**Lab assignments**

1. In your app componemt include the product component you made before so that the product component become child in the app component.
2. In the product component make a method name it renderValues() .

1. In the app component make a button display data when click on it will call the renderValues method of the child. (viewchild)

Note: in this logic you may use ngIf structural directive to make the table f products hidden first then when click on the button from parent make its value changes.

Bonus: Search for subject behaviour service.

1. What it do?
2. Make a demo about it (big bouns)

**Pipes**

**Task 1: Display a Price using CurrencyPipe**

* **Component**: price.component.ts
* **Task**:
  1. Create a PriceComponent.
  2. Define a price variable in the component class (e.g., price = 1500).
  3. In the template, display the price using the **currency pipe** in the default currency (USD).
  4. Add a button to switch between USD and EUR using the currency pipe.

**Task 2: Format a Date using DatePipe**

* **Component**: launch-date.component.ts
* **Task**:
  1. Create a LaunchDateComponent.
  2. Define a launchDate variable (e.g., launchDate = new Date()).
  3. In the template, display the launch date using the **date pipe** in different formats (shortDate, fullDate, and yyyy-MM-dd).
  4. Add a dropdown to switch between these formats dynamically.

**Task 3: Create a Uppercase Pipe**

* **Component**: product-name.component.ts
* **Task**:
  1. Create a ProductNameComponent.
  2. Define a productName variable (e.g., productName = 'laptop').
  3. In the template, display the product name using a pipe that converts the name to uppercase.
  4. Implement the custom UppercasePipe.

**Task 4: Filter a List of Products using FilterPipe**

* **Component**: product-list.component.ts
* **Task**:
  1. Create a ProductListComponent.
  2. Define an array of product names (e.g., ['Laptop', 'Phone', 'Tablet']).
  3. Add an input field in the template to capture the user's search term.
  4. Create a simple custom pipe to filter the list of products based on the search term.
  5. Display only the products that match the search term.

**Task 5: Truncate a Long Text using Custom Pipe**

* **Component**: description.component.ts
* **Task**:
  1. Create a DescriptionComponent.
  2. Define a description variable with a long text.
  3. Create a custom TruncatePipe that limits the description to a specified number of characters (e.g., first 50 characters).
  4. Use the custom pipe to display a truncated description in the template.