

Customer buying products from an 'online' store.  
Customer can buy (maybe refund).  
Customer can pick which product to buy from a list (by number or name)  
Customer can pick how many of the product to buy.  
Customer can view all the items in their shopping cart.  
Customer can use discount codes at checkout.

System will display product name, product id and price.  
System will ensure the customer cannot buy more of a product than available.  
System will ask the user if they want to add more items to their shopping cart.  
System will ask the user if they want to check out.

## **Classes**

### **Product**

#### Sub-class

-> discounted Products (using virtual keyword, maybe get\_price)

### **SupermarketChain**

#### Sub-class

-> Adelaide Supermarket (using virtual pure for Abstract class)

-> Canberra Supermarket (virtual pure)

### **Customer**

## **Variables**

### **Products**

ID

- Gives the product ID

Name

- Gives the product name

Price

- Price of product

Current Stock

- Number of available stock of the product

Discounted products (get\_price)

- Applies discounts on specific products

### **Shopping Cart Class**

Shopping Cart

- Current list of the products that are selected by the customer

Total Price

- Current total price of the products selected

Total Items

- Increment and decrement the amount of items in the shopping cart

### **Customer Class**

Name

- Name of customer

Address

- Address of customer

### **Supermarket**

Supermarket Chain

Adelaide & Canberra

- Different locations for delivery / pickup

## **Methods**

### **Products**

GetID

- Retrieves the specific product ID

GetName

- Retrieves the specific product name

GetPrice

- Retrieves the Price of the product

GetStock

- Retrieves the specific number of available stock of the product

Discounted products (get\_price)

- Retrieves and Applies discounts on specific products ( on top of the normal price )

### **Shopping Cart Class**

Remove Items

- Allows the removal of items in the shopping cart

Add Item

- Allows the addition of items in the shopping cart

Clear Cart

- Clears the entire shopping cart

## **Customer Class**

GetName

- Retrieves the name of the customer

SetName

- Sets the name of the customer

GetAddress

- Retrieves the address of the customer

SetAddress

- Sets the address of the customer

## **Supermarket**

GetDiscountCode

- Retrieves discount code depending on location chosen

Adelaide & Canberra

- Retrieves the specific discount code

## **Testing**

1. Logging in as a customer with Name and Address
  2. Test all functions in the Customer Class
  3. Test all functions in the Product Class
  4. Test all functions in the Supermarket Class
  5. If the code is running correctly the code will display:
    - a. -
  6. Test for different Scenarios
    - a. Customer buys 3 products, removes 2, adds another 2 different ones
    - b. Customer buys 5 products and removes 6 products
    - c. Customers decides not to buy items and cancels whole order
- 
1. Test that a maximum of 5 products can be added to the shopping cart
  2. Test that the discount code provided from the Adelaide and Canberra classes apply the discount during the checkout phase.
  3. Test that a product with 0 current stock cannot be purchased
  4. Test that the customer cannot access checkout without having at least 1 item in the shopping cart.
  - 5.

