

Lesson 02 Demo 02

Data Binding

Objective: To understand how to work with data binding in Angular

Tools required: Ubuntu, Visual Studio, Node.js, and Angular

Prerequisites: Knowledge of JavaScript, Typescript, and Node.js

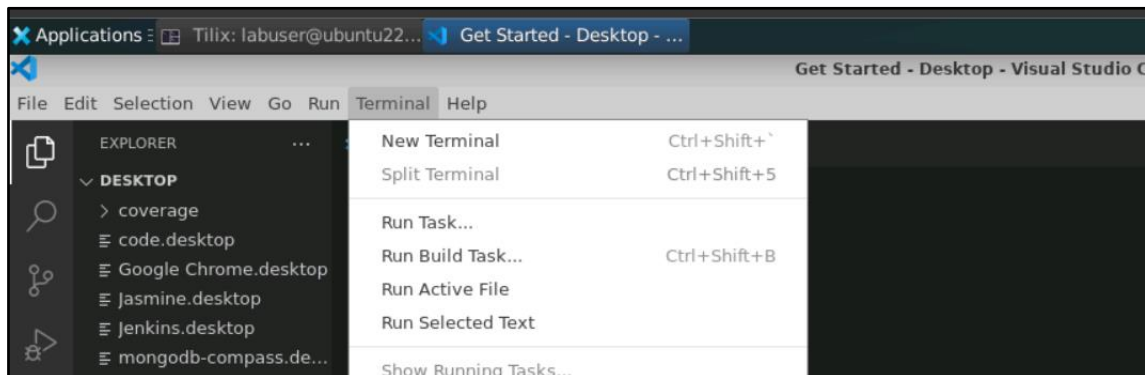
Steps to be followed:

1. Create an Angular application with data binding

Step 1: Create an Angular application with data binding

1.1 Open Visual Studio and open New Terminal

1.2 Create an Angular application using the **ng new myapp** command



? Would you like to share pseudonymous usage data about this project with the Angular Team at Google under Google's Privacy Policy at <https://policies.google.com/privacy>. For more details and how to change this setting, see <https://angular.io/analytics>. [Yes](#)

Thank you for sharing pseudonymous usage data. Should you change your mind, the following command will disable this feature entirely:

1.3 Navigate to the application using the **cd myapp** command

```
● labuser@ubuntu2204:~$ cd myapp
○ labuser@ubuntu2204:~/myapp$
```

```
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL GITLENS SQL CONSOLE
○ labuser@ubuntu2204:~$ ng new myapp
```

1.4 Create a new component with the name **test** using the **ng g c test** command

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS SQL CONSOLE
● labuser@ubuntu2204:~/Desktop/angular-project/myapp$ ng g c test
CREATE src/app/test/test.component.css (0 bytes)
CREATE src/app/test/test.component.html (19 bytes)
CREATE src/app/test/test.component.spec.ts (585 bytes)
CREATE src/app/test/test.component.ts (194 bytes)
UPDATE src/app/app.module.ts (467 bytes)
○ labuser@ubuntu2204:~/Desktop/angular-project/myapp$
```

1.5 Copy the below code into the **test.component.ts** file:

```

TS test.component.ts U x <> test.component.html U
src > app > test > TS test.component.ts > TestComponent
1  import { Component } from '@angular/core';
2
3  @Component({
4    selector: 'app-test',
5    templateUrl: './test.component.html',
6    styleUrls: ['./test.component.css']
7  })
8  export class TestComponent {
9    public name = "Mark";

```

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

```

@Component({
  selector: 'app-test',
  templateUrl: './test.component.html',
  styleUrls: ['./test.component.css']
})
export class TestComponent {
  public name = "Mark";
}

```

1.7 Copy the below code into the **test.component.html** file:

```

TS test.component.ts U  <> test.component.html U x
src > app > test > <> test.component.html > ...
  Go to component
  1  <!-- able to access property define in class using interpolation -->
  2  <h1>Welcome {{ name }}</h1>
  3
  4  <h1>{{ 'Hello ' + name }}</h1>
  5
  6  <!-- able to use JavaScript property inside interpolation -->
  7  <h1>{{ name.length }}</h1>
  8
  9  <!-- able to use JavaScript functions inside interpolation -->
 10  <h1>{{ name.toUpperCase() }}</h1>
 11

```

<h1>

Welcome {{name}}

</h1>

<h1>

{{'Hello' + name}}

</h1>

<h1>

{{name.length}}

</h1>

<h1>

{{name.toUpperCase()}}

</h1>

1.8 Start the server using the **ng s -o** command

```

labuser@ubuntu2204:~/Desktop/angular-project/myapp$ ng s -o
✓ Browser application bundle generation complete.

Initial Chunk Files | Names          | Raw Size
vendor.js           | vendor         | 2.04 MB
polyfills.js        | polyfills      | 314.26 kB
styles.css, styles.js | styles        | 209.39 kB
main.js             | main           | 7.67 kB
runtime.js          | runtime        | 6.51 kB

| Initial Total | 2.56 MB

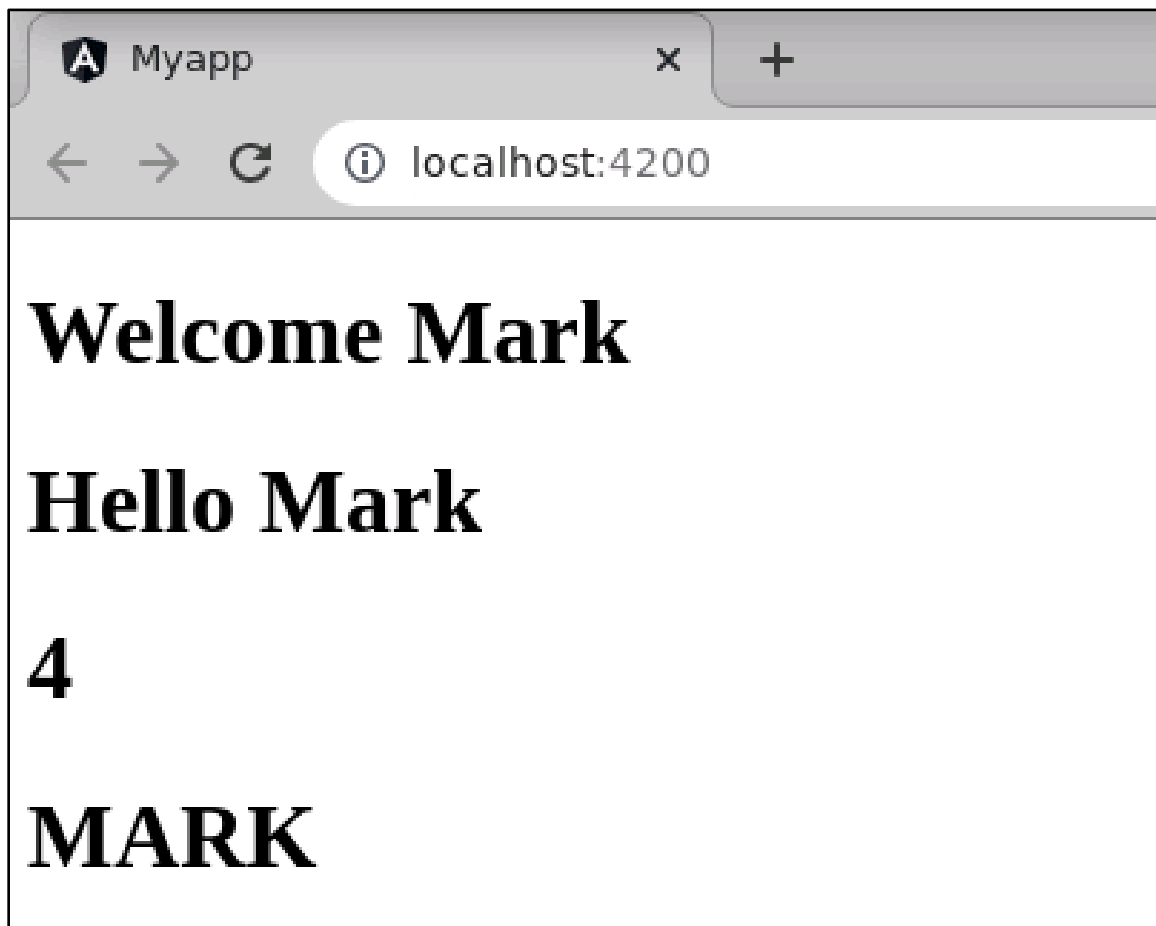
Build at: 2023-02-01T18:53:58.507Z - Hash: d671567c6946c425 - Time: 9044ms

** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

✓ Compiled successfully.

```

1.9 Open the browser and verify the output
<http://localhost:4200/>



Note: The above output shows the data binding in the Angular application.