**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 1**

**Aim**

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

**Procedure**

class Product{

String pcode, pname;

double price;

void details(){

System.out.println("PRODUCT DETAILS");

System.out.println("PCode : "+pcode);

System.out.println("PName : "+pname);

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

}

}

public class ProductDetails{

public static void main(String args[]){

Product p1 = new Product();

p1.pcode = "M200J9PI";

p1.pname = "POCO M2";

p1.price = 10999;

System.out.println("\nProduct 1:-");

p1.details();

Product p2 = new Product();

p2.pcode = "XMSH05HM";

p2.pname = "Mi Band 3";

p2.price = 1799;

System.out.println("\nProduct 2:-");

p2.details();

Product p3 = new Product();

p3.pcode = "EPSP5248";

p3.pname = "Camlin Scale";

p3.price = 5;

System.out.println("\nProduct 3:-");

p3.details();

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

System.out.println("\n\nProduct with lowest price is :");

p1.details();

}

else if(p2.price < p3.price){

System.out.println("\n\nProduct with lowest price is :");

p2.details();

}

else{

System.out.println("\n\nProduct with lowest price is :");

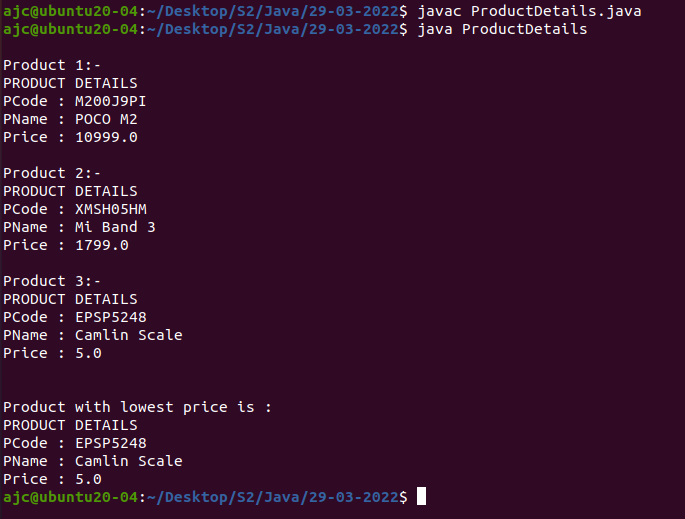
p3.details();

}

}

}

**Output**



**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 2**

**Aim**

Read 2 matrices from the console and perform matrix addition.

**Procedure**

import java.util.\*;

class MatrixAddition{

public static void main(String args[]){

int row, col;

Scanner sc= new Scanner(System.in);

System.out.print("Enter the number of rows for the Matrices : ");

row= sc.nextInt();

System.out.print("Enter the number of columns for the Matrices : ");

col= sc.nextInt();

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

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

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

System.out.println("Enter the "+row+" elements for the Matrix A : ");

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

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

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

}

}

System.out.println("\n");

System.out.println("Enter the "+col+" elements for the Matrix B : ");

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

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

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

}

}

System.out.println("\n");

System.out.println("Matrix A is : ");

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

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

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

}

System.out.println("\n");

}

System.out.println("Matrix B is : ");

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

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

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

}

System.out.println("\n");

}

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

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

matrixSum[i][j]= matrixA[i][j] + matrixB[i][j];

}

}

System.out.println("MatrixSum is : ");

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

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

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

}

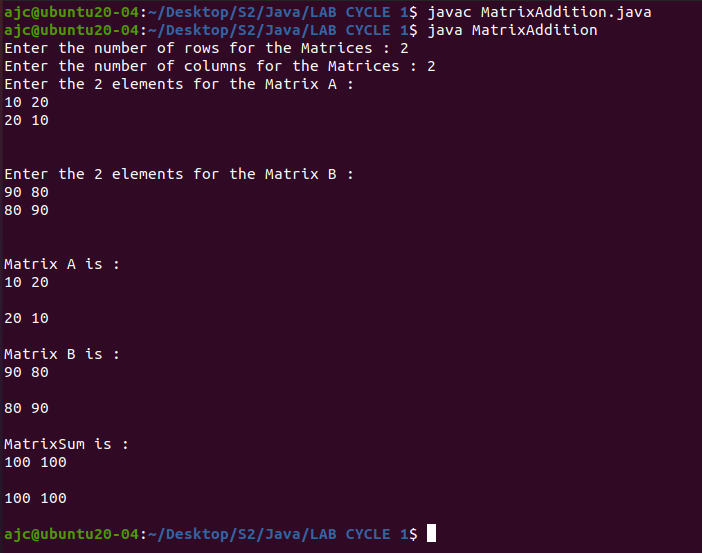
System.out.println("\n");

}

}

}

**Output**



**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 3**

**Aim**

Java program to perform Addition of complex numbers.

**Procedure**

import java.util.\*;

class ComplexNumbers{

int real, imaginary;

ComplexNumbers(){ }

ComplexNumbers(int real, int imaginary){

this.real= real;

this.imaginary= imaginary;

}

void complexAdd(ComplexNumbers compNum){

int real\_sum, imaginary\_sum;

real\_sum= this.real+compNum.real;

imaginary\_sum= this.imaginary+compNum.imaginary;

System.out.println("The sum of the complex numbers are : "+real\_sum+" + "+imaginary\_sum+"i");

}

void display(){

System.out.println("The entered complex number is : "+real+" + "+imaginary+"i");

System.out.println("\n");

}

public static void main(String[] args){

int real\_num, imaginary\_num;

Scanner sc= new Scanner(System.in);

System.out.print("Enter the real value of the 1st complex number : ");

real\_num= sc.nextInt();

System.out.print("Enter the imaginary value of the 1st complex number : ");

imaginary\_num= sc.nextInt();

ComplexNumbers com1= new ComplexNumbers(real\_num, imaginary\_num);

com1.display();

System.out.print("Enter the real value of the 2nd complex number : ");

real\_num= sc.nextInt();

System.out.print("Enter the imaginary value of the 2nd complex number : ");

imaginary\_num= sc.nextInt();

ComplexNumbers com2= new ComplexNumbers(real\_num, imaginary\_num);

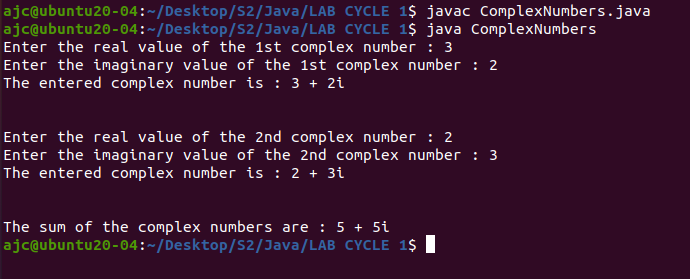
com2.display();

com1.complexAdd(com2);

}

}

**Output**



**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 4**

**Aim**

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

**Procedure**

import java.util.\*;

class SymmetricMatrix{

public static void main(String args[]){

int row, col;

Scanner sc= new Scanner(System.in);

boolean isSymmetic= true;

System.out.print("Enter the number of rows for the Matrices : ");

row= sc.nextInt();

System.out.print("Enter the number of columns for the Matrices : ");

col= sc.nextInt();

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

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

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

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

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

}

}

System.out.println("\n");

System.out.println("The entered matrix is : ");

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

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

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

}

System.out.println("\n");

}

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

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

if(i!=j){

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

isSymmetic= false;

break;

}

}

}

if(!isSymmetic)

break;

}

if(isSymmetic){

System.out.println("The entered matrix is Symmetric Matrix");

}

else{

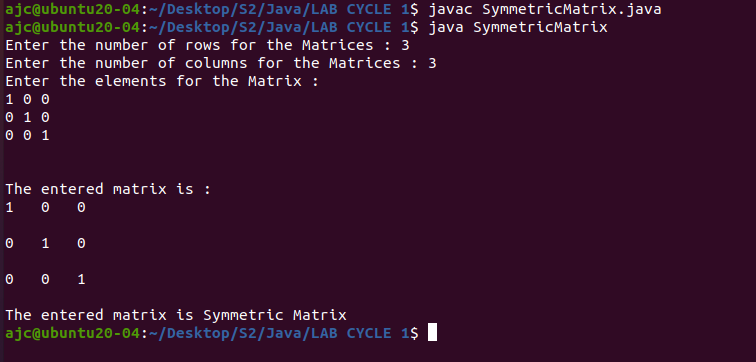
System.out.println("The entered matrix is not a Symmetric Matrix");

}

}

}

**Output**



**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 5**

**Aim**

**Procedure**

**Output**

**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 6**

**Aim**

**Procedure**

**Output**

**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 7**

**Aim**

**Procedure**

**Output**

**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 8**

**Aim**

**Procedure**

**Output**

**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 9**

**Aim**

**Procedure**

**Output**

**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 10**

**Aim**

**Procedure**

**Output**

**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 11**

**Aim**

**Procedure**

**Output**

**Name: SHAMJAD MAZOOD NAZER**

**Roll No: 36**

**Batch: B**

**Date: 29/03/2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 12**

**Aim**

**Procedure**

**Output**