---------========== TEST EXAM ===========---------

Angular 5+ Components => Used Angular 7

1. Create an Angular application (online store) using Angular-CLI, which consists of two or more modules:

    - AppModule,

    - CartModule,

    - ProductsModule,

    - OrdersModule

    - CoreModule (that in a single variant)

    - SharedModule (what is used many times)

**Note: Created all the mentioned modules**

2. Create a ProductListComponent that delegates the display of the product list

   component ProductComponent. Provide the component with the help of the getProducts () method of the ProductService.

   Create an entity model. Use one or two optional fields. Interface + Class.

   Implement the ability to add goods to the cart (click) if the product is in stock (available).

   The product must be added to the basket ProductListComponent. ProductComponent to implement as a presentation component.

   If there is no product, make the button unavailable.

**Note: Created the Components and implemented the functionalities**

3. Create a CartListComponent that displays a list of purchased items.

   In addition to the list of goods purchased, display the number of goods purchased and the total amount.

**Note: Created the Component and implemented the functionalities**

4. Create a СartItemComponent component that displays one entity from the list above.

   Use this component in CartListComponent.

   Implement the ability to change the property of an entity (quantity), delete an entity.

**Note: Created the Components and implemented the functionalities**

5. If possible, use decorators: @Input (), @Output ().

**Note: Used both @input() and @output decorators CartItemComponent**

6. If possible, use OnPush strategy for components

**Note: Did not use it but if required we can implement**

7. If possible, use two or more methods of the component life cycle hook.

**Note: Used OnInit in all components and AfterViewInit in AppComponent**

8. Use the DOM event (click, wheel, blur ...).

**Note: Used click in many places. for example CartListComponent**

9. Create or modify a service (CartService) to provide data to components, counting quantities and amounts.

**Note: Implemented the logic**

10. Add an <h1 #appTitle> </ h1> element with a template variable to the AppComponent markup.

    Use @ViewChild, access the DOM element of the template,

    set the title for the application from the class.

**Note: Implemented in AppComponent**

11. When hovering the mouse or clicking on a list item (CartItem), style it by changing the background.

    To do this, create a directive and register it in the SharedModule.

    Use @HostBinding, @HostListener decorators.

**Note: Implemented it and can be seen in shared folder**

12. If possible, apply the directive ngClass or ngStyle.

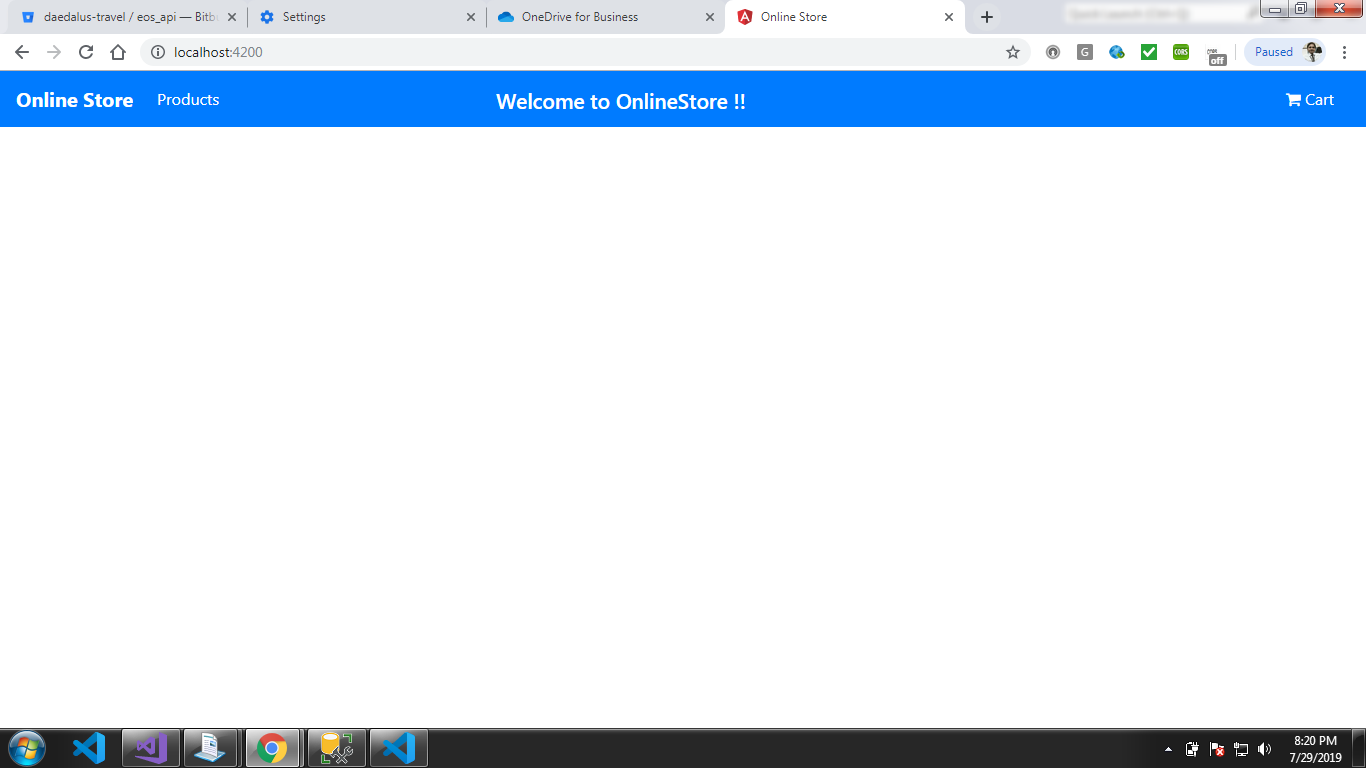
**Note: Implemented ngClass in ProductlistComponent**

13. Implement any other functionality related to online store - whatever you think is relevant and could demonstrate your skills.

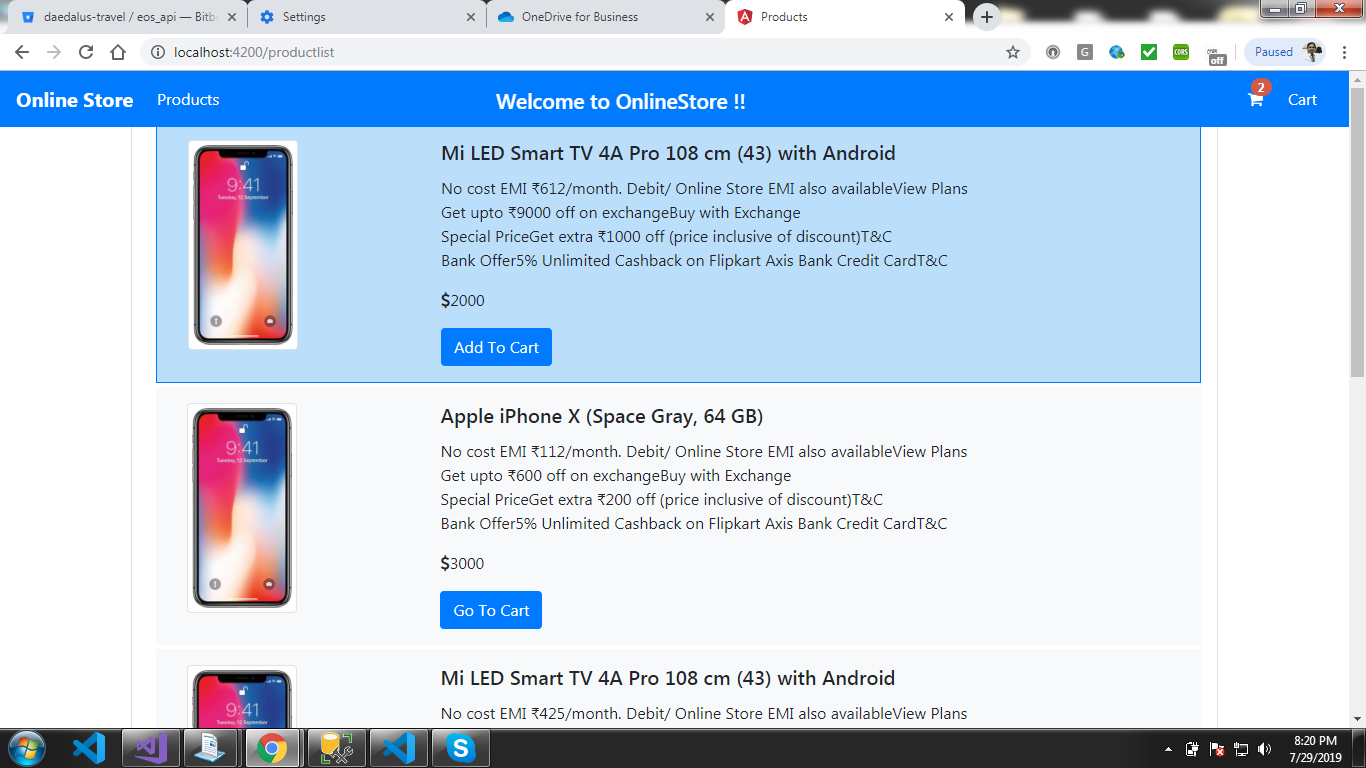
14. Describe the changes in the project.

15. Upload your project to github and send me a link.

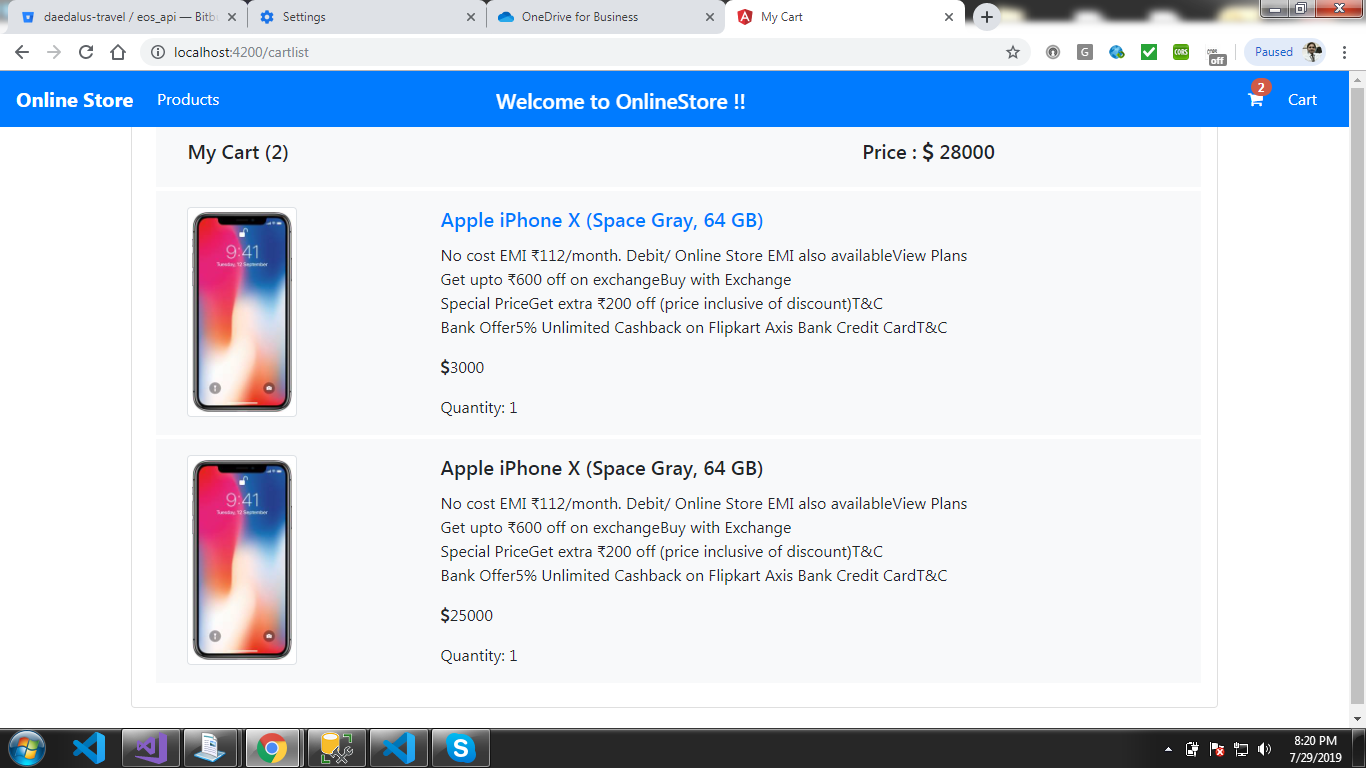
Screen:-1 Home



Screen:-2 Product List



Screen:-3 CartItemList



Screen:-4 CartItem

