Data Bindings

Types :

1. Source to view (.ts to .html)
2. View to Source (.html to .ts)
3. Source to view to source (Two way communication)

Data bindings can be done in two ways

1. Using interpolation Binding {{ }}
2. Property Binding [ ]

Let’s prepare binding component from node js and add the following command

* ng g c binding //binding is the name of component

goto **binding.component.ts** file

import { Component, OnInit } from '@angular/core';

@Component({

  selector: 'app-binding',

  templateUrl: './binding.component.html',

  styleUrls: ['./binding.component.css']

})

export class BindingComponent implements OnInit {

  constructor() { }

  ngOnInit(): void {

  }

  name:string="Nikunj Soni";

  url:string="https://www.google.com"

}

goto **binding.component.html**

<h1>Data Binding</h1>

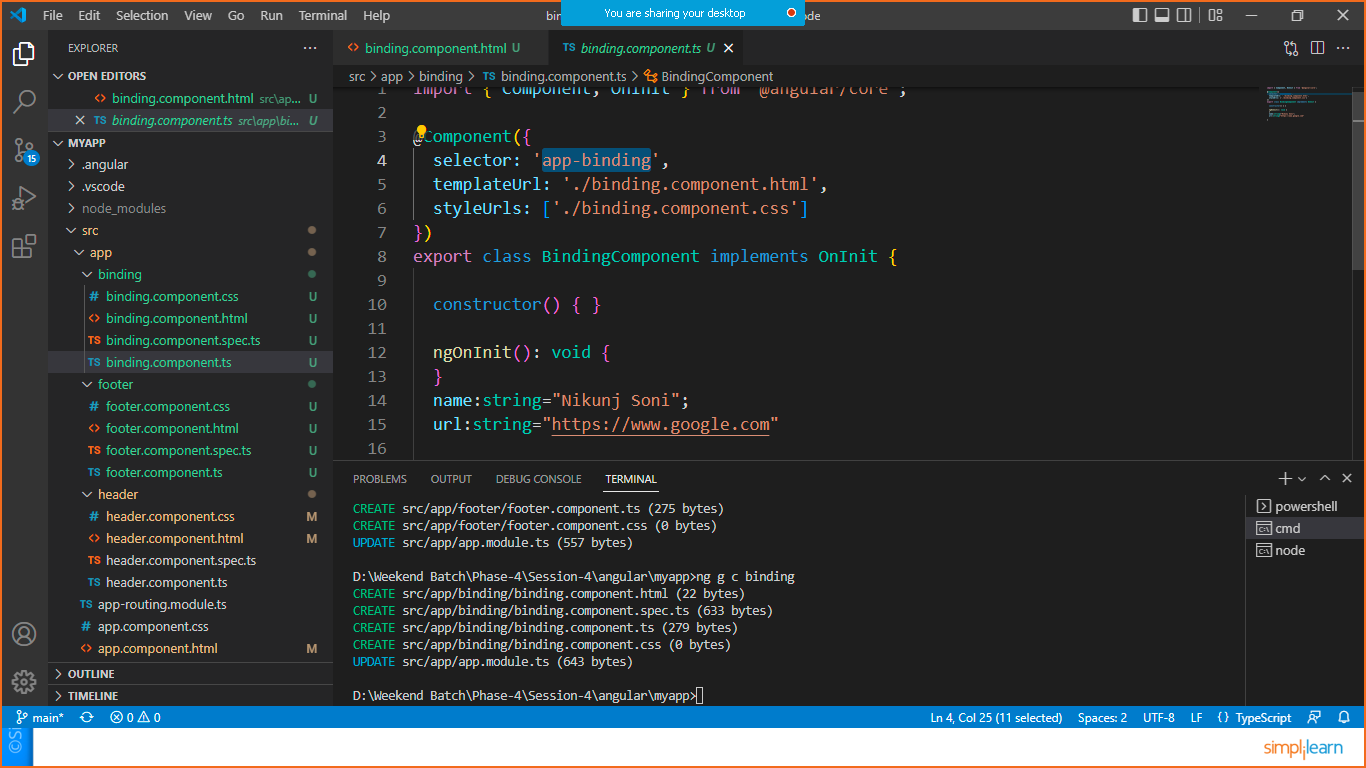
<!--use interpolation-->

<h2>Let's bind the data from source to view using Interpolation</h2>

<hr>

<p>{{name}}</p>

Now add the selector tag of binding.component.ts file to **app.component.html file**



**In app.component.html**

<h1>Data Binding</h1>

<!--use interpolation-->

<h2>Let's bind the data from source to view using Interpolation</h2>

<hr>

<p>{{name}}</p>

<hr>

<h2>Let's bind the data from source to view using  property binding, bind the value with attribute</h2>

<hr>

<a [href]="url">Connect to Google</a>

Save and check the output

**CLASS BINDING**

Goto your component.ts file

Eg: **binding.component.ts**

import { Component, OnInit } from '@angular/core';

@Component({

  selector: 'app-binding',

  templateUrl: './binding.component.html',

  styleUrls: ['./binding.component.css']

})

export class BindingComponent implements OnInit {

  constructor() { }

  ngOnInit(): void {

  }

  name:string="Nikunj Soni";

  url:string="https://www.google.com"

  products=[{name:"pencile",price:"8.3",available:"26-06-2022",rating:4.8},

            {name:"pen",price:"9.0",available:"30-06-2022",rating:5.0},

            {name:"Eraser",price:"5.5",available:"01-07-2022",rating:4.5},

            {name:"Scale",price:"10",available:"27-06-2022",rating:4.8},

            {name:"Strencil",price:"12.25",available:"26-06-2022",rating:4.9},

            ]

}

Goto > **binding.compoent.html** to print array using table

<h1>Print JSON Array</h1>

<table class="table table-striped table-bordered">

    <thead>

        <tr>

            <th>Name</th><th>Price</th><th>Availability</th><th>Ratings</th>

        </tr>

    </thead>

    <tbody>

        <!--NgDirectives  Eg:\*ngFor,\*ngIf,\*ngSwitch-->

        <tr \*ngFor="let p of products">

            <td>{{p.name}}</td>

            <td>{{p.price}}</td>

            <td>{{p.available}}</td>

            <td>{{p.rating}}</td>

        </tr>

    </tbody>

</table>

Now goto index.html file and add bootstarp link and script file

Like below

<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <title>Myapp</title>

  <base href="/">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" type="image/x-icon" href="favicon.ico">

  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css" integrity="sha384-zCbKRCUGaJDkqS1kPbPd7TveP5iyJE0EjAuZQTgFLD2ylzuqKfdKlfG/eSrtxUkn" crossorigin="anonymous">

</head>

<body>

  <app-root></app-root>

  <script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-fQybjgWLrvvRgtW6bFlB7jaZrFsaBXjsOMm/tB9LTS58ONXgqbR9W8oWht/amnpF" crossorigin="anonymous"></script>

</body>

</html>

Save it and check the output

