Inheritance:

**Discount For Customer**

In the bank, customers can be **Normal, Priviledged, SeniorCitizen** and so on. The bank also introduces an offer where privileged customers get a 30% off on the bill while senior citizens get 12% off. Let’s implement the inheritance with discount yet again a better understanding.  
  
1. Create Customer, PrivilegedCustomer & SeniorCitizenCustomer class with data members as given below.  
2. Implement GenerateBillAmount Method as per the specification.  
  
Write a program to get the customer details and display bill, discount amount based on customer type.

Consider a class named **Customer** with the following protected attributes

|  |  |
| --- | --- |
| **Data Type** | **Attributes** |
| string | \_name |
| string | \_address |
| string | \_mobileNumber |
| int | \_age |

Include the following public method in **Customer**class.

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public void DisplayCustomer() | This method displays the customer details. |

Consider a class **SeniorCitizenCustomer** which extends the class **Customer**.  
  
Include the following public method in **SeniorCitizenCustomer**class.

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| double GenerateBillAmount(int amount) | This method is used to calculate and return the payment amount where the discount is 12%. |

Consider a class **PrivilegeCustomer** which extends the class **Customer**.  
  
Include the following public method in **PrivilegeCustomer**class.

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| double GenerateBillAmount(int amount) | This method is used to calculate and return the payment amount where the discount is 30%. |

Consider a driver class named **Program** which creates an instance of the above mentioned classes and their functionalities are tested.  
Use **base** Keyword to call the base class constructor.  
Read the respective customer details (Senior Citizen or Privileged) and call the corresponding GenerateBillAmount() method based on the choice as shown in the sample output.

Polymorphism, Abstract Classes and Interfaces

**Basic Interface**

Write a program to read and display the stall details using display method in stall interface.

Create an interface **IStall**with the following abstract method.

|  |  |
| --- | --- |
| **Method** | **Description** |
| void display() | Display the Stall details |

Consider a class named **GoldStall**with the following private attributes which implements the **IStall** interface.

|  |  |
| --- | --- |
| **Data Type** | **Attributes** |
| string | \_name |
| string | \_details |
| double | \_cost |
| string | \_ownerName |

Implement the abstract method of **IStall** in the class **GoldStall** which should display the details of the stall.   
Assign the GoldStall reference to the IStall reference and call display method using interface reference.

Consider a class named **ExecutiveStall**with the following private attributes which implements the IStall interface.

|  |  |
| --- | --- |
| **Data Type** | **Attributes** |
| string | \_name |
| string | \_details |
| double | \_cost |
| string | \_ownerName |
| int | \_numberOfTVSet |

Implement the abstract method of **IStall** in the class **ExecutiveStall** which should display the details of the stall.  
Assign the ExecutiveStall reference to the IStall reference and call display method using interface reference.

Consider a class named **PremiumStall**with the following private attributes which implements the IStall interface.

|  |  |
| --- | --- |
| **Data Type** | **Attributes** |
| string | \_name |
| string | \_details |
| double | \_cost |
| string | \_ownerName |
| int | \_numberOfProjector |

The methods for **getters**, **setters** and **constructors** are given in the template code.  
Implement the abstract method of **IStall** in the class **PremiumStall** which should display the details of the stall.  
Assign the PremiumStall reference to the IStall reference and call display method using interface reference.

Consider class **Program**with **Main** method to get the details of the stall from the user and display the details using the Display() method of the respective class. Display “**Invalid Input**” when user gives invalid choice of stall.