Two-Way Data Binding

Contents

1 PROJECT ABSTRACT

Real-time synchronization of data between the UI and the component is a core feature in modern web applications. This assignment focuses on mastering **Two-Way Data Binding** in Angular, allowing seamless data flow between the component class and the template.

The goal is to create a simple **Input Display Single Page Application (SPA)** where user input in a text field is instantly reflected elsewhere on the screen. This will enhance understanding of how Angular's two-way binding works using the `[(ngModel)]` directive.

2 PROBLEM STATEMENT

You are tasked with building a **Two-Way Data Binding SPA** using Angular.
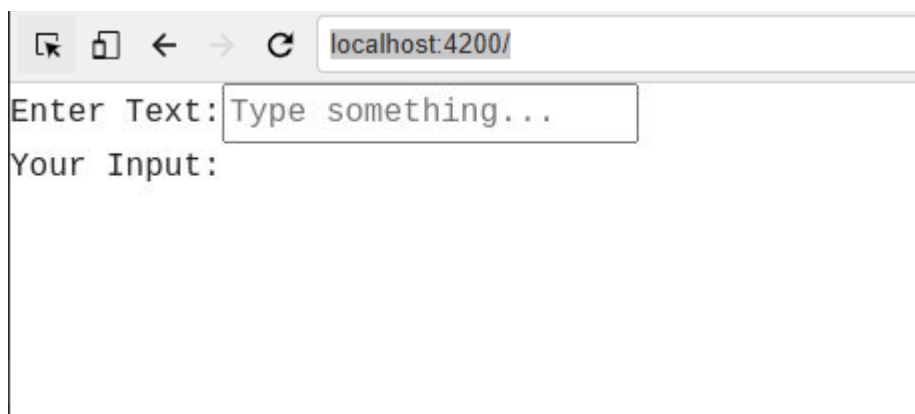
The application should:

- Provide an input field for the user to type text.
- Display the entered text dynamically below the input field.
- Reflect real-time updates of user input without page reload.
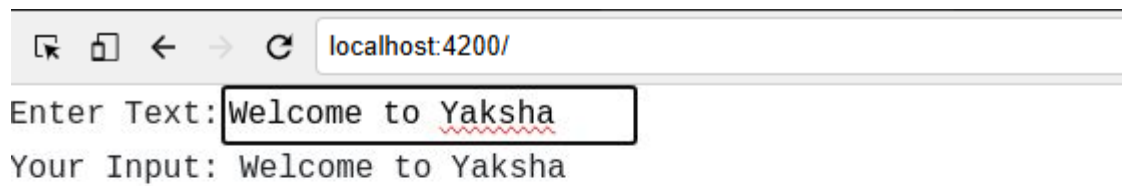- Use Angular's `[(ngModel)]` directive for two-way data binding.

3 PROPOSED TWO-WAY DATA BINDING APPLICATION WIREFRAME

UI needs improvisation and modification as per given use case and to make test cases passed.

3.1 SCREENSHOTS

```
Enter Text: Welcome to Yaksha
Your Input: Welcome to Yaksha
```

4 BUSINESS-REQUIREMENT:

As an application developer, develop the Two-Way Data Binding (Single Page App) with below guidelines:

User Story #	User Story Name	User Story
US_01	Welcome Page	As a user I should be able to visit the welcome page as the default page. Acceptance criteria: **AppComponent:** 1. Should display a **label** with the text **"Enter Text:"**. 2. Should provide an **input field** for entering text. 3. The input field should have a placeholder text **"Type something..."**. 4. Below the input, display the text **"Your Input:"** followed by the user's entered text. 5. As the user types, the entered text should be dynamically updated and displayed in real-time using **Two-Way Data Binding ([(ngModel)])**. **** Kindly refer to the screenshots for any clarifications. ****

5 CONSTRAINTS

1. You should be able to press the "TAB" key and "SHIFT + TAB" to navigate from top field to bottom field and vice-versa.

6 MANDATORY ASSESSMENT GUIDELINES

1. **All actions like build, compile, running application, running test cases will be through Command Terminal.**

2. **To open the command terminal the test takers, need to go to**

   Application menu (Three horizontal lines at left top) -> Terminal ->New Terminal.

3. **This editor Auto Saves the code.**

4. **These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.**

5. **This is a web-based application, to run the application on a browser, use the internal browser in the workspace. Click on the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.**
   Note: The application will not run in the local browser

6. **You can follow series of command to setup Angular environment once you are in your project-name folder:**
   a. **npm install -> Will install all dependencies -> takes 10 to 15 min.**
   b. **npm run start -> To compile and deploy the project in browser. You can press the <Ctrl> key while clicking on localhost:4200 to open the project in the browser -> takes 2 to 3 min.**
   c. **npm run test -> to run all test cases. It is mandatory to run this command before submission of workspace -> takes 5 to 6 min.**

7. **Once you are done with development and ready with submission, you may navigate to the previous tab and submit the workspace. It is mandatory to click on "Submit Assessment" after you are done with code.**