**Lab Task-10**

**Instructions: Please read carefully**

* Please rename this file as only your ID number **(e.g. 20-\*\*\*\*\*-3.doc or 20-\*\*\*\*\*-3.pdf).**
* Submit the file before **11:59 PM on 26/12/2020** in VUES section labeled **Lab task-10. If you cannot complete the full task, do not worry. Just upload what you have completed.**

|  |
| --- |
| **Question No.1**  Develop a simple application as per your wish that must demonstrate the three Object Oriented Principles and must contain modularization for all implemented task. |
| **Your code here:**    #include<iostream>  using namespace std;  class inventory //Creating class  {  protected: // Encapsulation  int code=(rand()%1000)+1;  char name[50];  public:  void setCode (int a) //Setting function  {  code = a;  }  int getCode () //Setting function  {  return code;  }  float price;  void goods() //creating function  {  cout<<"\nEnter Item Name: ";  cin>>name;  cout<<"\nItem Code: "<<getCode ();  cout<<"\nEnter Item Price: ";  cin>>price;  }  void print() //creating function  {  cout<<"\nWithout Discount";  cout<<"\n Item Code: "<<getCode ();  cout<<"\n Item Name: "<<name;  cout<<"\n Item Price: "<<price;  }  };  class billing : public inventory // Inheritance : billing class inheriting inventory class  {  protected: // Encapsulation  float discount\_percent;  public:  float discounted\_price;  void goods() // Polymorphism : Runtime Polymorphism : Function Overriding  {  cout<<"Enter Item Name: ";  cin>>name;  cout<<"Item Code: "<<getCode ()<<endl;  cout<<"Enter Item Price: ";  cin>>price;  cout<<"Enter Discount Percent: ";  cin>>discount\_percent;  cout<<"\n ----------------------\n";  discounted\_price = price - price \* discount\_percent / 100;  }  void print() // Polymorphism : Runtime Polymorphism : Function Overriding  {  cout<<"\n Item Code : "<<getCode ();  cout<<"\n Item Name : "<<name;  cout<<"\n Item Price : "<<price;  cout<<"\n Discount Percent : "<<discount\_percent;  cout<<"\n Discounted Price : "<<discounted\_price;  cout<<"\n ----------------------\n";  }  };  void call () //creating function  {  cout<<"\tHello There, Select What do you want to do \n\n 1. Create a New Invoice. \n 2. Check Inventory \n 3.Exit "<<endl;  int op;  start: // setting goto statement : unconditional jump  cout<<"\n\tOption: ";  cin>>op;  cout<<endl;  if (op==1)  {  bill: // setting goto statement : unconditional jump  int i, cnt, discount = 0, price = 0;  billing dt[100]; //class type object initialization  cout<<"How Many Items are Getting to be Billed: ";  cin>>cnt;  cout<<endl;  for(i=1; i<=cnt; i++) //using loop  {  dt[i].goods(); //calling class function  }  for(i=1; i<=cnt; i++)  {  dt[i].print (); //calling class function  }  for(i=1; i<=cnt; i++)  {  price = price + dt[i].price;  }  for(i=1; i<=cnt; i++)  {  discount = discount + dt[i].price-dt[i].discounted\_price;  }  cout<<"\n Total Cost: "<<price;  cout<<"\n Total Discount: "<<discount;  cout<<"\n\n\aAmount needed to be paid: "<<price-discount<<endl<<endl;  cout<<"Do you want to Bill another Invoice? (y/n): ";  string y;  cin>>y;  if (y=="y" || y=="Y")  {  cout<<endl;  goto bill; //using goto statement  }  else  {  cout << "Exiting";  }  }  else if (op==2)  {  cout<<"\aSorry, this portion is under construction. Only option one is available now."<<endl;  cout<<"\n\tSelect Option Again";  goto start; //using goto statement  }  else if (op==3)  {  cout<<"\a\t\t\t\tExiting\n\n\n";  }  else  {  cout<<"\t\aWrong input. Please Try Again";  goto start; //using goto statement  }  }  int main() //main function  {  call (); //calling function  } |
| **Your whole Screenshot here: (Console Output):** |