

### Question 1:

You are building an Angular component that receives a **user object** from a parent component and displays the user's name. However, the parent updates the user's name **without changing the object reference**.

- Implement a **UserComponent** that detects changes in the **user** object using both **ngOnChanges()** and **ngDoCheck()**.
- Log the changes in each lifecycle hook.
- Explain why **ngOnChanges()** does not detect some updates, while **ngDoCheck()** does.

### Question 2 :

You are tasked with creating a **CardComponent** that allows users to project content inside it using **<ng-content>**.

- Implement a **CardComponent** that uses **content projection** to allow a dynamic title and content.
- Use **ngAfterContentInit()** to log when the projected content is available.
- In the **AppComponent**, pass different content inside the **CardComponent**.

### Question 3:

You have a component containing a child element

**<p #message>Welcome to Angular!</p>**.

- Use **@ViewChild()** to access this element inside **ngAfterViewInit()**.
- Modify the text inside **ngAfterViewInit()** and explain why **ngAfterViewChecked()** is needed to track further updates.

#### Question 4:

You are building a **UserProfileComponent** that accepts projected content inside a **<ng-content>**.

- Use **ngAfterContentChecked()** to log whenever projected content changes.
- In **AppComponent**, dynamically change the projected content using a button click.

#### Question 5:

You have a **CartComponent** that maintains a list of selected products.

- Implement a method **that adds a new product to the cart** and updates the total price.
- Use **ngDoCheck()** to **detect deep changes** in the cart object.
- Use **ngAfterViewChecked()** to **detect changes in the displayed total price**.