

## Data binding

Data binding provide bridge between component to template or vice-versa.

Using data binding we can share the data between component to template or template to component.

2 types of data binding

1. One way data binding : uni directional

- a. **String interpolation** : it is a type of one way data binding  
Component to View or template  
.ts ----> .html

Syntax

{{}}

Example

{{variableName}}

{{5+7}}

{{functionName()}}

- b. **Property binding** : it is a type of one way data binding

If we want to use property binding we need to use tag or dom elements.

Component to View or Template

.ts → .html

Syntax

[]

<input type="text"/>

<input type="text" value="Raj">

<input type="text" value="fname"> then value itself is fname.

<input type="text" [value]="fname"> then angular compiler check fname value  
Present in component and that fname value  
Will display in text filed.

<p [innerText]="fname"></p>

in html

text field will create and inside that textfield  
Raj default value present.

{{}}

Vs

[]

{{fname}}

<div [innerText]="fname"></div>

ng g c property-binding

- c. Event binding : flow the application template to component.  
.html --> .ts

Syntax

()

Angular use same event provided by JavaScript. Only different in angular on pre-fix removed and event name wrap with ().

JS Event

Angular event

onClick  
onDbClick  
onMouseOver  
onSubmit

(click)  
(dblclick)  
(mouseover)  
(ngSubmit)

etc

etc

ng g c event-binding

template reference

if we want to pass any tag value to component we can take the help of template reference.

<input type="text" #nameRef />

nameRef is a template reference of input text field.

Two way data binding using

Event binding and string interpolation or property binding

## 2. Two way data binding : bi directional

Create new project with name as angular-data-binding

ng new angular-data-binding

routing → no

styling → css enter

cd angular-data-binding

ng serve -o

then open the project in vs code

please open app.component.html : remove all code and write below code.

```
<div>
  <h2>Angular data binding concept</h2>
</div>
```

ng g c string-interpolation

### Two – way data binding using ngModel

Angular provided two way data binding : if we do any changes in component it will update in template and vice-versa.

[(ngModel)]

ngModel is a pre defined attribute we can use with [(ngModel)] to achieve two way data binding.

ngModel is a pre defined attribute part of FormsModule.

In app.module.ts file we need to import FormsModule in import section.

ng g c two-way-data-binding

### Angular directives

Angular directive help us to add extra behaviour or functionality to dom or html code.

3 types of directive

1. Component directive:

Component is a type of directive which help to create user defined tags with help of selector attribute.

```
@Component({
  Selector:"my-tag",
  templateUrl:"./my-page.html"
})
class MyComponent{

}
```

2. Structure directive

Using structure directive we can add or remove html code in existing dom.

ng-if  
ng-for

in angular we can use if and for inside html page.

3. Attribute directive : attribute directive mainly use to apply styling.  
ngStyle and ngClass

Please create new project

ng new angular-directive

move inside a project

cd angular-directive

then create two components

ng g c structure-directive

ng g c attribute-directive

using

ng serve -o                      start the project

and open the project in vs code.

Then open app.component.html page and remove pre-defined contents.

```

```