



CJC

Complete Java Classes

by **Kunal Sir**

Angular Databinding

❖ **What is databinding ?**

Databinding is one of the key feature of an Angular, that allows us define communication between component and DOM (Document Object Model)(view) , which helps us to implement (develop) dynamic web applications.

It means after using databinding , we don't need to worried about how data will be injected form component to HTML and HTML to component.

In databinding , data stays in sync between the component and the view. Whenever the user updates the data in the view, Angular updates the component. When the component gets new data, the Angular updates the view.

❖ **Below are the four types of data binding :**

A. String interpolation : - (TS ----> HTML)

In Angular , String interpolation is use to display data (values of a data-members) in to the view (html).

The data flow in String interpolation is one way , And flow direction is component to view of a component.

String interpolation is denoted by expression `{{ }}` .



by Kunal Sir

Angular Databinding

❖ Example on string interpolation :-

1. Write a program to declare a property (data member) in AppComponent .

```
src > app > app.component.ts > AppComponent
1  import { Component } from '@angular/core';
2  import { User } from './model/user';
3
4  @Component({
5    selector: 'app-root',
6    templateUrl: './app.component.html',
7    styleUrls: ['./app.component.css']
8  })
9  export class AppComponent {
10
11    name:string='Complete Java classes..!'
12
13  }
14
```

2. Print the property in same component's HTML

NOTE:- Using data binding we can access component properties only within same component's HTML.

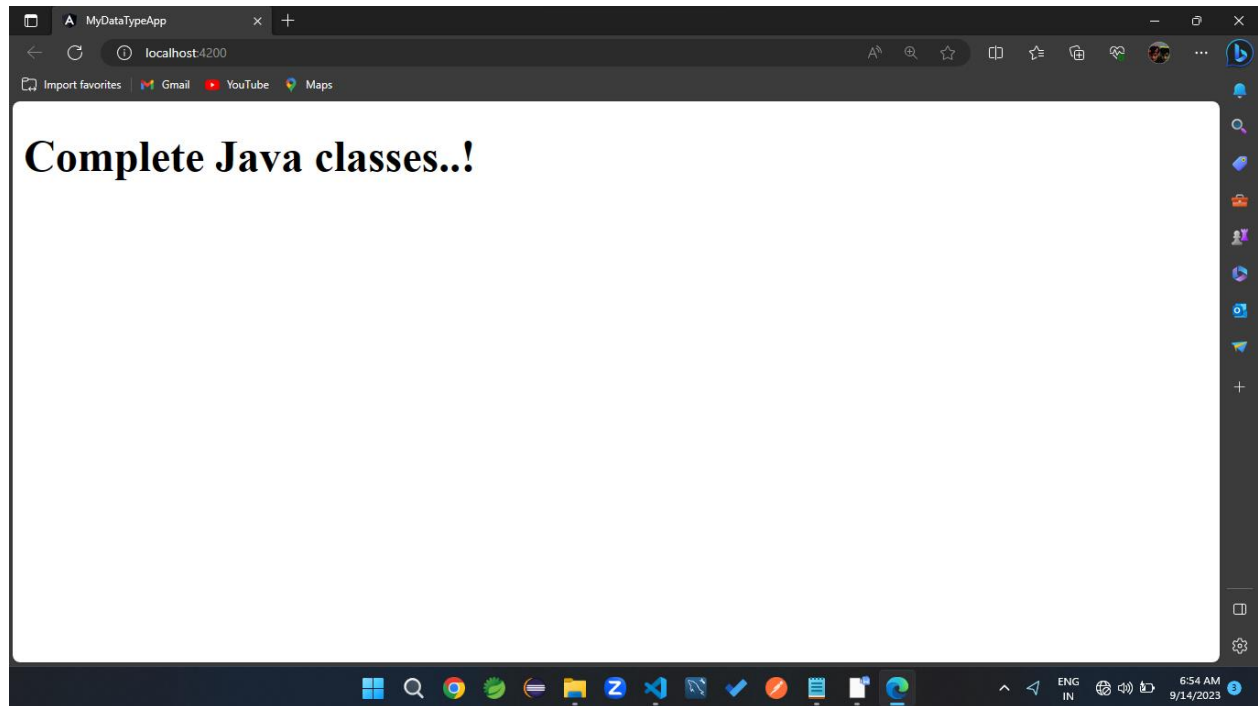
```
src > app > app.component.html > h1
Go to component
1
2  <h1>{{name}}</h1>
```



by **Kunal Sir**

Angular Databinding

Output:-



B. Property Binding : - (TS ---> HTML)

The property binding allows us to bind value of component data-member to the property (attribute) of HTML elements.

The data flow in property binding is one way , And the flow direction is again component to view (html).

Property binding is denoted by expression [] .

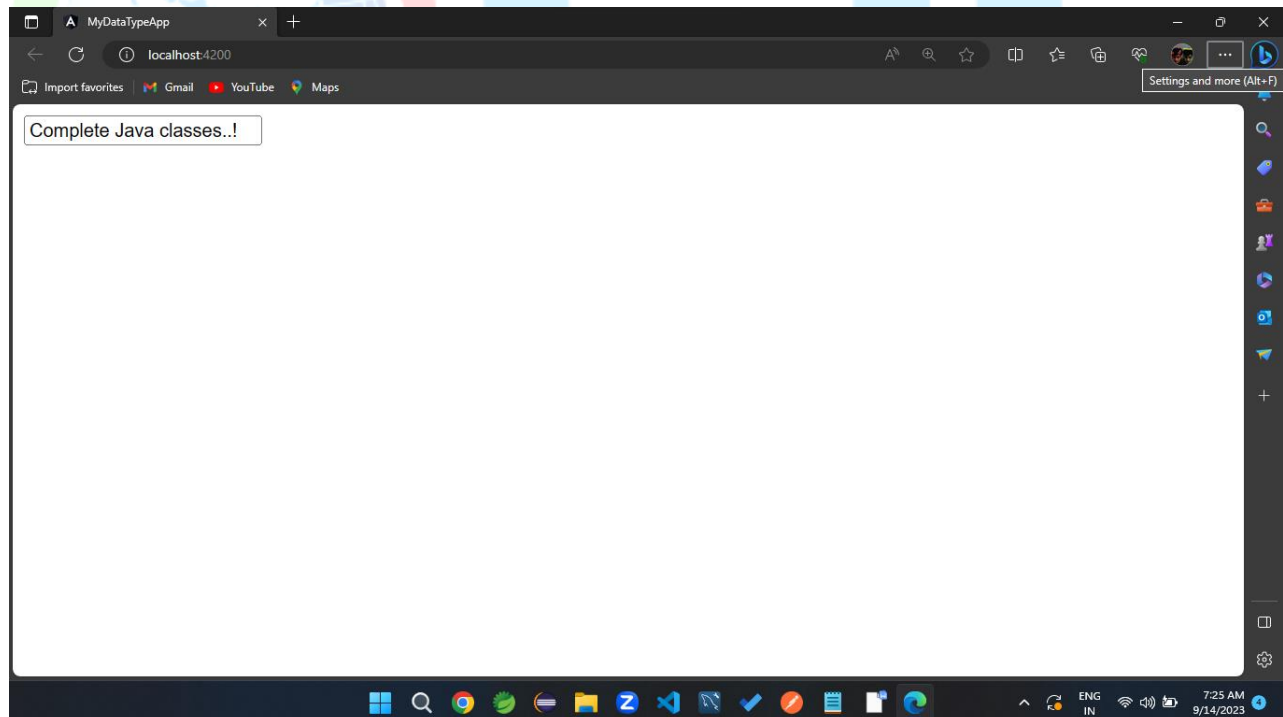
Angular Databinding

❖ Example of Property binding :-

1. Bind the existing data member 'name' to the value property of input element.

```
app.component.ts M  app.component.html M X
src > app > app.component.html > input
Go to component
1
2 <!-- <h1>{{name}}</h1> -->
3 <input type="text" [value]="name">
```

Output:-



Angular Databinding

C. Event Binding : - (HTML ---> TS)

Whenever end users (clients) interact to the view of our application, They perform some activities like clicking the button hovering on elements , Entering value in input boxes

This user activities may raise some events , and to handle this events raised by user activities we use event binding.

The data flow in event binding is one way , And the flow direction is view (html) to component.

Event binding is denoted by expression () .

Some angular events are click , mouseover, mouseout ,keypress.

❖ Example on event binding :-

1. Define a function 'onClick()' in component .

```
app.component.ts M x  app.component.html M  app.component.spec.ts M
src > app > app.component.ts > AppComponent > onClickButton
1  import { Component } from '@angular/core';
2  import { User } from './model/user';
3
4  @Component({
5    selector: 'app-root',
6    templateUrl: './app.component.html',
7    styleUrls: ['./app.component.css']
8  })
9  export class AppComponent {
10
11    onClickButton()
12    {
13      alert("Welcome User..!");
14    }
15  }
16
```

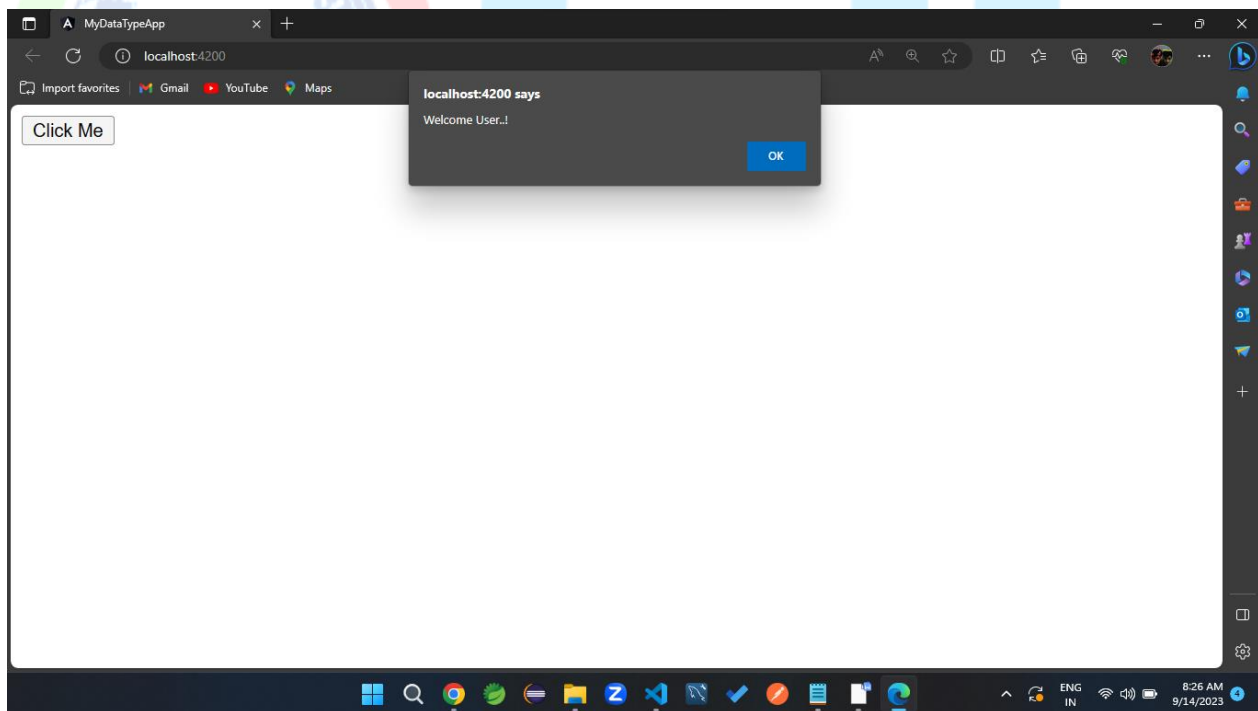
by Kunal Sir

Angular Databinding

2. Bind the function in button element of the html.

```
app.component.ts M  app.component.html M  app.component.spec.ts M
src > app > app.component.html > ...
  Go to component
  1
  2 <button>Click Me</button>
  3
```

Output:-





by Kunal Sir

Angular Databinding

D. Two-Way-Data Binding :- (TS <---> HTML)

Two way data binding is one of the most powerful data binding , It binds the component data member to the input element of a view and , whenever end user changes that value it updates the value of data member in component.

The data flow in two-way-data binding is two way , And the flow direction is component to view & view to component

Event binding is denoted by expression [(ngModel)] .

[]->property binding => ts-->html

()->Event binding => html--ts

ngModel= angular model (to store data inside)

Note :- Two user two way data binding we have to import FormsModule in our Module , Where our Component is declared.

❖ Example on two-way data binding

1. Import FormsModule in your Module file, Where your component is declared.

```
app.component.ts M  app.component.css  app.module.ts M x  app.component.html M
src > app > app.module.ts > AppModule
6 | import { FormsModule } from '@angular/forms';
7
8 @NgModule({
9   declarations: [
10     AppComponent
11   ],
12   imports: [
13     BrowserModule,
14     AppRoutingModule,
15     FormsModule
16   ],
17   providers: [],
18   bootstrap: [AppComponent]
19 })
20 export class AppModule { }
21
```



by Kunal Sir

Angular Databinding

1. Declare a data member in ts file.

```
app.component.ts M x app.component.css app.module.ts M app.component.html M
src > app > app.component.ts > AppComponent
1  import { Component } from '@angular/core';
2
3  @Component({
4    selector: 'app-root',
5    templateUrl: './app.component.html',
6    styleUrls: ['./app.component.css']
7  })
8  export class AppComponent {
9
10     curentTopict:string='Data binding'
11
12
13  }
14
```

2. Using string interpolation print value of curentTopic in html . also bind the it in one input element.

```
app.component.ts M app.component.css app.module.ts M app.component.html M x
src > app > app.component.html > input
Go to component
1
2  {{curentTopict}}<br>
3  <input type="text" [(ngModel)]="curentTopict">
```