**NAME- AMAN KUMAR**

**COURSE- B.VOC SOFTWARE DEVELOPMENT**

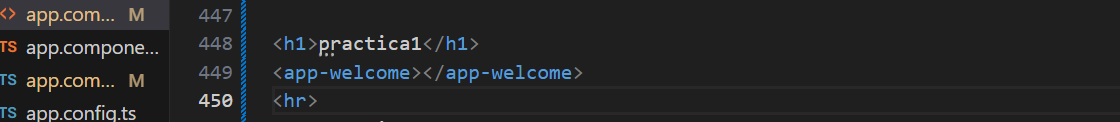
**ROLL NO- 20232703**

**SUBJECT- FULLSTACK**

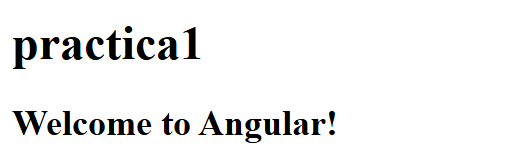
**6. Create and Display a Basic Component**

* Create a component named welcome.
* Display a welcome message (<h2>Welcome to Angular!</h2>).
* Add the component to app.component.html.

**CODE:-**



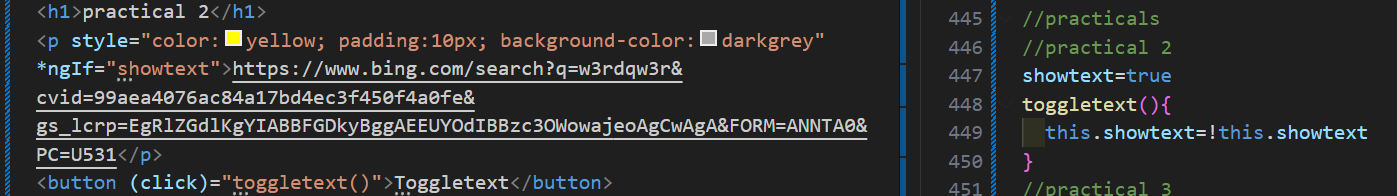
**Output:-**



**7. Create a Button to Show/Hide Text**

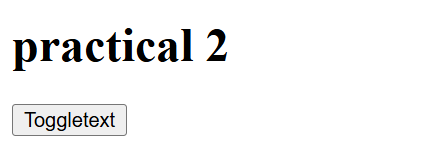
* Add a button with Toggle Message text.
* Use \*ngIf to show/hide a message when the button is clicked.
* Use a function to toggle the message visibility.

**CODE:-**



**Output:-**

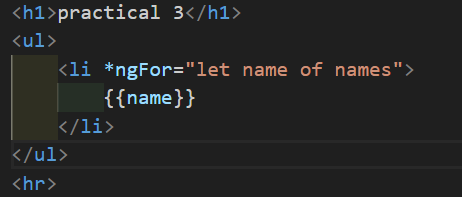


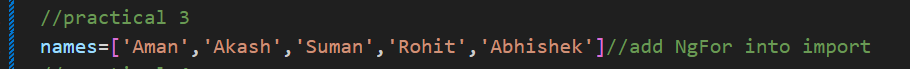


**8. Display a List Using \*ngFor**

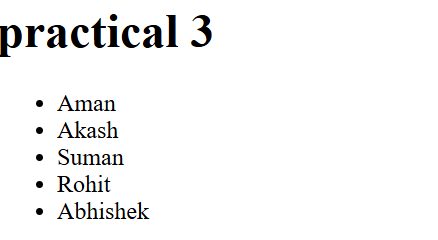
* Create an array of names or items.
* Use \*ngFor to loop through and display the items in a list.
* Add the list to the app.component.html.

**CODE:-**





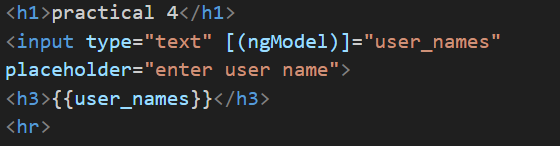
**Output:-**



**9. Implement Two-Way Data Binding with ngModel**

* Add an input field to take the user's name.
* Display a personalized greeting message that updates dynamically
* Use [(ngModel)] to bind data between the input and the component.

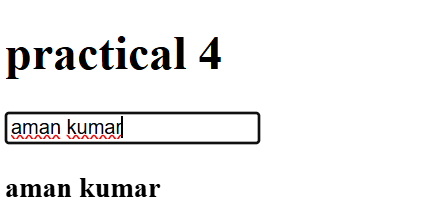
**CODE:-**





**Output:-**

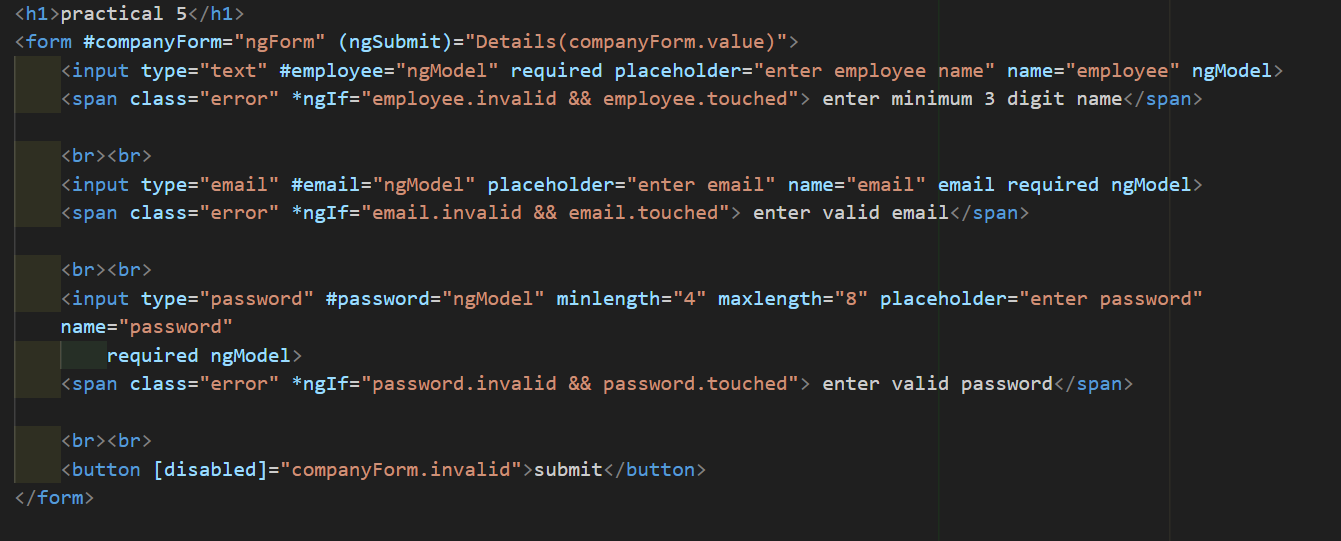


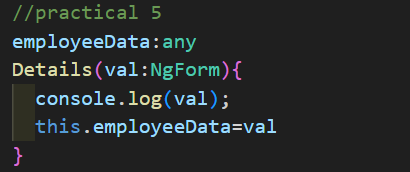


**10. Form Handling and Validation in Angular**

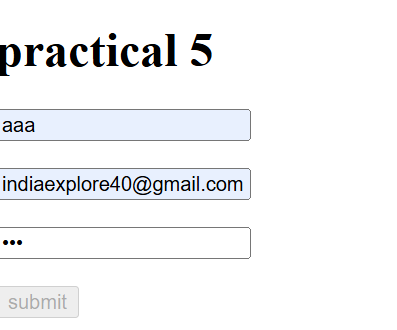
* Create a form with name, email, and password fields.
* Validate required fields and email format.
* Display error messages if validation fails.

**CODE:-**





**Output:-**

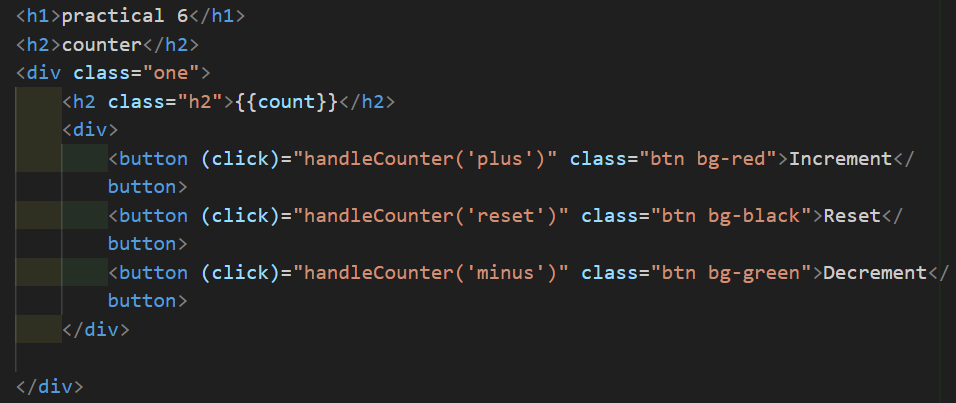


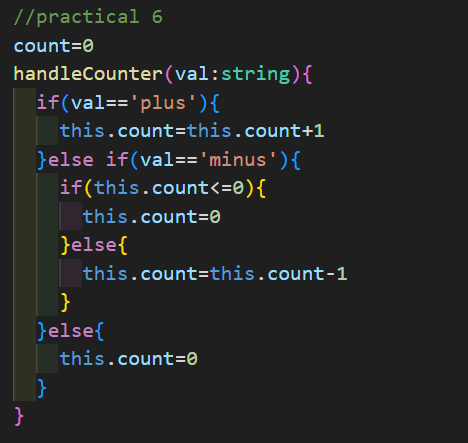


**11. Build a Simple Counter Application**

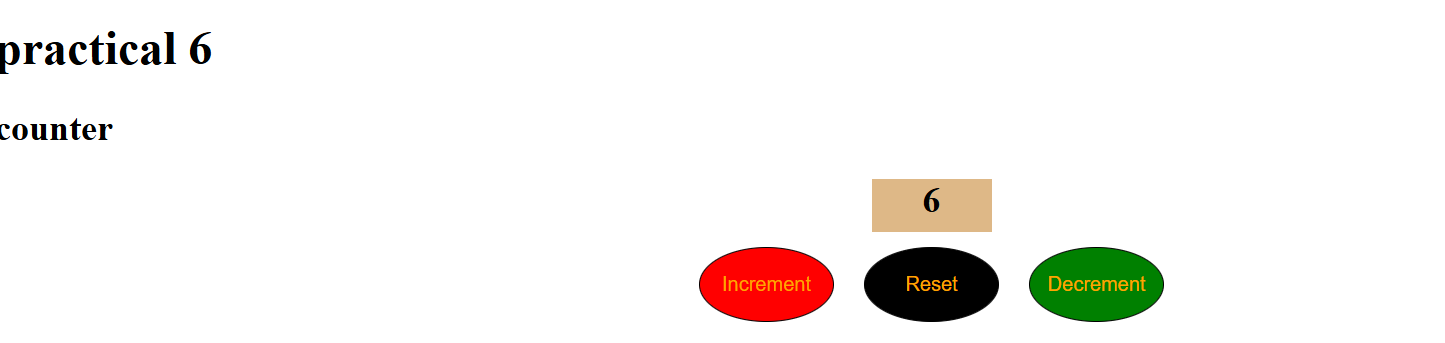
* Create a counter variable with an initial value of 0.
* Add buttons to increment and decrement the counter.
* Display the updated counter value.

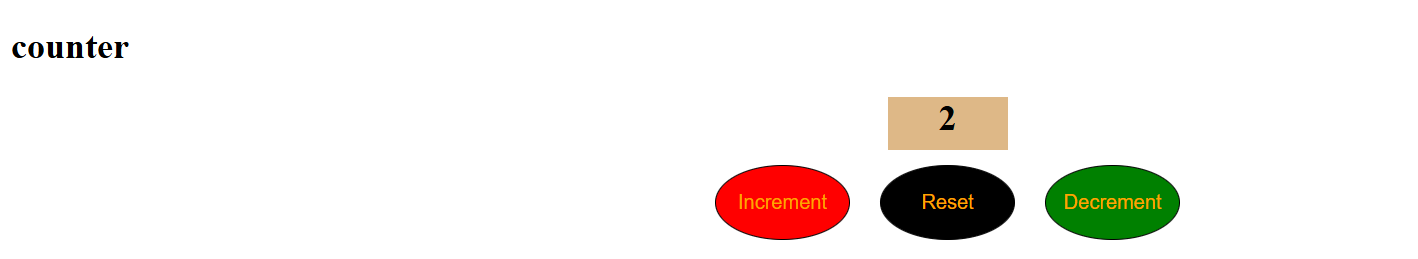
**CODE:-**





**Output:-**

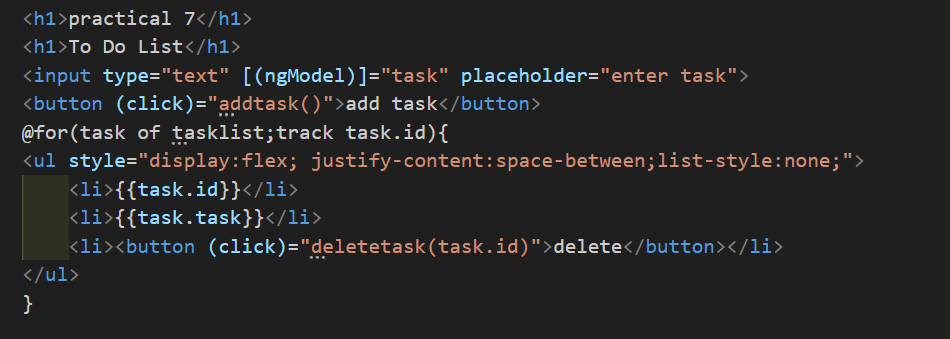


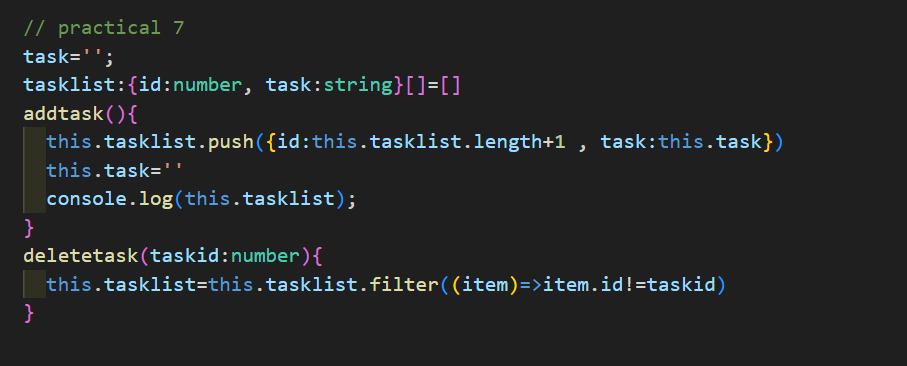


**12. Build a To-Do List Application OR Your application**

* Add an input field to add new tasks.
* Display the task list using \*ngFor.
* Allow marking tasks as complete and deleting them

**CODE:-**





**Output:-**

