LAB - 4 & 5

/*Develop a Java program to create an abstract class named Shape that contains two integers and
an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain
only the method printArea( ) that prints the area of the given Shape.
*/

```
abstract class Shape
{
  int a=3;
  int b=4;
  abstract public void print_area();
}
class rectangle extends Shape
{
  public int area_rect;
          @Override
  public void print_area()
 {
   area_rect=a*b;
   System.out.println("The area of rectangle is: "+area_rect);
 }
}
class triangle extends Shape
{
 int area_tri;
  @Override
 public void print_area()
 {
  area_tri=(int) (0.5*a*b);
  System.out.println("The area of triangle is: "+area_tri);
 }
}
```

```
class circle extends Shape
{
 int area_circle;
  @Override
 public void print_area()
 {
  area_circle=(int) (3.14*a*a);
  System.out.println("The area of circle is: "+area_circle);
 }
}
class abs{
  public static void main(String[] args){
          rectangle rec = new rectangle();
          rec.print_area();
          triangle tri = new triangle();
          tri.print_area();
          circle cir = new circle();
          cir.print_area();
          }
}
```

```
D:\coding files\OOJ Lab>javac 14-abstract_class.java

D:\coding files\OOJ Lab>java abs
The area of rectangle is: 12
The area of triangle is: 6
The area of circle is: 28

D:\coding files\OOJ Lab>
```

/*Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks: • Accept deposit from customer and update the balance.
• Display the balance.
• Compute and deposit interest
• Permit withdrawal and update the balance
• Check for the minimum balance,
impose penalty if necessary and update the balance
*/

```
import java.util.Scanner;
class account
{
   private String name;
   private long account_number;
   private int account_type;
   double balance;
   void set_data()
   {
          Scanner ss=new Scanner(System.in);
          System.out.println("Enter Account Holder Name");
          name = ss.next();
          System.out.println("Enter the account Number");
          account_number=ss.nextLong();
     System.out.println("Choose the account type:\n1.savings account\n2.current account");
          account_type=ss.nextInt();
   }
 void get_data(){
         System.out.println("Account Holder: "+name);
         System.out.println("Account Number: "+account_number);
 }
   int return_account_type()
   {
          return account_type;
   }
```

```
}
class savings extends account
{
   Scanner ss=new Scanner(System.in);
   double amount;
   void get_sav_balance()
   {
         System.out.println("Enter the Amount to be placed in your Savings Account");
         amount=ss.nextDouble();
         balance+=amount;
   }
   void display_sav_blnce()
   {
         System.out.println("balance= "+balance);
   }
   void compute_sav_interest()
   {
        System.out.println("\n***Calculating Compound Interest***");
        System.out.println("Enter annual interest rate: ");
        float rate = ss.nextFloat();
        System.out.println("Enter time in years: ");
        float time = ss.nextFloat();
        System.out.println("Enter principle: ");
        float principle = ss.nextFloat();
        float CI = (float)((principle*(Math.pow((1 + rate / (12*100)),(12*time))))-principle);
        System.out.println("The Compound Interest is: " +CI);
        balance = balance+CI;
        System.out.println("Balance after adding Interest: "+balance);
   }
   void withdrawl_sav()
   {
         System.out.println("Enter the amount to be withdrawn");
         amount = ss.nextDouble();
         balance=balance-amount;
   }
}
class current extends account
{
   Scanner ss = new Scanner(System.in);
   double amount;
   final double min_balance=500;
   void get_cur_balance()
   {
         System.out.println("Enter the amount to be placed in your current account");
```

```
        amount=ss.nextDouble();
        balance+=amount;
    }
    void display_cur_blnce()
    {
        System.out.println("Balance = "+balance);
    }
    void compute_cur_service_charges()
    {
        if(balance<min_balance)
        {
            System.out.println("service tax of rs.100 shall be levied");
            balance=balance-100;
        }
        else
        {
            System.out.println("Minimum balance is Maintained");
        }
    }
    void withdrawl_cur()
    {
        System.out.println("Enter the amount to be withdrawn");
        amount=ss.nextDouble();
        balance=balance-amount;
    }
}
class bank
{
    public static void main(String args[])
    {
        Scanner ss = new Scanner(System.in);
        int type;
        System.out.println("Enter the bank details");
        account acc=new account();
        acc.set_data();
        type=acc.return_account_type();
        if (type==1)
        {
        System.out.println("SAVINGS ACCOUNT");
        acc.get_data();
        savings sav = new savings();
            sav.get_sav_balance();
            sav.display_sav_blnce();
        System.out.println("Do you want to calculate Interest or not:\nIf yes press 1 else 0");
```

```
        int ch = ss.nextInt();
        if (ch == 1)
        {
                sav.compute_sav_interest();
        }
                sav.display_sav_blnce();
                sav.withdrawl_sav();
                sav.display_sav_blnce();
        }
        if(type==2)
        {
        System.out.println("CURRENT ACCOUNT");
        acc.get_data();
                current cur=new current();
                cur.get_cur_balance();
                cur.display_cur_blnce();
        cur.compute_cur_service_charges();
                cur.display_cur_blnce();
                cur.withdrawl_cur();
                cur.display_cur_blnce();
         }
   }
}
```

```
D:\coding files\OOJ Lab>javac l4-bank_acc.java

D:\coding files\OOJ Lab>java bank
Enter the bank details
Enter Account Holder Name
hemang
Enter the account Number
12345
Choose the account type:
1.savings account
2.current account
1
SAVINGS ACCOUNT
Account Holder: hemang
Account Number: 12345
Enter the Amount to be placed in your Savings Account
2000
balance= 2000.0
Do you want to calculate Interest or not:
If yes press 1 else 0
1

***Calculating Compound Interest***
Enter annual interest rate:
10
Enter time in years:
2
Enter principle:
2000
The Compound Interest is: 440.78146
Balance after adding Interest: 2440.781463623047
balance= 2440.781463623047
Enter the amount to be withdrawn
500
balance= 1940.7814636230469
```

```
D:\coding files\OOJ Lab>java bank
Enter the bank details
Enter Account Holder Name
Hemang
Enter the account Number
67890
Choose the account type:
1.savings account
2.current account
2
CURRENT ACCOUNT
Account Holder: Hemang
Account Number: 67890
Enter the amount to be placed in your current account
5000
Balance = 5000.0
Minimum balance is Maintained
Balance = 5000.0
Enter the amount to be withdrawn
300
Balance = 4700.0
```

```
D:\coding files\OOJ Lab>java bank
Enter the bank details
Enter Account Holder Name
Hemang
Enter the account Number
123
Choose the account type:
1.savings account
2.current account
2
CURRENT ACCOUNT
Account Holder: Hemang
Account Number: 123
Enter the amount to be placed in your current account
400
Balance = 400.0
service tax of rs.100 shall be levied
Balance = 300.0
Enter the amount to be withdrawn
150
Balance = 150.0
```