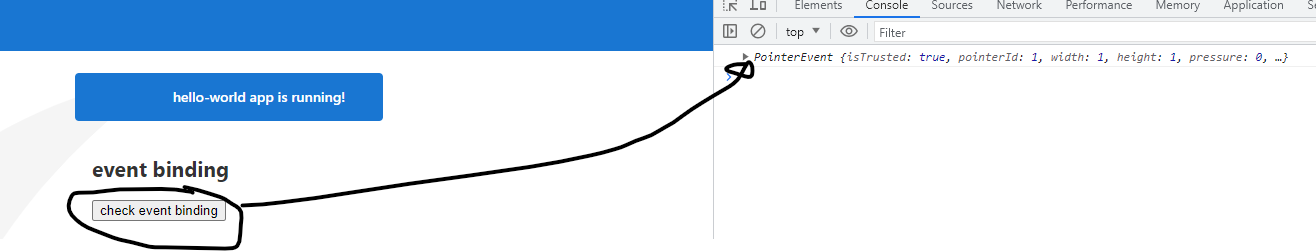


* To apply multiple style we make use of ngStyle directive



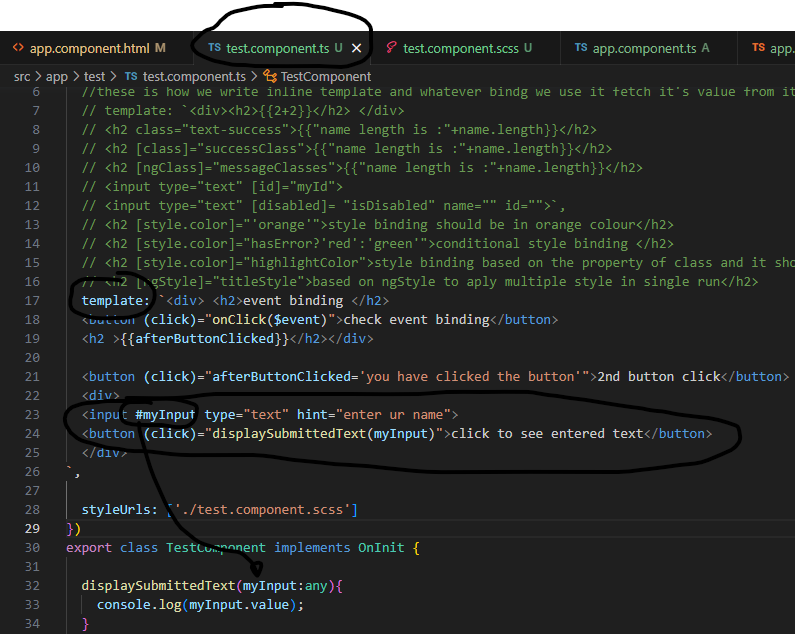
**Binding**



* **Data Binding** : data flow is from component class to the component template
* **Event binding** : is used when we need data flow from component template to component class. Basically it is used to capure event like what sould happen on button click
* 
* Sometime we need information about the event itself like info about click event for that we simply send a parameter in the event handler that parameter is $event
* $ event give you all the info about the dom event
* 
* 
* 

**Template ref variable**

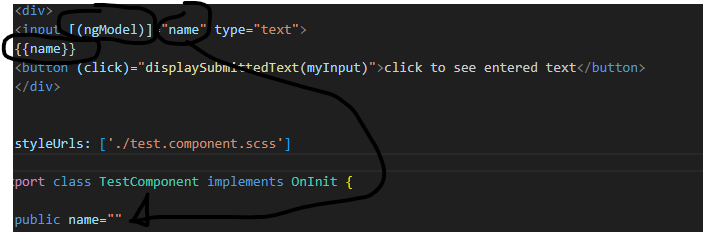
* When we need Data to flow from view to class to perform some operation then for this angular provide template reference variable.
* So temp ref variable can be used to refer a HTML element and all of its DOM property and we **create it using # followed by variable name** in the HTML field.



* **Two way binding** : it allow us to update a property and at the same time display val of that property and for it we use **ng model** directive. So it’s like a view and a model should always be in sync.
* []-> we have square bracket for property binding which is for data flow from class to template
* ()-> we have parenthesis for event binding which for data flow from template to class
* So in ng model there would be 2 way data flow hence we write 2 way binding.

[(ngModel)]=”propertyName” -> syntax is like banana in a box

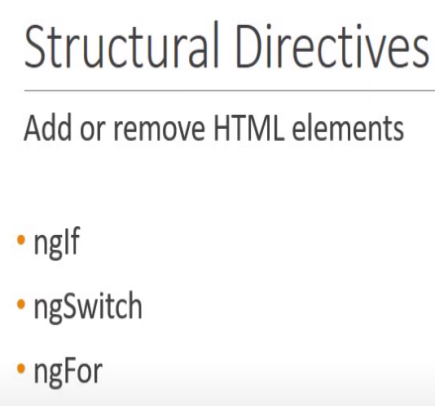
* By default angular is not aware about the ngmodel directive so to enable this we have to import forms modul. So we can import it in app.module.ts file



This will ensure template and property value is always n sync and consistent

**Structural directives**

* It let you add or remove html element from DOM

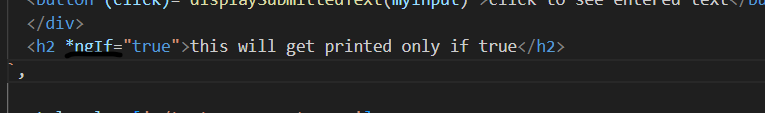


* ngIf and ngSwitch is used to conditionally render the element
* ngFor is used to render a list of html element

11 videos completed

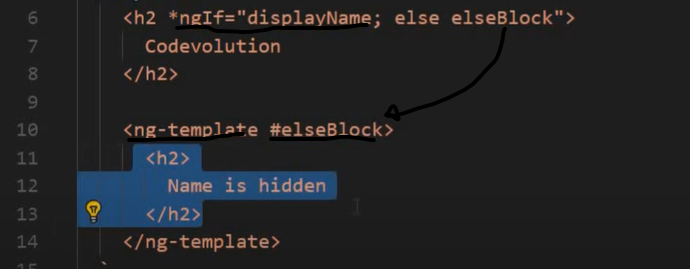
**Ng if directive**

**-> \*ngIf=true or false condition**



If block can also have a else block

Check if ng if condition true if false then chk if there is else statement for that we use **ng template container** to make reference to if block like below



**Another syntax for ng if**

First create if and else block an then create ng-template tag so flow will be like if true then then block else else block

