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.

Program:

```
class Product {
   int pcode;
   String pname;
   double price;

   public Product(int x, String y, double z) {
     pcode = x;
     pname = y;
     price = z;
   }
 }
public class price {
public static void main(String[] args) {
System.out.println("Abin saji\n 23MCA003 \n 13-FEBRUARY-2024");
Product p1 = new Product(1, "Apple", 100);
Product p2 = new Product(2, "Grapes", 50);
Product p3 = new Product(3, "Pineapple", 70);
Product cheapestProduct = p1;
if (p2.price < cheapestProduct.price) {
cheapestProduct = p2;
 }
if (p3.price < cheapestProduct.price) {
cheapestProduct = p3;
 }
System.out.println("The product with the lowest price is " + cheapestProduct.pname + " with a
price of "  + cheapestProduct.price);
  }
}

      Product lowestPriceProduct = products[0];
      for (int i = 1; i < products.length; i++) {
         if (products[i].price < lowestPriceProduct.price) {
            lowestPriceProduct = products[i];
         }
      }
```

Output:

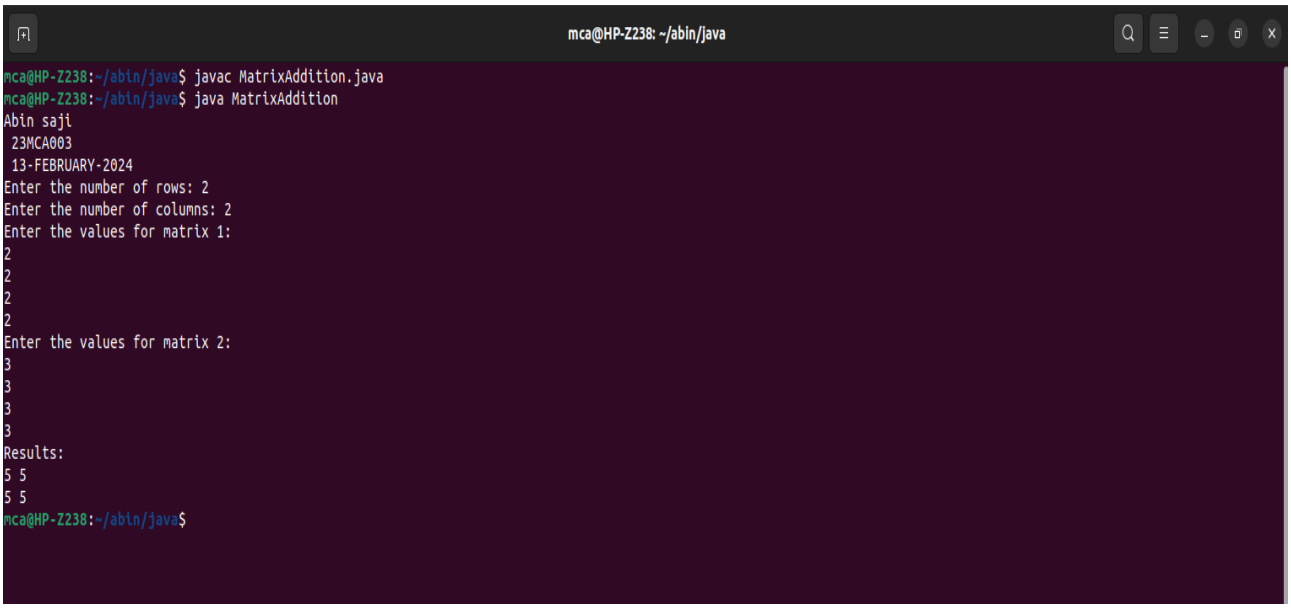

mca@HP-Z238: ~/abin/java
mca@HP-Z238:~/abin/java$ javac price.java
mca@HP-Z238:~/abin/java$ java price
Abin saji
23MCA003
13-FEBRUARY-2024
The product with the lowest price is Grapes with a price of 50.0
mca@HP-Z238:~/abin/java$

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

Program:

```
import java.util.Scanner;
public class MatrixAddition
 {
public static void main(String[] args)
{
System.out.println("Abin saji \n 23MCA003 \n 13-FEBRUARY-2024");
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of rows: ");
int rows = scanner.nextInt();
System.out.print("Enter the number of columns: ");
int cols = scanner.nextInt();
int[][] matrix1 = new int[rows][cols];
System.out.println("Enter the values for matrix 1:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
matrix1[i][j] = scanner.nextInt();
}
}
int[][] matrix2 = new int[rows][cols];
System.out.println("Enter the values for matrix 2:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
matrix2[i][j] = scanner.nextInt();
}
}
int[][] result = new int[rows][cols];
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
result[i][j] = matrix1[i][j] + matrix2[i][j];
}
}
System.out.println("Results:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < rows; j++) {
System.out.print(result[i][j] + " ");
}
System.out.println();
}
scanner.close();
}}
```

Output:


mca@HP-Z238: ~/abin/java
mca@HP-Z238:~/abin/java$ javac MatrixAddition.java
mca@HP-Z238:~/abin/java$ java MatrixAddition
Abin saji
23MCA003
13-FEBRUARY-2024
Enter the number of rows: 2
Enter the number of columns: 2
Enter the values for matrix 1:
2
2
2
2
Enter the values for matrix 2:
3
3
3
3
Results:
5 5
5 5
mca@HP-Z238:~/abin/java$

3. Add complex numbers

Program:

```
public class complex{
int r;
int i;
    complex(int real,int img){
    r=real;
    i=img;
    }
    void display(){
    System.out.println(r+"+"+i+"i");
    }
    static void add(int r1,int i1,int r2,int i2){
    r1=r1+r2;
    i1=i1+i2;
    System.out.println("After Addition ="+r1+"+"+i1+"i");
    }
    public static void main(String[] args){
    System.out.println("Abin saji\n 23MCA003 \n 14-FEBRUARY-2024");
            complex first=new complex(5,4);
            complex second=new complex(7,9);
            System.out.println("Complex Numbers are:");
            first.display();
            second.display();
            add(first.r,first.i,second.r,second.i);
  }}
```

Output:

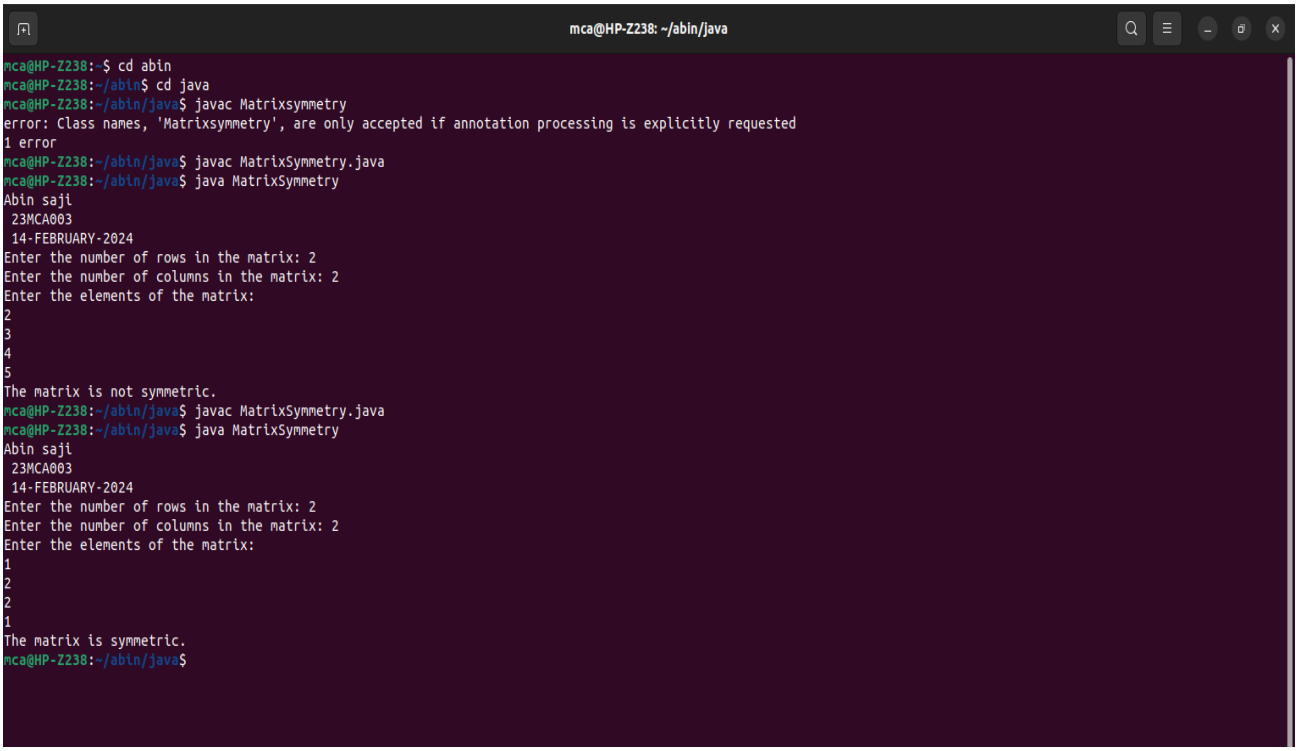

mca@HP-Z238: ~/abin/java
mca@HP-Z238:~/abin/java$ javac complex.java
mca@HP-Z238:~/abin/java$ java complex
Abin saji
23MCA003
14-FEBRUARY-2024
Complex Numbers are:
5+4i
7+9i
After Addition =12+13i
mca@HP-Z238:~/abin/java$

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

Program:

```
import java.util.Scanner;
public class MatrixSymmetry {
public static void main(String[] args) {
System.out.println("Abin saji\n 23MCA003 \n 14-FEBRUARY-2024");
Scanner input = new Scanner(System.in);
System.out.print("Enter the number of rows in the matrix: ");
int rows = input.nextInt();
System.out.print("Enter the number of columns in the matrix: ");
int cols = input.nextInt();
int[][] matrix = new int[rows][cols];
System.out.println("Enter the elements of the matrix:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
matrix[i][j] = input.nextInt();
}
}
boolean isSymmetric = true;
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
if (matrix[i][j] != matrix[j][i]) {
isSymmetric = false;
break;
}
}
if (!isSymmetric) {
break;
}
}
if (isSymmetric) {
System.out.println("The matrix is symmetric.");
} else {
System.out.println("The matrix is not symmetric.");
}
}
}
```

Output:


mca@HP-Z238: ~/abin/java
mca@HP-Z238:~$ cd abin
mca@HP-Z238:~/abin$ cd java
mca@HP-Z238:~/abin/java$ javac Matrixsymmetry
error: Class names, 'Matrixsymmetry', are only accepted if annotation processing is explicitly requested
1 error
mca@HP-Z238:~/abin/java$ javac MatrixSymmetry.java
mca@HP-Z238:~/abin/java$ java MatrixSymmetry
Abin saji
23MCA003
14-FEBRUARY-2024
Enter the number of rows in the matrix: 2
Enter the number of columns in the matrix: 2
Enter the elements of the matrix:
2
3
4
5
The matrix is not symmetric.
mca@HP-Z238:~/abin/java$ javac MatrixSymmetry.java
mca@HP-Z238:~/abin/java$ java MatrixSymmetry
Abin saji
23MCA003
14-FEBRUARY-2024
Enter the number of rows in the matrix: 2
Enter the number of columns in the matrix: 2
Enter the elements of the matrix:
1
2
2
1
The matrix is symmetric.
mca@HP-Z238:~/abin/java$

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

Program:

```
public class CPU {
private double price;
private Processor processor;
private RAM ram;
public CPU(double price, Processor processor, RAM ram) {
this.price = price;
this.processor = processor;
this.ram = ram;
}
public double getPrice() {
return price;
}
public Processor getProcessor() {
return processor;
}
public RAM getRam() {
return ram;
}
public static class Processor {
private int numberOfCores;
private String manufacturer;
public Processor(int numberOfCores, String manufacturer) {
this.numberOfCores = numberOfCores;
this.manufacturer = manufacturer;
}
public int getNumberOfCores() {
return numberOfCores;
}
public String getManufacturer() {
return manufacturer;
}
}
public static class RAM {
private int memory;
private String manufacturer;
public RAM(int memory, String manufacturer) {
this.memory = memory;
this.manufacturer = manufacturer;
```

```
}
public int getMemory() {
return memory;
}
public String getManufacturer() {
return manufacturer;
}
}
public static void main(String[] args) {
System.out.println("Abin saji \n 23MCA003 \n 14-FEBRUARY-2024");
Processor processor = new Processor(4, "Intel");
RAM ram = new RAM(8, "Kingston");
CPU cpu = new CPU(500.0, processor, ram);
System.out.println("CPU Price: " + cpu.getPrice());
System.out.println("Processor: " + cpu.getProcessor().getManufacturer() + " " +
cpu.getProcessor().getNumberOfCores() + " cores");
System.out.println("RAM: " + cpu.getRam().getManufacturer() + " " +
cpu.getRam().getMemory() + " GB");
}
}
```

Output: