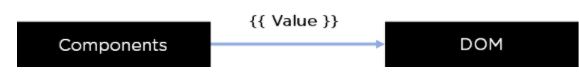
Contents

* Data Binding
* Basic Routing

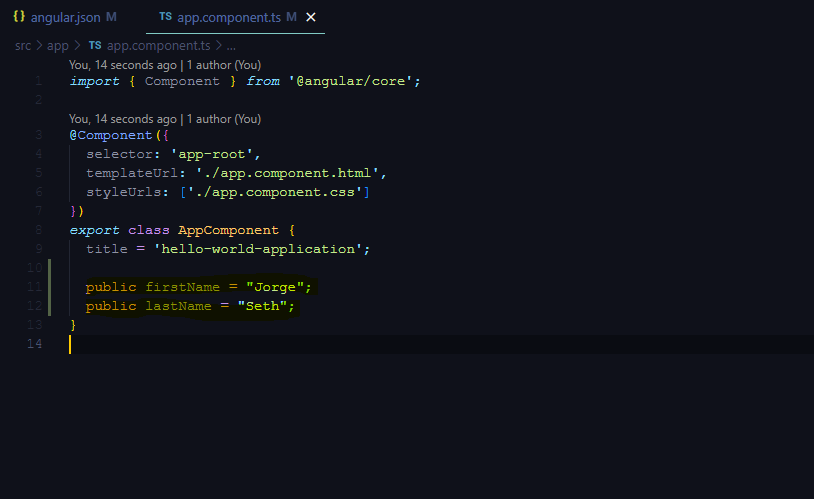


**Data Binding**

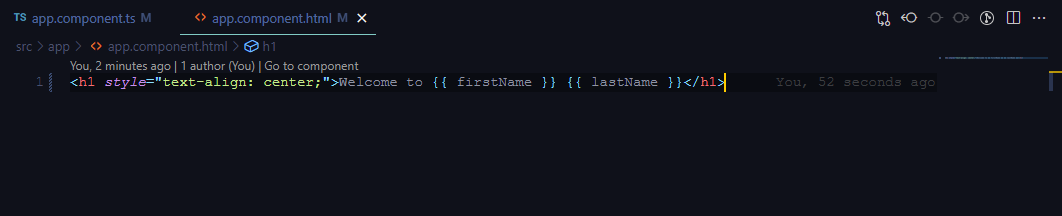
* **What is angular data binding?**
  1. Data binding allows Internet users to manipulate web page elements with the help of a web browser.
  2. It includes dynamic HTML.
  3. If the user wants to display dynamic data on a web page for that we use it to bind the data to our HTML Elements.
* **Type of data binding?**
  1. **Interpolation Binding**
     + Interpolation is a procedure that allows the user to bind a value to the user interface element.
     + Interpolation binds the data one way, which means that data moves in one direction from the components to HTML elements.
     + It uses the "{{}}" syntax for data binding.



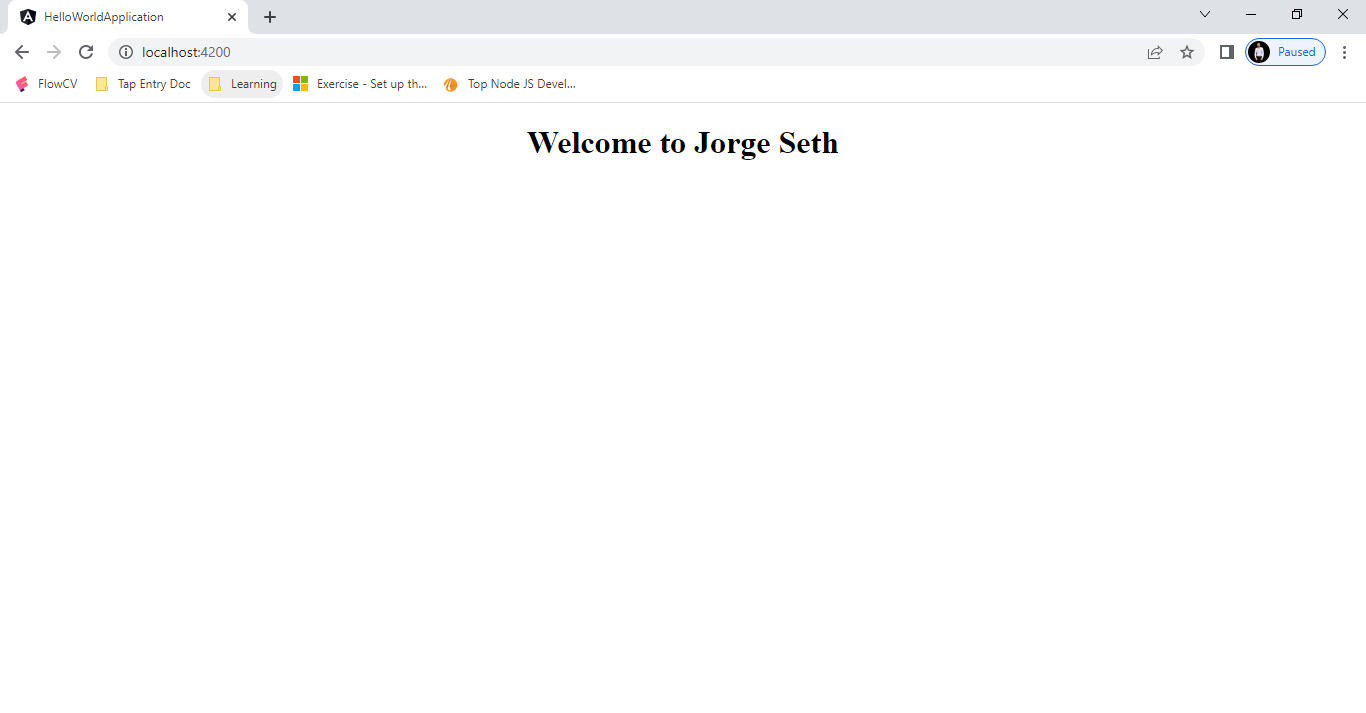
* + - We’ve added the code for the same below.



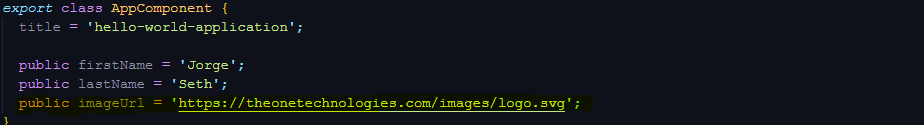
* + - In the above picture, **app.component.ts** file, we’ve created two properties called **firstName** and **lastName.**
    - To interpolate and bind them in the HTML file, type the following code



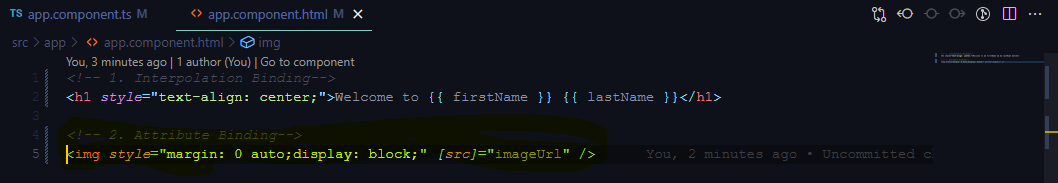
* + - **The output will look like this:**



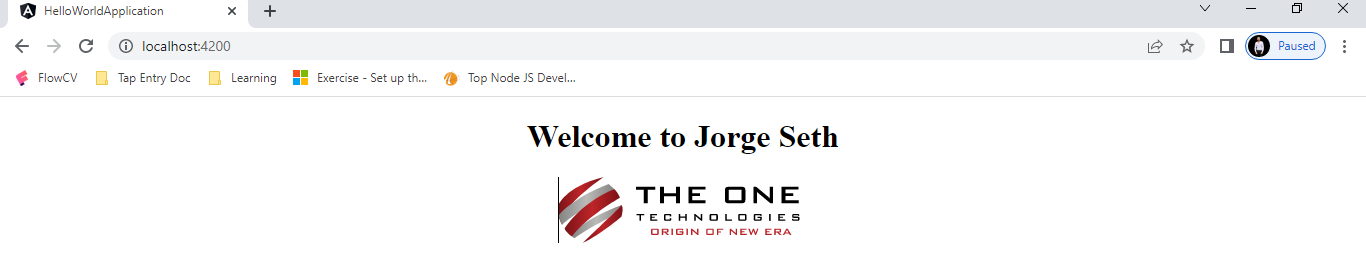
* 1. **Property Binding**
     + Property binding is a one-way data binding mechanism that allows you to set the properties for HTML elements.
     + It involves updating a property value in the component and binding the value to an HTML element in the same view.
     + It uses the "[]" syntax for data binding.
     + In the app.component.ts file, I’ve created another property called image and provided the URL of an image.



* + - In the app.component.html



* + - The output now looks like this.



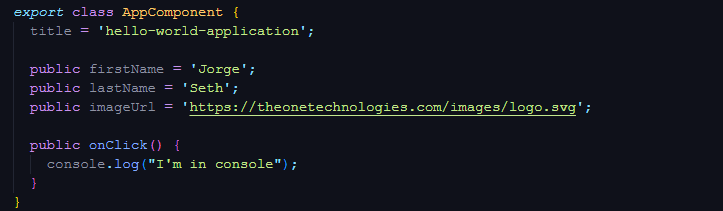
1. **Event Binding**
   * Event binding type is when information flows from the view to the component when an event is triggered.
   * The event could be a mouse **click** or **keypress**.
   * The view sends the data to update the component.

view-event

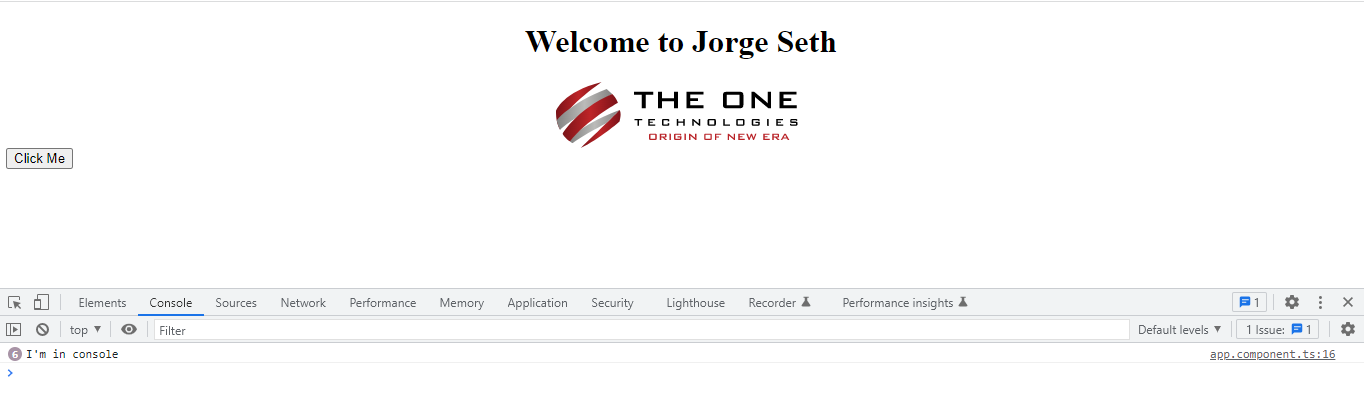
* + We have created a click me button that displays an **“I’m console log!”** message when clicked on.



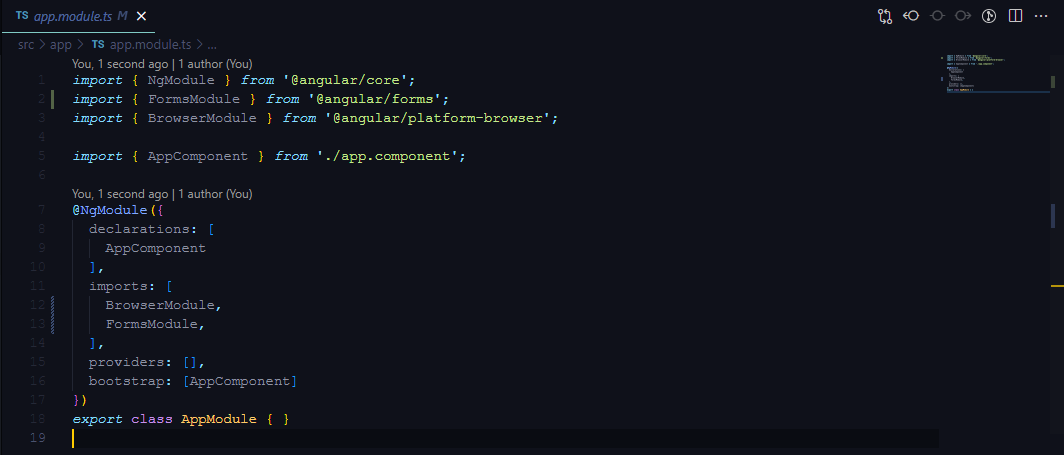
* + To display the message on the console, we’ve created a function called onClick() in the app.component.ts.



* + Every time the user clicks on the button, the message is displayed in the console.



1. **Two way-data Binding**
   * Two-way binding is a mechanism where data flows from the component to the view and back.
   * This binding ensures that the component and view are always in sync.
   * Any changes made on either end are immediately reflected on both.
   * The general syntax to denote two-way data binding is a combination of Square brackets and parentheses "[()]".
   * Before using this binding, we have to add the **FormModule** in our **app.module.ts** cause of that **ngModel** is not a part of angular’s code library, it is defined in the forms module library.

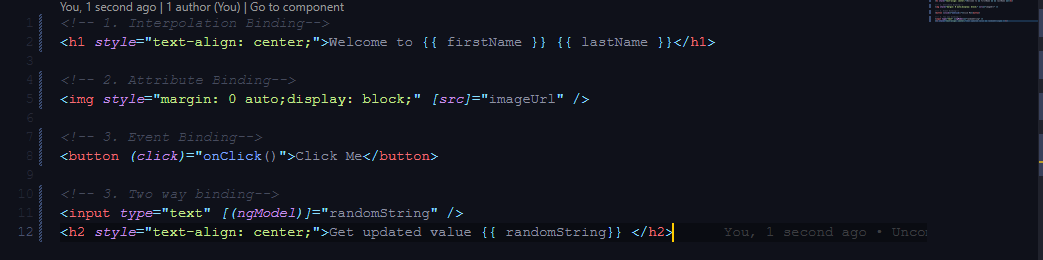


2-way

* + We’ve created a property with an empty string and an input box for the user to type.
  + Whatever the user provides is displayed on the screen with the help of the property.
  + In .ts file



* In the .html file we’ve created an input field.





1. **Class binding**
   * **Class Binding** is used to bind the data from the component to the HTML class property. The syntax is as follows.

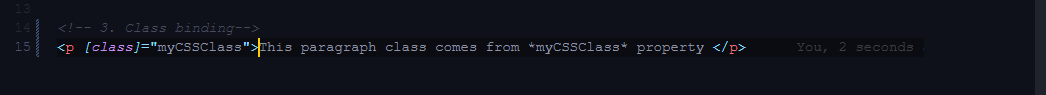
**<HTML Tag [class]=”component variable holding class name”>**

**Ex: <p [class]=”myClass”>**

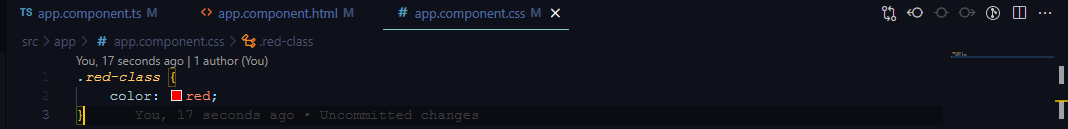
* + Add the below code in the app.component.ts file,



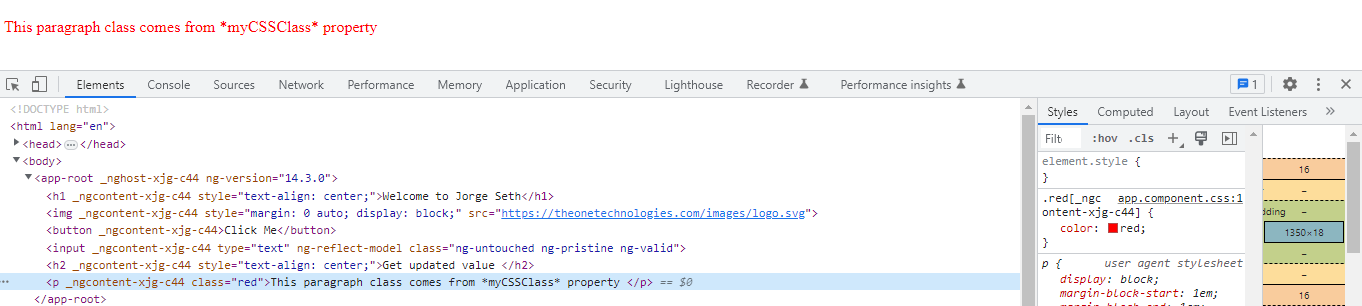
* + Add the below changes in the view app.component.html file.



* + And Add the below content in app.component.css



* + And the **output** look like this.



1. **Style Binding**
   * **Style binding** is usedto bind data from components into HTML style properties. The syntax is as follows:

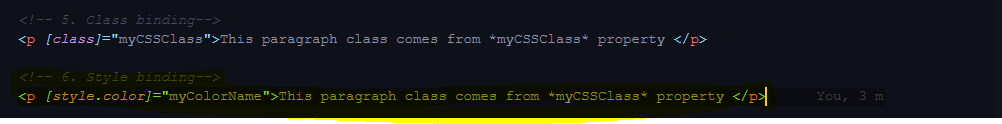
**<HTML Tag [style.STYLE]=”component data”>**

**Ex: <p [style.color]=”myColorName”>**

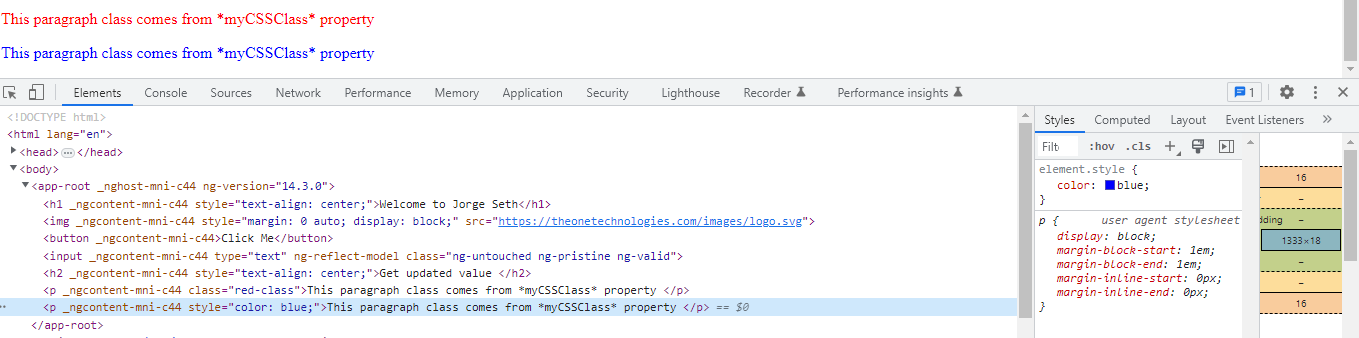
* + Add below changes in app.component.ts file.



* + Add below changes in app.component.html file.



* + And the output like this,



**Basic Routing**

* **What is routing?**
  1. When the user wants to navigate from one URL to another. It means one component to another component by using angular routing.
  2. Before the use of routing we can add the router with **Yes** when creating the new angular application.

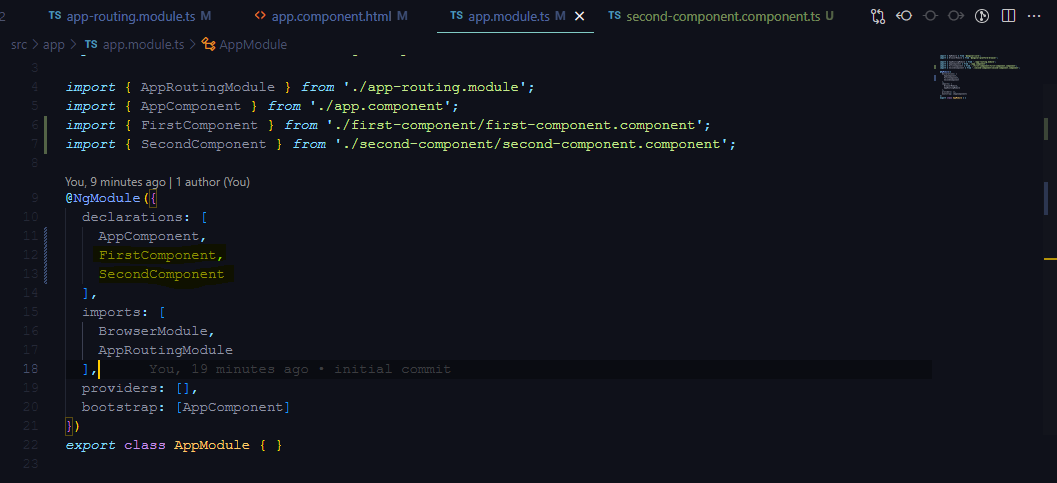
**? Would you like to add Angular routing? Yes**

* 1. First of all, we adding the multiple components for routing.

**Ex: ng g c first**

**Ex: ng g c second**

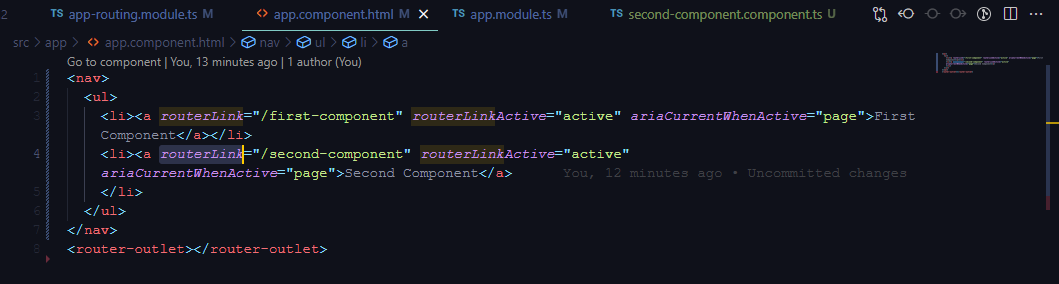
* 1. Angular auto updates the components in declarations in **app.module.ts** file.



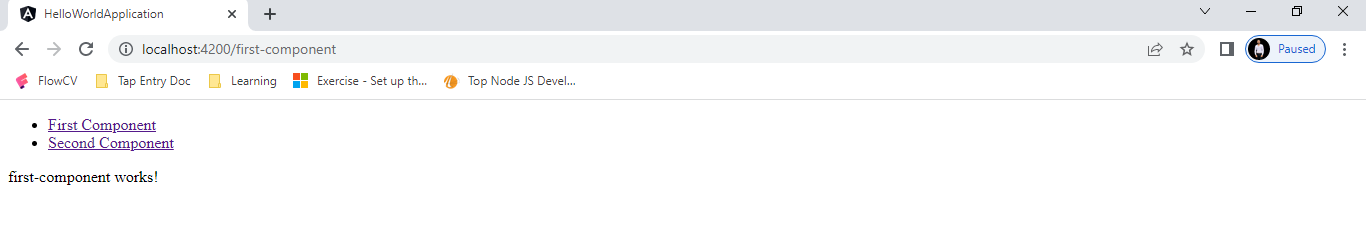
* 1. Navigate the file **app-routing.module.ts**
  2. Add the routes with name and specify the components for navigating.



* 1. Finally, Navigate one route to another route via use of **routerLink.**



* 1. **Output** like this.



**HERE IS THE COMMAND TO CREATE YOUR BUILD WHICH YOU HAVE TO UPLOAD ON IIS SERVER**

\*\*From Now onwards you have to upload your task on IIS, **only need to upload the production build on IIS server.**

ng build --configuration=production --base-href  
ng build --configuration=development --base-href

Reference URL: <https://www.c-sharpcorner.com/article/deply-of-a-angular-application-on-iis/>

Also refer to the TypeScript documentation and explore things around. (Don’t hesitate to implement new things from the documentation into the given task.) <https://www.typescriptlang.org/docs/>