**INSTITUTE OF ENGINEERING & MANAGEMENT**

**Department of Computer Science & Engineering**



|  |  |
| --- | --- |
| **Name** | **: Saptarshi Mondal** |
| **Class Roll** | **: 27** |
| **Enrollment No.** | **: 12019002002039** |
| **Subject Name** | **: OOP Lab** |
| **Assignment No.** | **: Day 4** |
| **Date** | **: 25/08/2021** |

# 1. Design a class to represent a Bank Account. Include the following things:

# Fields

# • Name of the depositor

# • Address of the depositor

# • Account number

# • Balance amount in the account

# Methods

# • To assign initial values

# • To deposit an amount

# • To withdraw an amount after checking balance

# • To display the name, address, and balance of a customer

# From main() method create object and call these methods.

# Ans:

import *java*.*util*.*\**;

*class* prog1class2 {

*static* String name, address, anumber;

*static* int balance;

    prog1class2() {

        name = address = anumber = "";

        balance = 0;

    }

    void deposit(int x) {

        balance += x;

    }

    void withdraw(int x) {

        if (balance >= x)

            balance = balance - x;

        else

            System.*out*.println(x + " is bigger than " + balance);

    }

    void display() {

        System.*out*.println("The name is " + name);

        System.*out*.println("The address is " + address);

        System.*out*.println("The account number is " + anumber);

        System.*out*.println("The current balance is " + balance);

    }

}

*class* prog1 {

*static* Scanner sc = new Scanner(System.*in*);

*public* *static* void main(String[] args) {

        prog1class2 obj1 = new prog1class2();

        System.*out*.println("Enter the name, address , account number and account balance in this order ");

        obj1.*name* = sc.nextLine();

        obj1.*address* = sc.nextLine();

        obj1.*anumber* = sc.nextLine();

        int bal = sc.nextInt();

        obj1.deposit(bal);

        System.*out*.println("Enter the amount to be withdrawn from the account ");

        int w = sc.nextInt();

        obj1.withdraw(w);

        obj1.display();

    }

}

# Output:

# 

# 2. Create a class and determine if method overloading holds good for return type of methods

# or not.

# Ans:

*class* Addition {

*public* int add(int a, int b) {

        int sum = a + b;

        return sum;

    }

*public* float add(int a, int b) {

        int sum = a + b;

        return sum;

    }

}

*class* prog2 {

*public* *static* void main(String[] args) {

        Addition ob = new Addition();

        int sum1 = ob.add(1, 2);

        System.*out*.println("sum of the two integer value :" + sum1);

        float sum2 = ob.add(1, 2);

        System.*out*.println("sum of the three integer value :" + sum2);

    }

}

# Output:

# 

# 3. Overload the constructors for classes Area and Volume of a rectangular figure and display

# its area and volume. Area is the superclass and Volume is the subclass.

# Ans:

*class* Area {

    double length, breadth, area;

    Area(double w, double h) {

        length = w;

        breadth = h;

    }

    Area(double len) {

        length = breadth = len;

    }

    Area() {

        length = breadth = 0;

    }

    double areacalc() {

        return length \* breadth;

    }

}

*class* Volume *extends* Area {

    double vol, height;

    Volume(double l, double b, double h) {

        length = l;

        breadth = b;

        height = h;

    }

    Volume() {

        height = 0;

    }

    double volcalc() {

        return areacalc() \* height;

    }

}

*public* *class* prog3 {

*public* *static* void main(String args[]) {

        Volume mybox1 = new Volume(10, 20, 15);

        double vol, areax;

        vol = mybox1.volcalc();

        areax = mybox1.areacalc();

        System.*out*.println(" Volume of mybox1 is " + vol);

        System.*out*.println(" Area of mybox1 is " + areax);

    }

}

# Output:

# 