Class Customer

Private String custNum

Private String name

Private Integer numCDs

Public Module Customer()

Set custNum = “ “

Set name = “ “

Set numCDs = 0

End Module

Public Module Customer (String newCustnum, String newName, Integer newNumCDs)

Set custNum = newCustNum

Set name = newName

Call setNumCDs(newNumCDs)

End Module

End Class

Public Function String getCustNum()

Return custNum

End Function

Public Module setCustNum(String newCustNum)

Set custNum = newCustNum

End Module

Public Function String getName()

Return name

End Function

Public Module setName (String new Name)

Set name = newName

End Module

Public Function Integer getNumCDs()

Return numCDs

End Function

Public Module setNumCDs(Integer newNumCDs)

Constant Integer MAX\_CDS = 5000

If newNumCDs > 0 AND newNumCDs <= MAX\_CDS Then

Set numCDs = newNumCDs

Else

Set numCDs = 0

Display newNumCDs, “ is invalid; it must be between 1 and “, MAX\_CDS

End If

End Module

|  |  |  |
| --- | --- | --- |
| An Internet music store sells all of its CDs at $10 each. It rewards customers who buy large quantities of CDs using the following discount table. Quantity Purchased | Discount |  |
| 10 - 19 | 20 % |  |
| 20 - 49 | 30 % |  |
| 50 - 99 | 40 % |  |
| 100 or more | 50 % |  |

Public module calculatePurchase()

Constant Real PRICE = 10.0

Declare Real cost

Declare Real discount

If numCDs <= 9 Then

Set discount = 0

Else If numCDs <= 19

Set discount = .20

Else If numCDS <= 49 Then

Set discount = .30

Else If numCDs <= 99 Then

Set discount = .40

Else

Set discount = .50

End If

Set cost = (1 – discount) \* numCDs \* PRICE

Display”Cost of”, numCDs, “ is$”, cost

End Module

End Class

Add to the class diagram and write the pseudocode for an application program with a main method that instantiates two objects of the Customer class--one that makes use of the default constructor and one that provides appropriate values for all the parameters in the constructor.

For one of the objects:

i. Include a call to display the customer’s number.

ii. Include a call to set the number of CDs purchased to 60

iii. Include a call to calculatePurchase()

Module main()

Declare Customer custOne

Declare Customer custTwo

Set custOne = New Customer()

Set custTwo = New Customer(“A1234”,”JohnJones”, 45)

Display “Customer number is”, custTwo.getCustnum()

Call custTwo.setNumCDs(60)

Call custTwo.calculatePurchase()

End Module

PET CLASS\*\*\*

Class Pet

Private String species

Private Integer age

Private String name

Public Module Pet()

Set species = “ “

Set age = 0

Set name = “ “

End Module

Public Module Pet(String newSpecies, Integer newAge, String newName)

Set species = newSpecies

Call setAge(newAge)

Set name = newName

End Module

Public Function String getSpecies()

Return species

End Function

Public Module setSpecies (String newSpecies)

Set species = newSpecies

End Module

Public Function Integer getAge()

Return age

End function

Public Module setAge(String newAge)

Constant Integer MAX\_AGE = 150

If newAge >= 0 AND newAge <= MAX\_AGE

Set age = newAge

Else

Display newAge, “is invalid; it must be between 0 and”,MAX\_AGE

Set age = 0

End If

End Module

Public Function String getName()

Return name

End Function

Public Module setName (String newName)

Set name = newName

End Module

Public Module displayAll()

Display “Species is “, species

Display “Age is ”, age

Display “Name is “, name

<<End Class

MAIN METHOD FOR THE PET CLASS- APPLICATION INSTANTIATING A SECOND OBJECT \*\*\*

Module main()

Declare Pet mrMax = New Pet()

Declare Pet msKitty = New Pet(“cat”, 7, “Miss Kitty”)

Call mrMax.setSpecies(“cat”)

Call mrMax.setAge(2)

Call mrMax.setName(“Mister Max”)

Display “Species:”, mrMax.getSpecies(), “ Age:”, mrMax.getAge(), “ Name:”, mrMax.getName()

Call msKitty.displayAll()

End Module