# DAY-2

NAME:B.S.shruthika

REG NO:192110032

1.Write a java program to find the Matrix Addition

Program:

public class MatrixAddition {

public static void main(String[] args) {

int[][] matrix1={{2,3,4},{2,4,3},{3,4,5}};

int[][] matrix2={{2,3,4},{2,4,3},{1,2,4}};

int rows=matrix1.length;

int cols=matrix1[0].length;

int[][] resultMatrix=new int[rows][cols];

for (int i=0;i<rows;i++) {

for (int j=0;j<cols;j++) {

resultMatrix[i][j]=matrix1[i][j] +

matrix2[i][j];

}

}

System.out.println("Resultant Matrix:");

for (int i=0;i<rows;i++) {

for (int j=0;j<cols;j++) {

System.out.print(resultMatrix[i][j]+" ");

}

System.out.println();

       }

    }

}

Output:

Resultant Matrix:

4 6 8

4 8 6

4 6 9

2. Write a java program to find the Matrix Sub

Program:

public class MatrixAddition {

public static void main(String[] args) {

int[][] matrix1={{2,4,5},{2,4,3},{3,4,5}};

int[][] matrix2={{1,3,4},{2,4,3},{1,2,4}};

int rows=matrix1.length;

int cols=matrix1[0].length;

int[][] resultMatrix=new int[rows][cols];

for (int i=0;i<rows;i++) {

for (int j=0;j<cols;j++) {

resultMatrix[i][j]=matrix1[i][j] -

matrix2[i][j];

}

}

System.out.println("Resultant Matrix:");

for (int i=0;i<rows;i++) {

for (int j=0;j<cols;j++) {

System.out.print(resultMatrix[i][j]+" ");

}

System.out.println();

       }

    }

}

Output:

Resultant Matrix:

1 1 1

0 0 0

2 2 1

3. Write a java program to print the Area of Rectangle.

Program:

class Rectangle{

int height,width;

void area(){

int result=height\*width;

System.out.println("Area of rectangle is = "+result);

}

}

class Rect {

public static void main(String[] args){

Rectangle obj=new Rectangle();

obj.height=6;

obj.width=10;

obj.area();

}

}

Output:

Area of rectangle is = 60

4. Write the java program to print the Area of rectangle using constructor .

Program:

import java.util.\*;

class Rectangle{

int height,width;

public Rectangle(){

Scanner s =new Scanner(System.in);

System.out.println("Enter a height");

height=s.nextInt();

System.out.println("Enter a width");

width=s.nextInt();

}

void cal()

{

int result=height\*width;

System.out.println("Area of rectangle is = "+result);

}

public static void main(String[] args){

Rectangle obj=new Rectangle();

obj.cal();

}

}

Output:

Enter a height 9

Enter a width 4

Area of rectangle is = 36

5.Write a java program for sum of series in oops.

Program:

import java.util.Scanner;

class SeriesCalculator {

public int n; // Number of terms in the series

public SeriesCalculator(int n) {

this.n = n;

}

public int calculateSum() {

int sum = 0;

for (int i = 1; i <= n; i++) {

// Replace this line with the formula for your series

sum += i; // Example: Sum of first 'n' natural numbers

}

return sum;

}

}

class SumOfSeries {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of terms in the series: ");

int n = scanner.nextInt();

SeriesCalculator series = new SeriesCalculator(n);

int sum = series.calculateSum();

System.out.println("Sum of the series: " + sum);

scanner.close();

}

}

Output:

Enter the number of terms in the series: 10

Sum of the series: 55

6. Write a java program for area of circle.

Program:

import java.util.\*;

class cicle{

float r,pi;

Scanner s=new Scanner(System.in);

System.out.println("Enter radious of circle");

r=s.nextFloat();

System.out.println("Enter the pi vales");

pi=s.nextFloat();

void area()

}

void cal()

{

float result=pi\*r\*r;

System.out.println("Area of circle is="+"result");

}

}

class circle

{

public static void main(string args())

{

cicle obj=new circle();

obj area();

obj cal();

}

}

Output:

Enter radious of circle: 4

Enter the pi vales:3.14

Area of circle is= 50.24

7. Write a java program for simple interest.

Program:

import java.util.\*;

class SimpleInterest{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the principal amount: ");

double principal = scanner.nextDouble();

System.out.print("Enter the rate of interest: ");

double rate = scanner.nextDouble();

System.out.print("Enter the time (in years): ");

double time = scanner.nextDouble();

double simpleInterest = (principal \* rate \* time) / 100;

System.out.println("Simple Interest: " + simpleInterest);

}

void cal()

{

float result=p\*t\*r/100;

System.out.println("simpleinterest is="+"result");

}

class simple

{

public static void main(string args())

{

SimpleInterest obj=new SimpleInterest();

obj simple();

obj cal();

}

}

Output:

Enter the principal amount:10000

Enter the rate of interest:5

Enter the time (in years):2

simpleinterest is=1000

8.Write a java program for matrix mul .

Program:

mport java.util.Scanner;

class matrixmultiplication

{

public static void main(String args[]){

int row1, col1, row2, col2;

Scanner s = new Scanner(System.in);

System.out.print("Enter number of rows in first matrix:");

row1 = s.nextInt();

System.out.print("Enter number of columns in first matrix:");

col1 = s.nextInt();

System.out.print("Enter number of rows in second matrix:");

row2 = s.nextInt();

System.out.print("Enter number of columns in second matrix:");

col2 = s.nextInt();

if (col1 != row2) {

System.out.println("Matrix multiplication is not possible");

}

else {

int a[][] = new int[row1][col1];

int b[][] = new int[row2][col2];

int c[][] = new int[row1][col2];

System.out.println("Enter values for matrix A : \n");

for (int i = 0; i < row1; i++) {

for (int j = 0; j < col1; j++)

a[i][j] = s.nextInt();

}

System.out.println("Enter values for matrix B : \n");

for (int i = 0; i < row2; i++) {

for (int j = 0; j < col2; j++)

b[i][j] = s.nextInt();

}

System.out.println("Matrix multiplication is : \n");

for(int i = 0; i < row1; i++) {

for(int j = 0; j < col2; j++){

c[i][j]=0;

for(int k = 0; k < col1; k++){

c[i][j] += a[i][k] \* b[k][j];

}

System.out.print(c[i][j] + " ");

}

System.out.println();

}

}

}

}

Output:

Enter number of rows in first matrix:2Enter number of columns in first matrix:2Enter number of rows in second matrix:2Enter number of columns in second matrix:2Enter values for matrix A : 1 25 3Enter values for matrix B : 2 34 1Matrix multiplication is : 10 5 22 18

9. Write a java program for area of triangle.

Program:

import java.util.Scanner;

class Triangle {

public double base;

public double height;

public Triangle(double base, double height) {

this.base = base;

this.height = height;

}

public double calculateArea() {

return 0.5 \* base \* height;

}

}

public class TriangleAreaCalculator {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the base of the triangle: ");

double base = scanner.nextDouble();

System.out.print("Enter the height of the triangle: ");

double height = scanner.nextDouble();

Triangle triangle = new Triangle(base, height);

double area = triangle.calculateArea();

System.out.println("The area of the triangle is: " + area);

scanner.close();

}

}

Output:

Enter the base of the triangle:10

Enter the height of the triangle:20

The area of the triangle is:100

10. Write a java program for argument constructor.

Program:

Class Box

{

double width, heighth, depth; Box (double w, double h, double d) || alguments Constructor

{

width=W;

height=h;

depth =d,

}

double volume ()

{

return width \*height \* depth;

}

public static void main ( String[ ] args)

{

BOX B= new BOX [4, 5, 20, 5, 10, 4)

double res3=63.volume();"

System.out.println (" result 3 :"+res3) ;

}

}

Output:

4, 5, 20, 5, 10, 4

11. Implement a class Account. An account has

• a balance

• functions to add

• and withdraw money,

• and a function to inquire about the current balance.

Condition:

1. Pass a value into a constructor to set an initial balance.

2. If no value is passed the initial balance should be set to $0.

3. Charge a $5 penalty if an attempt is made to withdraw more money than is available in the account.

4. Enhance the Account class to compute interest on the current balance.

Program:

public class Account {

private double balance;

private double interestRate;

public Account() {

this.balance = 0.0;

this.interestRate = 0.02; // Default interest rate of 2%

}

public Account(double initialBalance) {

if (initialBalance < 0) {

System.out.println("Initial balance cannot be negative. Setting balance to $0.");

this.balance = 0.0;

} else {

this.balance = initialBalance;

}

this.interestRate = 0.02; // Default interest rate of 2%

}

public void deposit(double amount) {

if (amount > 0) {

balance += amount;

System.out.println("Deposited $" + amount);

} else {

System.out.println("Invalid deposit amount. Please deposit a positive amount.");

}

}

public void withdraw(double amount) {

if (amount > 0) {

if (balance >= amount) {

balance -= amount;

System.out.println("Withdrawn $" + amount);

} else {

System.out.println("Insufficient balance. A $5 penalty will be charged.");

balance -= 5.0; // Apply $5 penalty

}

} else {

System.out.println("Invalid withdrawal amount. Please withdraw a positive amount.");

}

}

public double getBalance() {

return balance;

}

public void addInterest() {

double interest = balance \* interestRate;

balance += interest;

System.out.println("Added interest of $" + interest);

}

public static void main(String[] args) {

Account account1 = new Account(1000); // Creating an account with an initial balance of $1000

Account account2 = new Account(); // Creating an account with an initial balance of $0

account1.deposit(500);

account2.deposit(200);

account1.withdraw(300);

account2.withdraw(50);

account1.addInterest();

account2.addInterest();

System.out.println("Account 1 balance: $" + account1.getBalance());

System.out.println("Account 2 balance: $" + account2.getBalance());

}

}

Output:

Deposited $500.0

Deposited $200.0

Withdrawn $300.0

Withdrawn $50.0

Added interest of $24.0

Added interest of $3.0Account 1 balance: $1224.0

Account 2 balance: $153.0

12. Write a class called Triangle that can be used to represent a triangle. It should include the following methods that return Boolean values indicating if the particular property holds:

• isRight (a right triangle)

• isScalene (no two sides are the same length)

• isIsosceles (exactly two sides are the same length)

• isEquilateral (all three sides are the same length)

Program:

public class Triangle {

private double side1;

private double side2;

private double side3;

public Triangle(double side1, double side2, double side3) {

this.side1 = side1;

this.side2 = side2;

this.side3 = side3;

}

public boolean isRight() {

return Math.pow(side1, 2) + Math.pow(side2, 2) == Math.pow(side3, 2)

|| Math.pow(side1, 2) + Math.pow(side3, 2) == Math.pow(side2, 2)

|| Math.pow(side2, 2) + Math.pow(side3, 2) == Math.pow(side1, 2);

}

public boolean isScalene() {

return !isIsosceles();

}

public boolean isIsosceles() {

return side1 == side2 || side1 == side3 || side2 == side3;

}

public boolean isEquilateral() {

return side1 == side2 && side1 == side3;

}

public static void main(String[] args) {

Triangle triangle1 = new Triangle(3, 4, 5);

Triangle triangle2 = new Triangle(2, 2, 3);

Triangle triangle3 = new Triangle(1, 1, 1);

System.out.println("Triangle 1 is a right triangle: " + triangle1.isRight());

System.out.println("Triangle 1 is scalene: " + triangle1.isScalene());

System.out.println("Triangle 1 is isosceles: " + triangle1.isIsosceles());

System.out.println("Triangle 1 is equilateral: " + triangle1.isEquilateral());

System.out.println("Triangle 2 is a right triangle: " + triangle2.isRight());

System.out.println("Triangle 2 is scalene: " + triangle2.isScalene());

System.out.println("Triangle 2 is isosceles: " + triangle2.isIsosceles());

System.out.println("Triangle 2 is equilateral: " + triangle2.isEquilateral());

System.out.println("Triangle 3 is a right triangle: " + triangle3.isRight());

System.out.println("Triangle 3 is scalene: " + triangle3.isScalene());

System.out.println("Triangle 3 is isosceles: " + triangle3.isIsosceles());

System.out.println("Triangle 3 is equilateral: " + triangle3.isEquilateral());

}

}

Output:

Triangle 1 is a right triangle: trueTriangle 1 is scalene: true

Triangle 1 is isosceles: false

Triangle 1 is equilateral: false

Triangle 2 is a right triangle: false

Triangle 2 is scalene: false

Triangle 2 is isosceles: true

Triangle 2 is equilateral: false

Triangle 3 is a right triangle: false

Triangle 3 is scalene: false

Triangle 3 is isosceles: true

Triangle 3 is equilateral: true