

Institute of Engineering & Management
Department of Computer Science & Engineering
Object Oriented Programming (IT) Lab for 3rd year 5th semester 2018
Code: CS594D

Date: 31/08/18

WEEK-4

Assignment-1

Problem Statement: There are 3 classes: box, box weight, box shipment

1. the member variable of box class is height, width, depth and there is a method volume which will calculate the volume of the box.
2. the box weight class also contains the above variables height, width, depth [apply inheritance] also contains an extra variable that is weight which denotes weight of the box
3. the box shipment class contains the above variable height, width, depth, weight, [apply inheritance] also contains the variable cost which denotes the cost for shipment.

Write a program in java that will calculate the volume of the box and print the value of the weight and cost of that box using the concept of multi-level inheritance.

Source code:

```
import java.util.Scanner;

class Box
{
    int height, width, length;
    Box(int n1, int n2, int n3)
    {
        length = n1;
        width = n2;
        height = n3;
    }
    int volume()
    {
        return height*width*length;
    }
}

class BoxWeight extends Box
{
    int weight;
    BoxWeight(int n1, int n2, int n3, int n4)
    {
        super(n1,n2,n3);
        weight = n4;
    }
}

class BoxShipment extends BoxWeight
{
    int unit_cost;
    BoxShipment(int n1, int n2, int n3, int n4, int n5)
    {
        super(n1, n2, n3, n4);
        unit_cost = n5;
    }
    int cost()
}
```

```

    {
        return weight*unit_cost;
    }
}

class Main
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        int ht, wdt, lt, wt, cost;
        System.out.print("Enter the length, width & height: ");
        lt = sc.nextInt();
        wdt = sc.nextInt();
        ht = sc.nextInt();
        System.out.print("Enter the weight: ");
        wt =sc.nextInt();
        System.out.print("Enter the cost per unit weight: ");
        cost=sc.nextInt();
        BoxShipment bs = new BoxShipment(lt, wdt, ht, wt, cost);
        System.out.println("The Volume is: "+bs.volume());
        System.out.println("The cost of the shipment is: "+bs.cost());
        sc.close();
    }
}

```

Screen-Shot:

```

rana@rana:~/Desktop/Java/4/box$ javac problem.java
rana@rana:~/Desktop/Java/4/box$ java Main
Enter the length, width & height: 5 4 3
Enter the weight: 90
Enter the cost per unit weight: 5
The Volume is: 60
The cost of the shipment is: 450
rana@rana:~/Desktop/Java/4/box$ |

```

Assignment-2

Problem Statement: Assume that a bank maintains 2 kinds of accounts for customers one called as savings other as current. the saving account provide interest and withdrawal facility but no cheque book facility. current has cheque but no interest. current maintain min balance. if the balance falls below the level a service charge is imposed. create a java class account that stores customer name, account no and type of account from this derive the classes CurAcc and SavAcc, to make them more specific to their requirement. Include necessary member function in order to achieve the following task.

1. accept deposit from customer and update the balance.
2. display the balance
3. compute and deposit interest
4. permit withdrawal and update value.
5. check for min balance, impose penalty if required and update the balance

Write a java program to implement this using the concept of hierarchical inheritance.

Source code:

```
import java.util.Scanner;

class Account
{
    String name;
    int acc_no, balance=0, type;
    Account(String st, int acc, int t)
    {
        name = st; acc_no = acc; type = t;
    }
    void deposit(int sum)
    {
        balance += sum;
        System.out.println("New Balance = "+balance);
    }
    void display()
    {
        System.out.println("Balance = "+balance);
    }
    void withdrawal(int sum)
    {
        if(balance-sum<0)
            System.out.println("Not enough Balance!");
        else balance -= sum;
        System.out.println("New Balance = "+balance);
    }
}

class CurAcc extends Account
{
    int min_bal = 2000, charge = 20;
    CurAcc(String st, int acc)
    {
        super(st, acc, 1);
    }
    void checkMin()
    {
        if(balance<min_bal)
        {
            balance -= charge;
            System.out.println("Balance is below minimum & service charge is deducted");
        }
        else System.out.println("Balance is above minimum");
    }
}

class SavAcc extends Account
{
    private int interest_rate = 4;
    SavAcc(String st, int acc)
    {
        super(st, acc, 2);
    }
    void interest(int month)
    {
        balance += month*interest_rate*balance/100;
        System.out.println("New Balance = "+balance);
    }
}
```

```

class Main
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        CurAcc acc1;
        SavAcc acc2;
        String name;
        int acc_no, com, flag=1, type;
        System.out.print("Enter name: ");
        name = sc.nextLine();
        System.out.print("Enter type of account: ");
        type = sc.nextInt();
        System.out.print("Enter acc no.: ");
        acc_no = sc.nextInt();
        if(type == 1)
        {
            acc1 = new CurAcc(name, acc_no);
            System.out.println("Enter the following comands:");
            System.out.println(" 1: deposit");
            System.out.println(" 2: withdrawal");
            System.out.println(" 3: check if balance is minimum");
            System.out.println(" 4: check balance");
            System.out.println(" Any other key to exit");
            do{
                System.out.print("Enter command: ");
                com = sc.nextInt();
                switch(com)
                {
                    case 1: System.out.print("Enter the amount: ");
                        acc1.deposit(sc.nextInt()); break;
                    case 2: System.out.print("Enter the amount: ");
                        acc1.withdrawal(sc.nextInt()); break;
                    case 3: acc1.checkMin(); break;
                    case 4: acc1.display(); break;
                    default: System.out.println("Bye!"); flag--;
                }
            }while(flag==1);
        }
        else if(type == 2)
        {
            acc2 = new SavAcc(name, acc_no);
            System.out.println("Enter the following comands:");
            System.out.println(" 1: deposit");
            System.out.println(" 2: withdrawal");
            System.out.println(" 3: claim interest");
            System.out.println(" 4: check balance");
            System.out.println(" Any other key to exit");
            do{
                System.out.print("Enter command: ");
                com = sc.nextInt();
                switch(com)
                {
                    case 1: System.out.print("Enter the amount: ");
                        acc2.deposit(sc.nextInt()); break;
                    case 2: System.out.print("Enter the amount: ");
                        acc2.withdrawal(sc.nextInt()); break;
                    case 3: System.out.print("Enter no. of year:");
                        acc2.interest(sc.nextInt()); break;
                    case 4: acc2.display(); break;
                    default: System.out.println("Bye!"); flag--;
                }
            }while(flag==1);
        }
    }
}

```

```

    }
    }while(flag==1);
}
else return;
sc.close();
}
}

```

Screen-Shot:

```

rana@rana:~/Desktop/Java/4/bank$ javac problem.java
rana@rana:~/Desktop/Java/4/bank$ java Main
Enter name: Ranajit Roy
Enter type of account: 1
Enter acc no.: 21325325
Enter the following comands:
  1: deposit
  2: withdrawal
  3: check if balance is minimum
  4: check balance
Any other key to exit
Enter command: 1
Enter the amount: 11000
New Balance = 11000
Enter command: 4
Balance = 11000
Enter command: 2
Enter the amount: 10000
New Balance = 1000
Enter command: 3
Balance is below minimum & service charge is deducted
Enter command: 4
Balance = 980
Enter command: 8
Bye!
rana@rana:~/Desktop/Java/4/bank$ |

```

Fig: Current account

```

rana@rana:~/Desktop/Java/4/bank$ javac problem.java
rana@rana:~/Desktop/Java/4/bank$ java Main
Enter name: Ranajit Roy
Enter type of account: 2
Enter acc no.: 23452342
Enter the following comands:
  1: deposit
  2: withdrawal
  3: claim interest
  4: check balance
Any other key to exit
Enter command: 1
Enter the amount: 10000
New Balance = 10000
Enter command: 2
Enter the amount: 2000
New Balance = 8000
Enter command: 3
Enter no. of year: 10
New Balance = 11200
Enter command: 4
Balance = 11200
Enter command: 6
Bye!
rana@rana:~/Desktop/Java/4/bank$ |

```

Fig: Savings Account