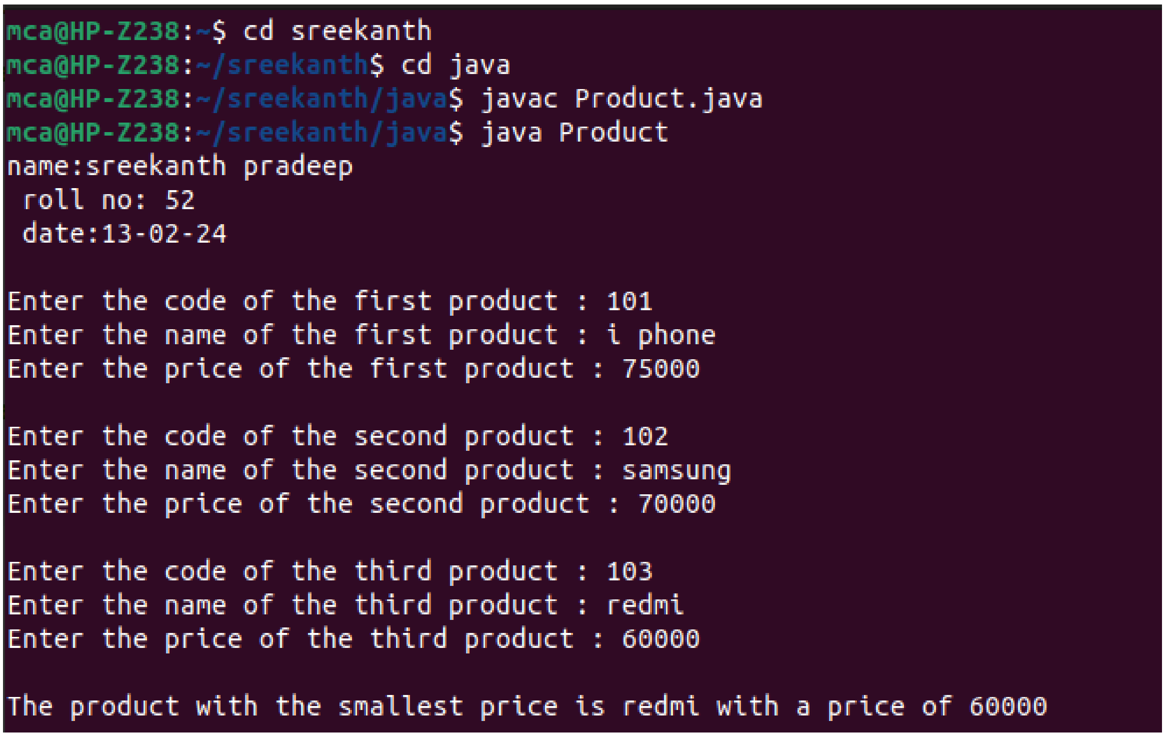
1. Define a class 'product' with data members pcode, pname and price. Create 3 objects of the class and find the product having the lowest price.

Source 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("name:sreekanth pradeep\n roll no: 52\n date:13-02-24");
                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");

                scan.close();
        }
}
```

Output:



```
mca@HP-Z238:~$ cd sreekanth
mca@HP-Z238:~/sreekanth$ cd java
mca@HP-Z238:~/sreekanth/java$ javac Product.java
mca@HP-Z238:~/sreekanth/java$ java Product
name:sreekanth pradeep
 roll no: 52
 date:13-02-24

Enter the code of the first product : 101
Enter the name of the first product : i phone
Enter the price of the first product : 75000

Enter the code of the second product : 102
Enter the name of the second product : samsung
Enter the price of the second product : 70000

Enter the code of the third product : 103
Enter the name of the third product : redmi
Enter the price of the third product : 60000

The product with the smallest price is redmi with a price of 60000
```

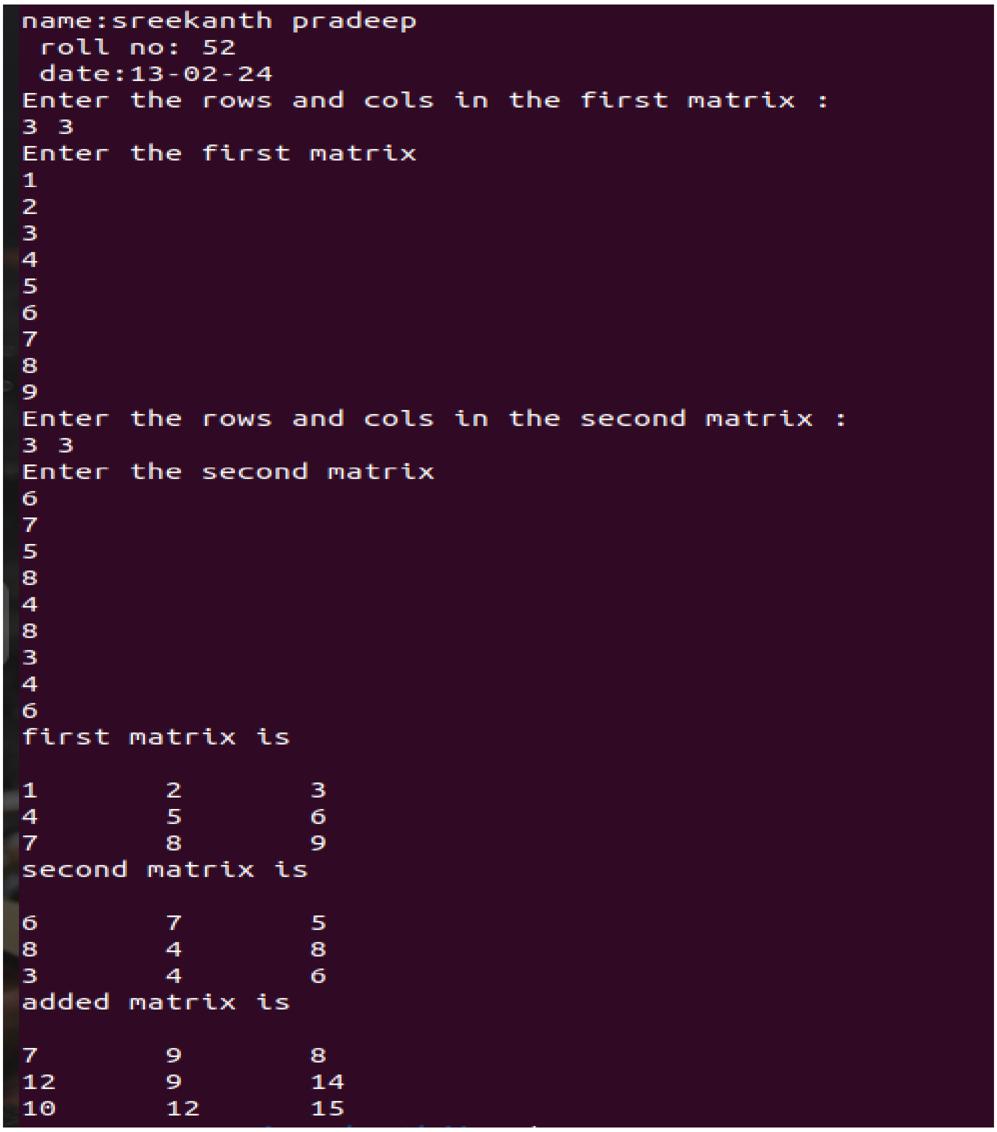
2. Read 2 matrices from the console and perform matrix addition.

Source Code:

```
  import java.util.Scanner;
public class Matrix{
    public static void main(String[] args){
    System.out.println("name:sreekanth pradeep\n roll no: 52\n date:13-02-24");
        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.");
    }
  }
}
```

Output:



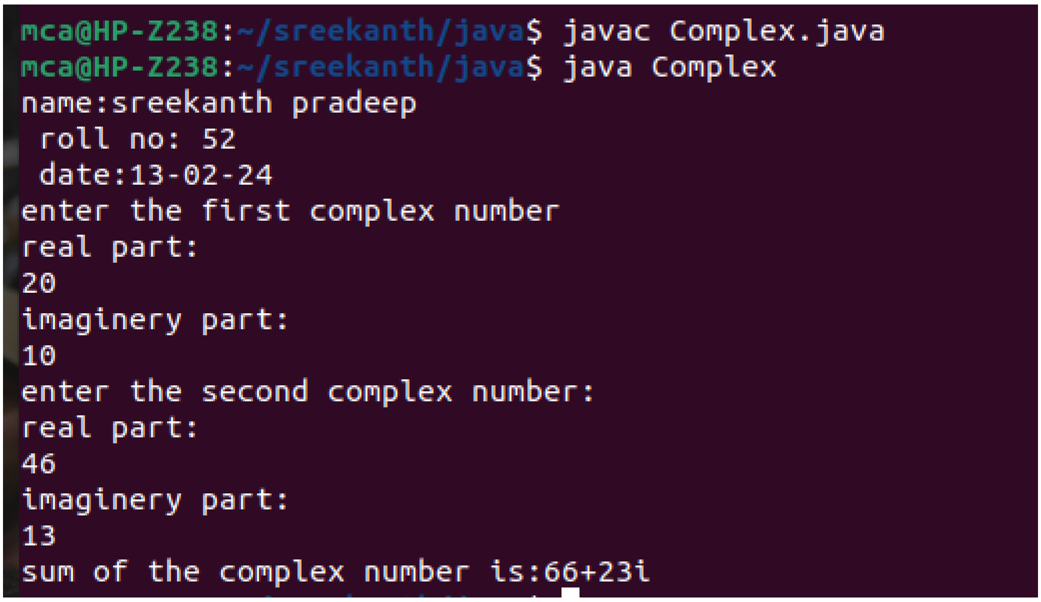
```
name:sreekanth pradeep
 roll no: 52
 date:13-02-24
Enter the rows and cols in the first matrix :
3 3
Enter the first matrix
1
2
3
4
5
6
7
8
9
Enter the rows and cols in the second matrix :
3 3
Enter the second matrix
6
7
5
8
4
8
3
4
6
first matrix is

1       2       3
4       5       6
7       8       9
second matrix is

6       7       5
8       4       8
3       4       6
added matrix is

7       9       8
12      9       14
10      12      15
```

3. Add complex numbers.

Source Code:

```
      import java.util.Scanner;
public class Complex{
        public static void main(String[] args){
                System.out.println("name:sreekanth pradeep\n roll no: 52\n date:13-02-24");
        class ComplexClass{
        int real;
        int img;
                public ComplexClass(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.println("real part:");
        a=scan.nextInt();
        System.out.println("imaginery part:");
        b=scan.nextInt();
        ComplexClass first = new ComplexClass(a,b);

        System.out.println("enter the second complex number:");
        System.out.println("real part:");
     a=scan.nextInt();
     System.out.println("imaginery part:");
     b=scan.nextInt();
     ComplexClass second = new ComplexClass(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");
     }
     }
```

Output:



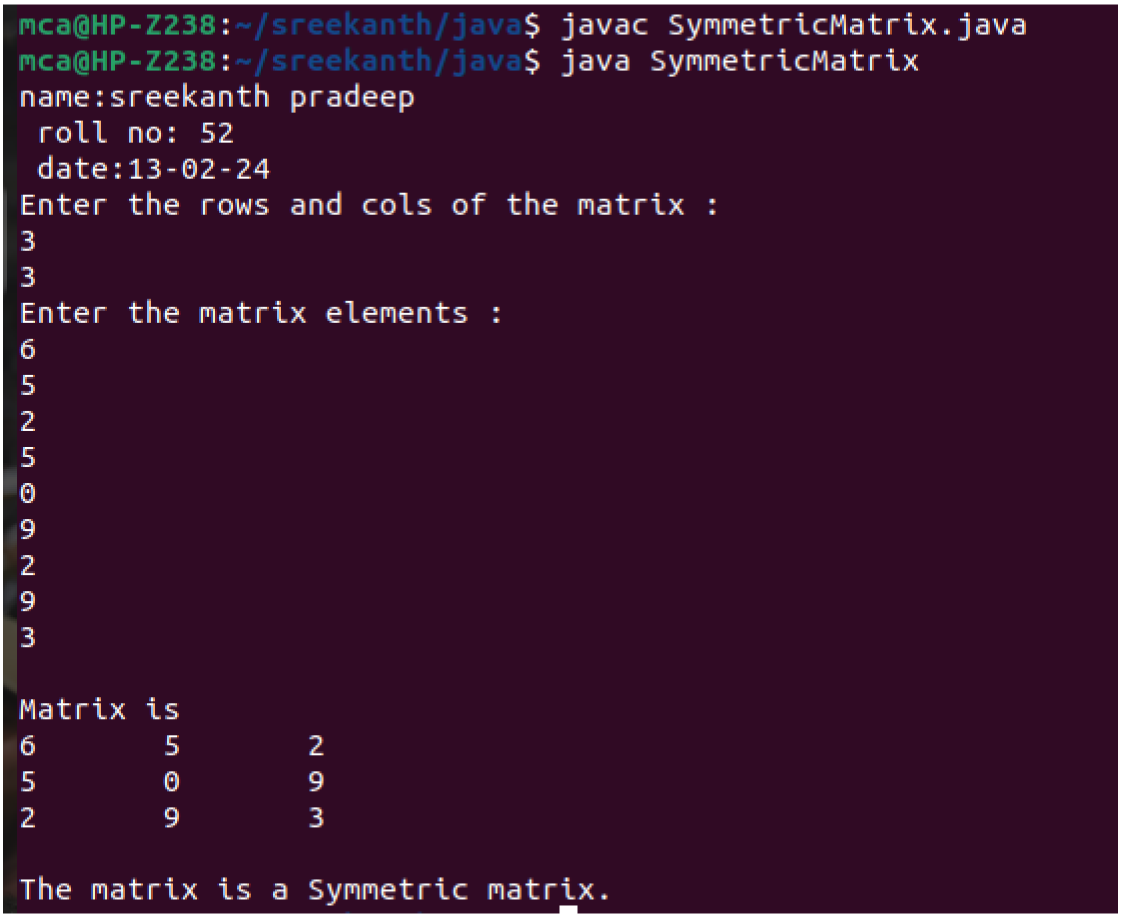
```
mca@HP-Z238:~/sreekanth/java$ javac Complex.java
mca@HP-Z238:~/sreekanth/java$ java Complex
name:sreekanth pradeep
 roll no: 52
 date:13-02-24
enter the first complex number
real part:
20
imaginery part:
10
enter the second complex number:
real part:
46
imaginery part:
13
sum of the complex number is:66+23i
```

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

Source Code:

```
import java.util.Scanner;

public class SymmetricMatrix{
    public static void main(String[] arg){
                        System.out.println("name:sreekanth pradeep\n roll no: 52\n date:13-02-24");
        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.");
        }
    }
```

Output:



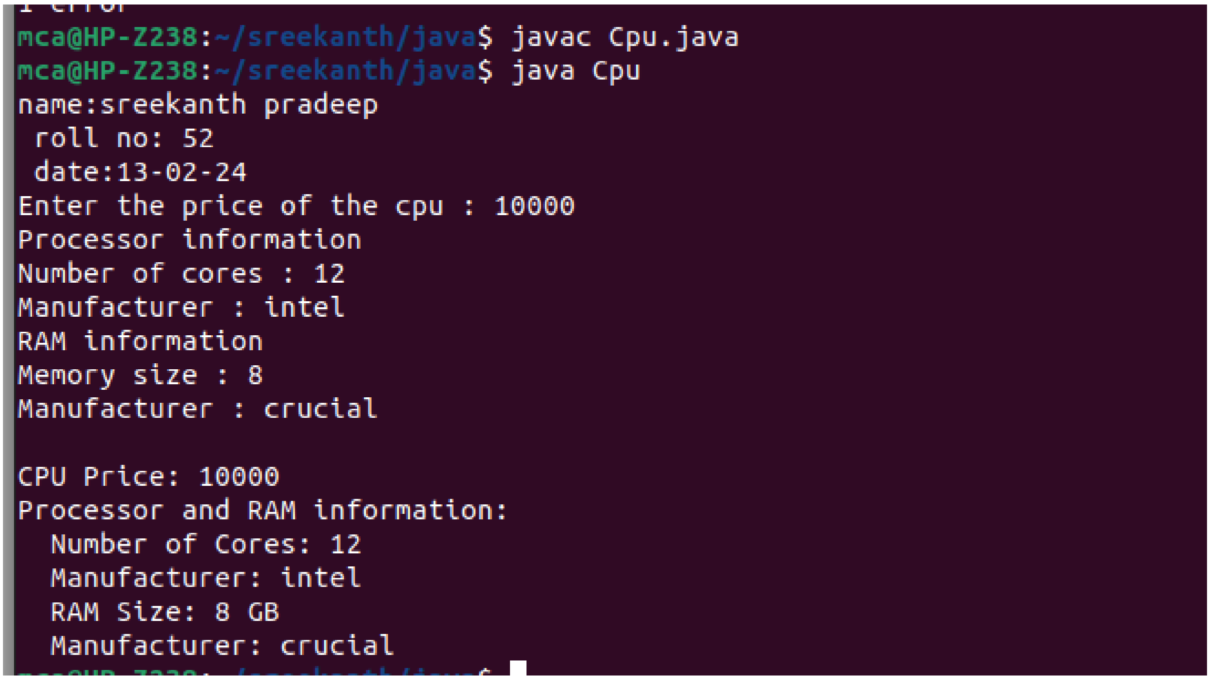
```
mca@HP-Z238:~/sreekanth/java$ javac SymmetricMatrix.java
mca@HP-Z238:~/sreekanth/java$ java SymmetricMatrix
name:sreekanth pradeep
 roll no: 52
 date:13-02-24
Enter the rows and cols of the matrix :
3
3
Enter the matrix elements :
6
5
2
5
0
9
2
9
3

Matrix is
6       5       2
5       0       9
2       9       3

The matrix is a Symmetric matrix.
```

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.

SourceCode:

```
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:sreekanth pradeep\n roll no: 52\n date:13-02-24");
        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:



```
1 error
mca@HP-Z238:~/sreekanth/java$ javac Cpu.java
mca@HP-Z238:~/sreekanth/java$ java Cpu
name:sreekanth pradeep
 roll no: 52
 date:13-02-24
Enter the price of the cpu : 10000
Processor information
Number of cores : 12
Manufacturer : intel
RAM information
Memory size : 8
Manufacturer : crucial

CPU Price: 10000
Processor and RAM information:
  Number of Cores: 12
  Manufacturer: intel
  RAM Size: 8 GB
  Manufacturer: crucial
mca@HP-Z238:~/sreekanth/java$ 
```