

Write a class for the problems below. Then instantiate the object and write a loop to repeatedly test the object.

1. Create a membership class to be used for a health club. The class and ultimately object should have the methods below. Once created, write a loop to repeatedly prompt the user for the items below and display all the items along with the membership cost. Use **ctrl+z** to stop. You can use the same object over again within the loop.

Member_First_Name

This method should allow a first name to be entered but also used to show the first name in the object. The default name should be “Not Entered”. That is if the user attempts to display a first name before they entered one, this message should display.

Member_Last_Name

This method should allow a last name to be entered but also used to show the last name in the object. The default name should be “Not Entered”. That is if the user attempts to display a first name before they entered one, this message should display.

Member Age

This method should allow the age of the member. The default age should be 18 if nothing is entered.

Membership Type

Allow the user to enter “Gold”, “Silver” or “Bronze”. Default to “Bronze”.

Membership Cost

Compute cost of the membership based on the type of membership. Give a 10% discount for members who are older than 50.

“Gold” membership cost is 1200.00

“Silver” membership cost is 1000.00

“Bronze” membership cost is 500.00

Once the class is created, write a program to instantiate the object using this defined class. Then ask the user for the above information and load it into the object. Lastly, display all information from the object.

input	process	output
Member first name Member last name Member age	If first or last name not entered → set to "Not Entered" If age not entered → default age = 18	First name Last name Age Membership type

Membership type: Gold / Silver / Bronze	Membership type default → "Bronze" Determine membership cost: Gold → 1200 Silver → 1000 Bronze → 500 If age > 50 → apply 10% discount	Final membership cost

2. Create a class representing a computer asset. The class should have the following attributes (methods). Once created, write a loop to repeatedly prompt the user for the items below and display all the items along with the cost. Use **ctrl+z** to stop. You can use the same object over again within the loop.

Computer Make

Allow the user to enter the make of the computer or “Lenovo”, “HP”, “Apple”. Default the make to “Lenovo”.

Computer Model

Allow the user to enter the model using a string with no spaces in it., i.e. L12345. Default the model to LE100A.

Computer CPU

Allow the user to enter CPU type of either “Intel” or “AMD”. Default to “Intel”.

Computer RAM

Allow the user to enter string of “8GB”, “16GB” or 32GB”. Default to “32GB”.

Computer Hard Drive

Allow the user to enter “SSD” or “HD”. Default to “HD”.

Computer Type

Allow the user to enter “Laptop”, “Notebook”, “Tablet”, or “Desktop”. Default to “Laptop”.

Computer Cost

Compute the cost as follows:

Lenovo add 1200.00

HP add 1000.00

Apple add 2000.00

Intel add 200.00

RAM: “16GB” add 300.00, “32GB” add 500.00

Hard Drive: “SSD” add 500.00

Computer Type: “Laptop” add 300.00, “Tablet” subtract 200.00

input	process	output
Make: Lenovo / HP / Apple Model CPU: Intel / AMD RAM: 8GB / 16GB / 32GB Hard drive: SSD / HD Type: Laptop / Notebook / Tablet / Desktop	Apply default values if blank Add cost based on selections: Make Cost Lenovo +1200 HP +1000 Apple +2000 CPU Cost Intel +200 RAM Cost 16GB +300 32GB +500 Hard Drive Cost SSD +500 Computer Type Cost Laptop +300 Tablet -200 Add all adjustments to compute total cost	Make Model CPU RAM Hard drive Computer type Total cost

