**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**.

**Code :**

import java.util.Scanner;

class ProductClass{

String pname;

int pcode;

int price;

public ProductClass(String name,int code,int price){

this.pname = name;

this.pcode = code;

this.price = price;

}

}

class Product {

public static void main(String[] arguments){

System.out.println("\nName : Sreyas Satheesh\nRoll.no : 53\nTitle : Product Class\nDate : 13/02/2024\n");

Scanner scan = new Scanner(System.in);

System.out.print("\nEnter the code of the first product : ");

int code1 = scan.nextInt();

System.out.print("Enter the name of the first product : ");

scan.nextLine();

String name1 = scan.nextLine();

System.out.print("Enter the price of the first product : ");

int price1 = scan.nextInt();

System.out.print("\nEnter the code of the second product : ");

int code2 = scan.nextInt();

System.out.print("Enter the name of the second product : ");

scan.nextLine();

String name2 = scan.nextLine();

System.out.print("Enter the price of the second product : ");

int price2 = scan.nextInt();

System.out.print("\nEnter the code of the third product : ");

int code3 = scan.nextInt();

System.out.print("Enter the name of the third product : ");

scan.nextLine();

String name3 = scan.nextLine();

System.out.print("Enter the price of the third product : ");

int price3 = scan.nextInt();

ProductClass obj1 = new ProductClass(name1, code1, price1);

ProductClass obj2 = new ProductClass(name2, code2, price2);

ProductClass obj3 = new ProductClass(name3, code3, price3);

ProductClass minimum = obj1.price < obj2.price ? obj1 : obj2;

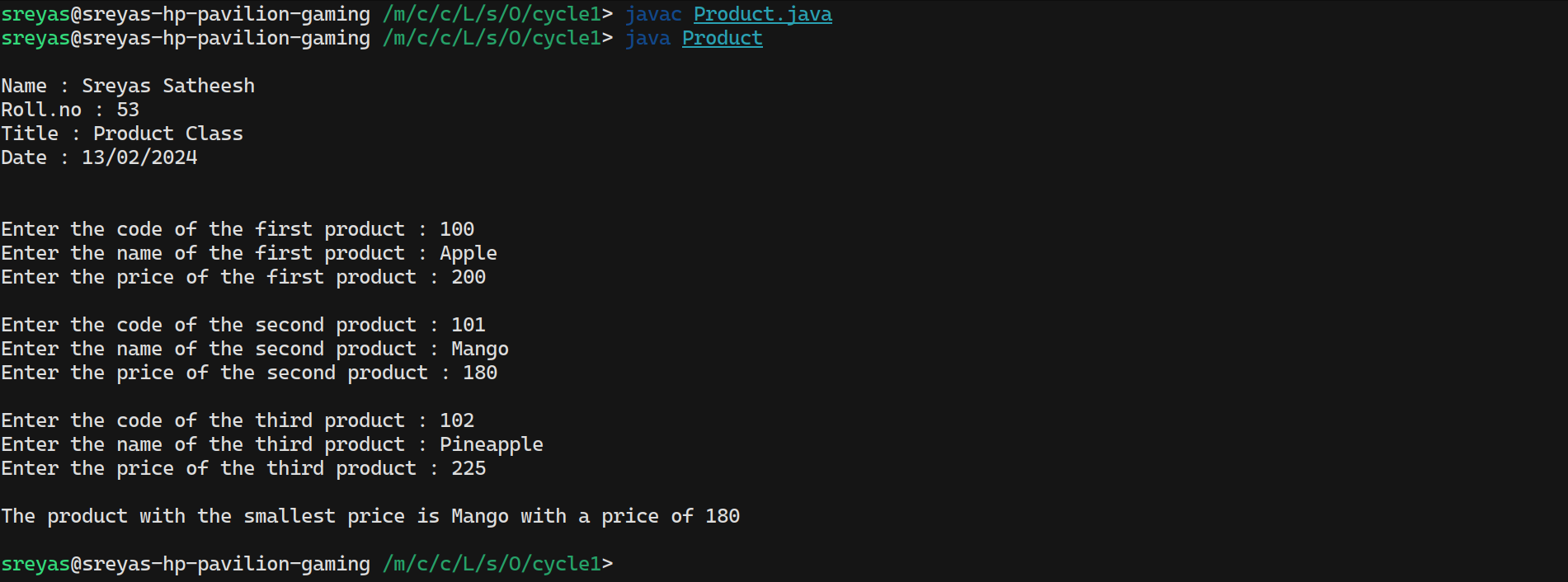
minimum = minimum.price < obj3.price ? minimum : obj3;

System.out.println("\nThe product with the smallest price is " + minimum.pname + " with a price of " + minimum.price + "\n");

}

}

**Output :**

****

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

**Code :**

import java.util.Scanner;

public class Matrix{

public static void main(String[] args){

System.out.println("\nName : Sreyas Satheesh\nRoll.no : 53\nTitle : Matrix addition\nDate : 13/02/2024\n");

Scanner read = new Scanner(System.in);

System.out.println("Enter the rows and cols in the first matrix : ");

int row1 = read.nextInt();

int col1 = read.nextInt();

int[][] mat1 = new int[row1][col1];

System.out.println("Enter the first matrix");

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

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

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

}

}

System.out.println("Enter the rows and cols in the second matrix : ");

int row2 = read.nextInt();

int col2 = read.nextInt();

int[][] mat2 = new int[row2][col2];

System.out.println("Enter the second matrix");

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

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

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

}

}

System.out.println("first matrix is \n");

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

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

System.out.print(mat1[i][j] + "\t");

}

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

}

System.out.println("second matrix is \n");

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

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

System.out.print(mat2[i][j] + "\t");

}

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

}

// Adding matrices

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

if(row1 == row2 && col1 == col2){

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

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

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

}

}

System.out.println("added matrix is \n");

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

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

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

}

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

}

} else {

System.out.println("Addition not possible.");

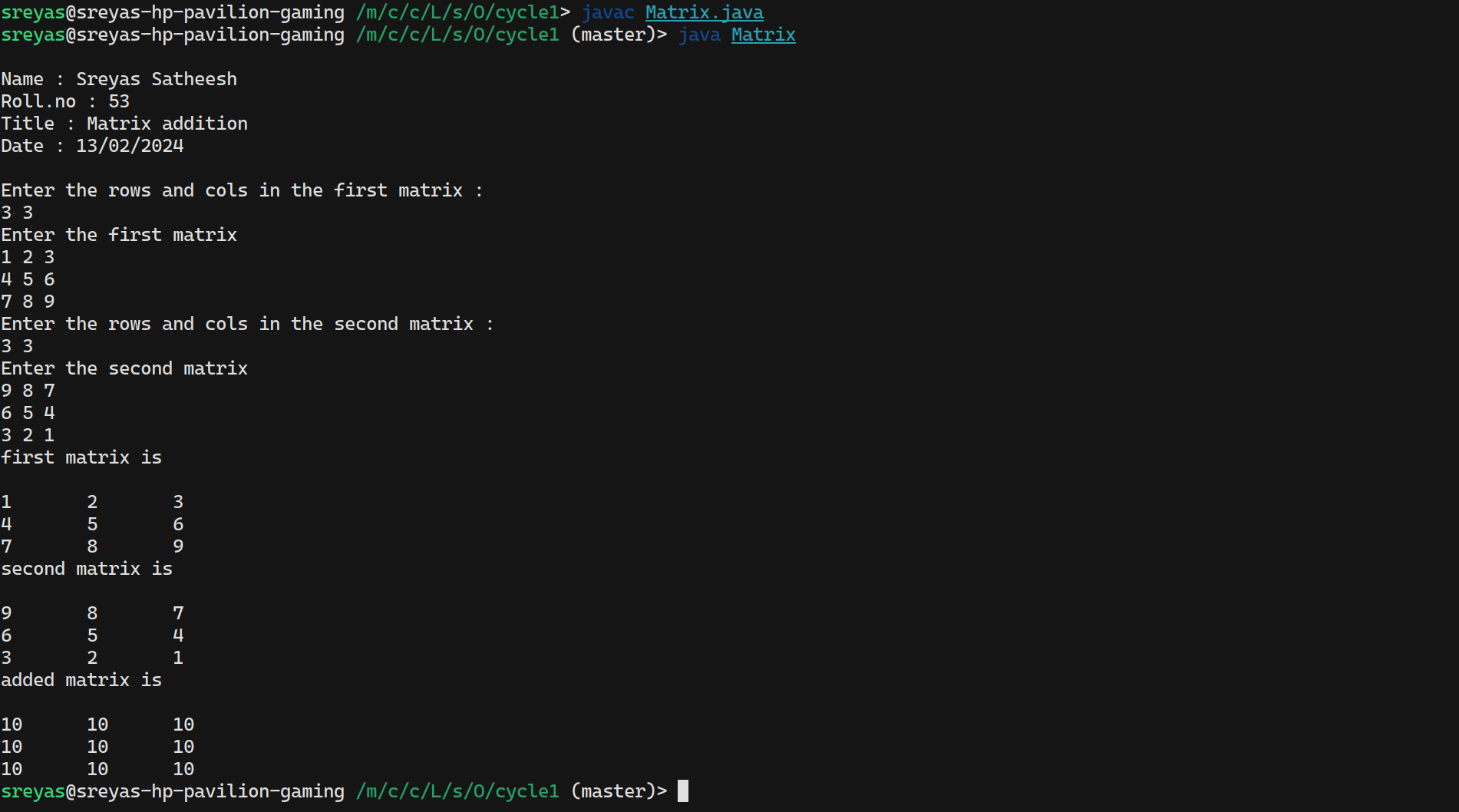
}

read.close();

}

}

**Output :**



**3. Add complex numbers**

**Code :**

import java.util.Scanner;

public class ComplexNumber{

public static void main(String[] arg){

System.out.println("\nName : Sreyas Satheesh\nRoll.no : 53\nTitle : Addition of complex numbers numbers\nDate : 13/02/2024\n");

class ComplexNumberClass{

int real;

int img;

public ComplexNumberClass(int r, int i) {

this.real = r;

this.img = i;

}

}

int a, b;

Scanner scan = new Scanner(System.in);

System.out.println("Enter the first complex number");

System.out.print("Real part : ");

a=scan.nextInt();

System.out.print("Imaginary part : ");

b=scan.nextInt();

ComplexNumberClass first = new ComplexNumberClass(a, b);

System.out.println("Enter the second complex number");

System.out.print("Real part : ");

a=scan.nextInt();

System.out.print("Imaginary part : ");

b=scan.nextInt();

ComplexNumberClass second = new ComplexNumberClass(a, b);

int real = first.real + second.real;

int img = first.img + second.img;

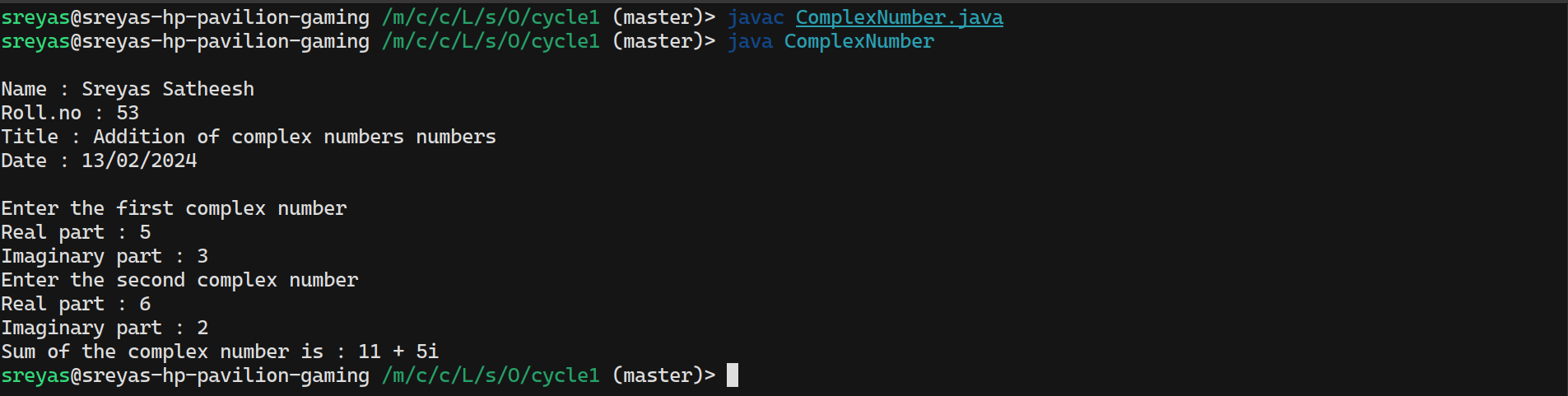
System.out.println("Sum of the complex number is : " + real + " + " + img + "i");

scan.close();

}

}

**Output :**



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

**Code :**

import java.util.Scanner;

public class SymmetricMatrix{

public static void main(String[] arg){

System.out.println("Name : Sreyas Satheesh\nRoll.no : 53\nTitle : Check for symmetric matrices\nDate : 13/02/2024\n");

Scanner read = new Scanner(System.in);

System.out.println("Enter the rows and cols of the matrix : ");

int rows = read.nextInt();

int columns = read.nextInt();

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

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

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

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

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

}

}

System.out.println("\nMatrix is");

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

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

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

}

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

}

if(rows == columns){

int flag=0;

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

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

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

flag=1;

break;

}

}

if(flag == 1) break;

}

if(flag == 0){

System.out.println("\nThe matrix is a Symmetric matrix.");

} else {

System.out.println("\nThe matrix is not a Symmetric matrix.");

}

} else {

System.out.println("\nThe matrix is not a Symmetric matrix.");

}

read.close();

}

}

**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.**

**Code :**

import java.util.Scanner;

class Cpu {

private int price;

public Cpu(int price) {

this.price = price;

}

public void printCPUInfo() {

Scanner scanner = new Scanner(System.in);

System.out.println("Processor information");

System.out.print("Number of cores : ");

int noOfCores = scanner.nextInt();

System.out.print("Manufacturer : ");

scanner.nextLine();

String ProcessorManufacturer = scanner.nextLine();

System.out.println("RAM information");

System.out.print("Memory size : ");

int size = scanner.nextInt();

System.out.print("Manufacturer : ");

scanner.nextLine();

String manufacturer = scanner.nextLine();

scanner.close();

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

System.out.println("Processor and RAM information:");

Processor processor = new Processor(noOfCores, ProcessorManufacturer);

RAM ram = RAM.getRAM(size, manufacturer);

processor.printProcessorInfo();

ram.printRAMInfo();

}

public static class Processor {

private int numberOfCores;

private String manufacturer;

public Processor(int numberOfCores, String manufacturer) {

this.numberOfCores = numberOfCores;

this.manufacturer = manufacturer;

}

public void printProcessorInfo() {

System.out.println(" Number of Cores: " + numberOfCores);

System.out.println(" Manufacturer: " + manufacturer);

}

}

public static class RAM {

private int memory;

private String manufacturer;

private RAM(int memory, String manufacturer) {

this.memory = memory;

this.manufacturer = manufacturer;

}

public static RAM getRAM(int memory, String manufacturer) {

return new RAM(memory, manufacturer); // Creates and returns a new RAM object

}

public void printRAMInfo() {

System.out.println(" RAM Size: " + memory + " GB");

System.out.println(" Manufacturer: " + manufacturer);

}

}

public static void main(String[] args) {

System.out.println("Name : Sreyas Satheesh\nRoll.no : 53\nTitle : Display CPU specs\nDate : 13/02/2024\n");

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the price of the cpu : ");

Cpu cpu = new Cpu(scanner.nextInt());

cpu.printCPUInfo();

scanner.close();

}

}

**Output :**

