**CYCLE 1**

**1. Define a class ‘product’ with data members pcode, pname and price. Create 3 objects of the class and find the product having the lowest price.**

import java.util.\*;

class Product{

int pcode;

String pname;

double price;

void assign(){

Scanner s=new Scanner(System.in);

System.out.println("\nEnter the product code");

pcode=s.nextInt();

System.out.println("\nEnter the product name");

pname=s.next();

System.out.println("\nEnter the product price");

price=s.nextInt();

}

static void lowest(double p1, double p2, double p3){

if(p1<p2 && p1<p3){

System.out.println("product 1 with cost "+ p1 + " is of the lowest price");

}

else if(p2<p1 && p2<p3){

System.out.println("product 2 with cost "+ p2 + " is of the lowest price");

}

else{

System.out.println("product 3 with cost "+ p3 + " is of the lowest price");

}

}

public static void main(String[] args){

System.out.println("J SIDDHARTH");

System.out.println("SJC22MCA-2034");

System.out.println("24-03-2023");

System.out.println("20MCA132 , Object Oriented Programming Lab \n");

Product obj1 = new Product();

Product obj2 = new Product();

Product obj3 = new Product();

System.out.println("Enter the first product details");

obj1.assign();

System.out.println("Enter the second product details");

obj2.assign();

System.out.println("Enter the third product details");

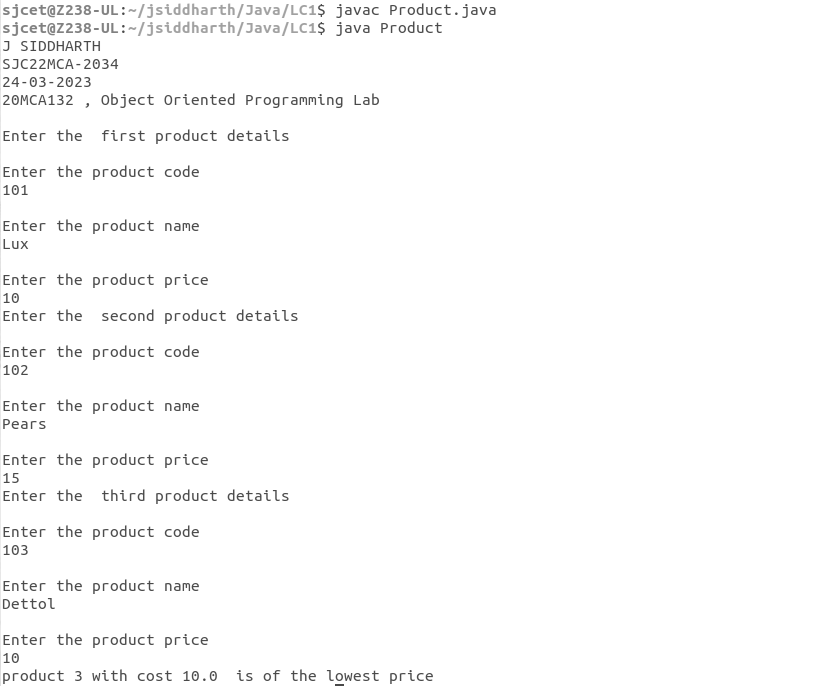
obj3.assign();

lowest(obj1.price,obj2.price,obj3.price);

}

}

**OUTPUT**



**2. Read 2 matrices from the console and perform matrix addition.**

import java.util.Scanner;

public class Matrixaddition{

public static void main(String args[]){

System.out.println("J SIDDHARTH");

System.out.println("SJC22MCA-2034");

System.out.println("28-03-2023");

System.out.println("20MCA132 , Object Oriented Programming Lab \n");

int row, col,i,j;

Scanner in = new Scanner(System.in);

System.out.println("Enter the number of rows");

row = in.nextInt();

System.out.println("Enter the number columns");

col = in.nextInt();

int mat1[][] = new int[row][col];

int mat2[][] = new int[row][col];

int res[][] = new int[row][col];

System.out.println("Enter the elements of matrix 1");

for ( i= 0 ; i < row ; i++ )

{

for ( j= 0 ; j < col ;j++ )

mat1[i][j] = in.nextInt();

System.out.println();

}

System.out.println("Enter the elements of matrix 2");

for ( i= 0 ; i < row ; i++ )

{

for ( j= 0 ; j < col ;j++ )

mat2[i][j] = in.nextInt();

System.out.println();

}

for ( i= 0 ; i < row ; i++ )

for ( j= 0 ; j < col ;j++ )

res[i][j] = mat1[i][j] + mat2[i][j] ;

System.out.println("Sum of matrices:-");

for ( i= 0 ; i < row ; i++ )

{

for ( j= 0 ; j < col ;j++ )

System.out.print(res[i][j]+"\t");

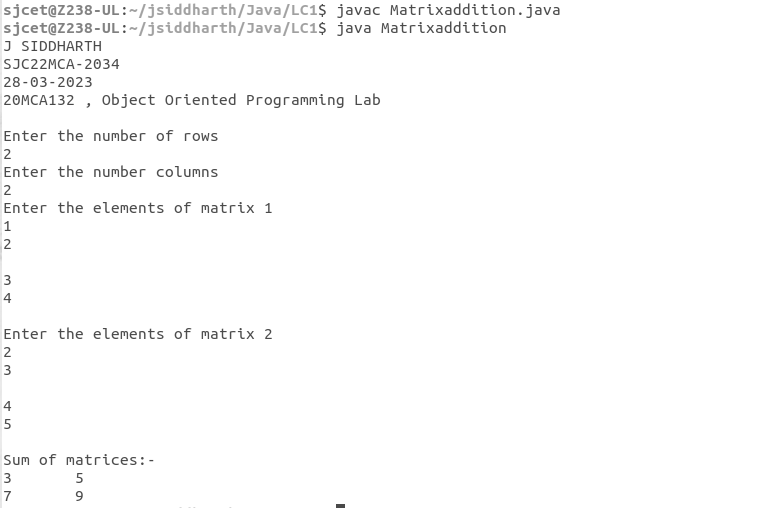
System.out.println();

}

}

}

**OUTPUT**



**3. Add complex numbers**

class Complex {

double real, img;

Complex(double r, double i) {

this.real = r;

this.img = i;

}

public static Complex sum(Complex c1, Complex c2) {

Complex temp = new Complex(0, 0);

temp.real = c1.real + c2.real;

temp.img =c1.img + c2.img;

return temp;

}

public static void main(String args[]) {

System.out.println("J SIDDHARTH");

System.out.println("SJC22MCA-2034");

System.out.println("04-04-2023");

System.out.println("20MCA132 , Object Oriented Programming Lab \n");

Complex c1 = new Complex(5.5, 4);

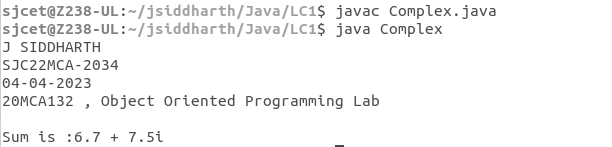
Complex c2 = new Complex(1.2, 3.5);

Complex temp = sum(c1, c2);

System.out.println("Sum is :" + temp.real+" + "+ temp.img + "i");

}}

**OUTPUT**



**4. Read a matrix from the console and check whether it is symmetric or not.**

**import java.util.Scanner;**

class Symmetric {

public static void main(String[] args) {

System.out.println("J SIDDHARTH");

System.out.println("SJC22MCA-2034");

System.out.println("12-04-2023");

System.out.println("20MCA132 , Object Oriented Programming Lab \n"); Scanner sc = new Scanner(System.in);

System.out.println("enter the number of rows : ");

int rows = sc.nextInt();

System.out.println("enter the number of columns : ");

int cols = sc.nextInt();

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

System.out.println("Enter the elements : ");

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

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

matrix[i][j] = sc.nextInt();

sc.close();

System.out.println("Printing input matrix : ");

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

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

System.out.print(matrix[i][j]+"\t");

System.out.println();

}

if (rows != cols)

System.out.println("The given matrix is not a square matix");

else {

boolean Symmetric = true;

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

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

if(matrix[i][j] != matrix[j][i]) {

Symmetric = false;

break;

}

if(Symmetric) {

System.out.println("The given matrix is symmetric..");

}

else {

System.out.println("The given matrix is not symmetric..");

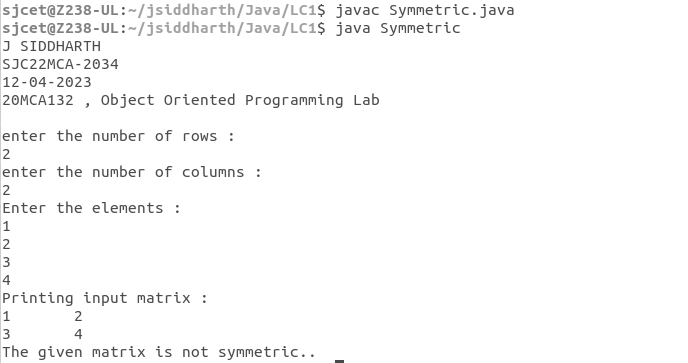
}

}

}

}

**OUTPUT**



**5. Create CPU with attribute price. Create inner class Processor (no. of cores, manufacturer) and static nested class RAM (memory, manufacturer). Create an object of CPU and print information of Processor and RAM.**

class Cpu

{

int price;

Cpu(int p)

{

this.price = p;

}

class Processor

{

int cores;

String manufacture;

Processor(int n, String m) {

this.cores = n;

this.manufacture = m;

}

void display()

{

System.out.println("No of Cores : " + this.cores);

System.out.println("Processor manufactures : " + this.manufacture);

}

}

static class Ram {

int memory;

String manufacture;

Ram(int n, String m) {

this.memory = n;

this.manufacture = m;

}

void display() {

System.out.println("Memory Size : " + this.memory);

System.out.println("Memory manufactures : " + this.manufacture);

}

}

void display() {

System.out.println("Price of CPU : " + this.price);

}

public static void main(String[] args) {

System.out.println("J SIDDHARTH");

System.out.println("SJC22MCA-2034");

System.out.println("24-04-2023");

System.out.println("20MCA132 , Object Oriented Programming Lab \n");

Cpu intel = new Cpu(23000);

Cpu.Processor i\_processor = intel.new Processor(4, "intel");

Cpu.Ram i\_ram = new Ram(1024, "Asus");

intel.display();

i\_processor.display();

i\_ram.display();

}

}

**OUTPUT**