Week 8

1)

```
abstract class Shape{
    int b,h;
    Shape(int b, int h){
        this.b = b;
        this.h = h;
    }
    Void printArea(){};
}

class Rectangle extends Shape{
    Rectangle(int b, int h){
        super(b,h);
    }

    Void printArea(){
        System.out.println("Area of the rectangle is "+
        (b*h));
}
class Triangle extends Shape{
    Triangle(int b,int h){
        super(b,h);
    }
    Void printArea(){ System.out.println("Area of
                        the triangle is " + ((b*h)/2));}
}
```

```
class Circle Extends Shape {
        Circle (int r){
        super(r,r);
    }
    Void printArea() {
      System.out.println("Area of the circle is "
      (Math.PI*h*h));
    };
}

public class Prog1 {
    public static void main (String args[]){
        Rectangle rect = new Rectangle (10, 5);
        Triangle tri = new Triangle (10,5);
        Circle c = new circle (int 10);
        rect.printArea();
        tri. printArea();
        c. print Area();
    }
}
```

2) import java.util.Scanner;

```
class abstract class Account {
    String cName, AccNum, accType;
    Scanner sc = new Scanner(System.in)
    Account (String name, String accNo,
             String acctype) {
        this. cName = name;
        this . accNum = accNo;
        this . accType = accType;
    }
Account () { }
}

class Current Acc extends Account {
    private double balance = 5000, rate=0.06;
    Private boolean withdraw = false;
  Current Acc (String name, string accNo, String acctype)
  {                                                    {
      super (name , accNo, accType);
      System.out.println("* Welcome " + cName);
  }
void getBalance () {
      System . out.format ("Your balance : %.2f%n
                , balance);
}
```

```
void deposit (double amount){
    char choice;
    System.out.println("Deposit. Account holder:"+
      cName + " Amount: " + amount);
    System.out.println("Approve Deposit (Y/N):");
    choice = sc.next().charAt(0);
    if (choice == 'Y' || choice == 'y'){
            balance += amount;
        System.out.println("Deposit approved. Updated
                balance:" + balance);
    } else {
        System.out.println("Deposit not approved");
    }
void withdraw (double amount){
    System.out.println("this account cannot
                withdraw any funds");
}
void checkMinAmount(){
        if (balance < 3000){
            balance -= 500;
          System.out.print("Balance is below level");
          System.out.print("Penalty imposed");
        }
    }
}
```

```
class Savings Acc extends Account {
    private double balance=5000, rate=0.06;
    Savings Acc (String name, String accNo, String
    accType) {
        super(name, accNo, actype);
        System.out.println("Welcome " +cName);
    }
void getBalance (){
    System.out.format(" Your Balance : %.2f\n",
                      balance);
}
void deposit (double amount) {
    char choice;
    System.out.println("Deposit approved
    Deposit : Account holder : "+ cName +" Amount
    +Amount);
    System.out.println("Approve Deposit? (Y/S)! ");
    choice = sc.next().charAt(0);
    if (choice == sc.next().charAt(0); 'Y'|| choice='y'){
        balance += amount;
        System.out.println(" Deposit approved.
        Updated Balance: +balance);
```

```
        cale Interest();
        checkMin amount();
    } else {
        System.out.println(ANSI "Deposit not
                                approved");
void Cale Interest(){
    double CI;
    CI = balance += * (Math.pow((1+rate/100), 2));
    balance += CI;
    System.out.println("Interest Added.");
void withdraw(double Amount){
    char choice;
    if(balance < amount){
        System.out.println("Account Balance
        is lower than Amount to be Withdrawn");
        return;
    }
System.out.println("Approve " + cName + "'s
    request for withdrawal? (Y/N) ");
choice = sc.next().charAt(0);
if( choice == 'Y' || choice == 'y'){
        balance -= amount
```

```
System.out.println("Withdrawal approved.
                    Updated Balance: "+balance);
    calcInterest();
    checkminAmount();
} else {
    System.out.println("withdrawal not
                        approved");
}
}
void checkminAmount(){
    int minamount=3000, penalty=500;
    if (balance<minAmount){
        balance -= penalty;
    }
System.out.println("Bal is under a minimum
    amount to be maintained");
System.out.println("Penalty imposed");
}
}
```

```
public class Bank {
    public static void main (String [] args) {
        int c;
        double temp;
        String name, accNo, accType;
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter Name: ");
        name = nextLine ();
        System.out.println ("Enter Account name:");
        accNo = sc.nextLine()
        System.out.println (" Enter type: ");
        accType= sc.nextline();
    if(accType.charAt(0) == 'c') {
            CurrentAcc a = new CurrentAcc(
                    name, accNo, accType);
    while (true) {
            System.out.println ("1. deposit \n
            2. Withdraw money \n 3. Display Money \n
            4. Exit");
        c= sc.nextInt
```

```
switch(c) {
        case 1: {
            System.out.println("Enter amount
            to be deposited: ");
          temp= sc.nextDouble();
          a.deposit(temp);
          break;
    }
   case 2: { System.out.println("Enter
            amount to be withdrawn: ");
        temp = sc.nextDouble();
        a.withdraw(temp);
        break;
    }
   case 3: {
        a.getBalance();
        break;
    }
   case 4:  {
        System.exit.(0);
        break;
  default: System("Enter correct option");
}
}
```

```
else if(acctype.charAt(0)== 'S'){
    Savings Acc a= new Savings Acc (name, accNo,
                                    accType);

while (true){
        c=sc.nextInt();
switch (c){
   case 1 : {
        System.out.println ("Enter
        amount ");
        temp = sc.next Double ();
        a.deposit (temp);
        break;
}
case 2: {
    System.out.println ("Enter amount
     to be withdrawn :");
    temp= sc.nextDouble();
    a.withdraw(temp);
    break;
}
```

```
case 3: {
        a.getBalance();
        break;
}

case 1: {
        System.exit(0);
        break;
  }
default: System.out.println("Enter the
        correct options");
  }
 }
}else {
      System.out.println("Enter valid type... Exiting");
    }
}
```