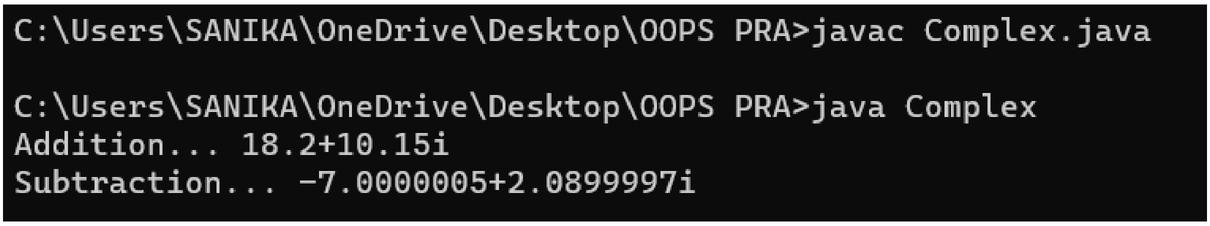
```
/*
Design a class ‘Complex ‘with data members for real and imaginary part.
Provide default and Parameterized constructors. Write a program to perform
arithmetic operations of two complex numbers.
*/
public class Complex
{
	float real,img;
	Complex(){ }

	Complex(float r,float i)
	{
		real = r;
		img = i;
	}
	Complex add(Complex a)
	{
		Complex temp = new Complex();
		temp.real = this.real + a.real;
		temp.img = this.img + a.img;
		return temp;
	}
	Complex sub(Complex a)
	{
		Complex temp = new Complex();
		temp.real = this.real - a.real;
		temp.img = this.img - a.img;
		return temp;
	}
	public static void main(String args[])
	{
```

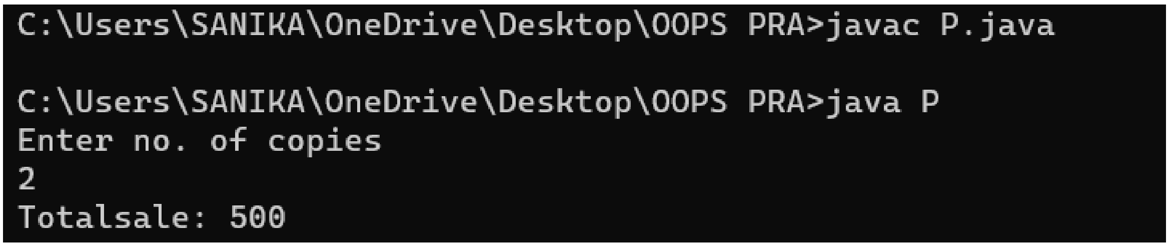
```
                Complex c1 = new Complex(5.6f,6.12f);

                Complex c2 = new Complex(12.6f,4.03f);

                Complex result;

                result = c1.add(c2);

                System.out.println("Addition... "+result.real+"+"+result.img+"i");

                result = c1.sub(c2);

                System.out.println("Subtraction... "+result.real+"+"+result.img+"i");

        }

}
```



```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac Complex.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Complex
Addition... 18.2+10.15i
Subtraction... -7.0000005+2.0899997i
```

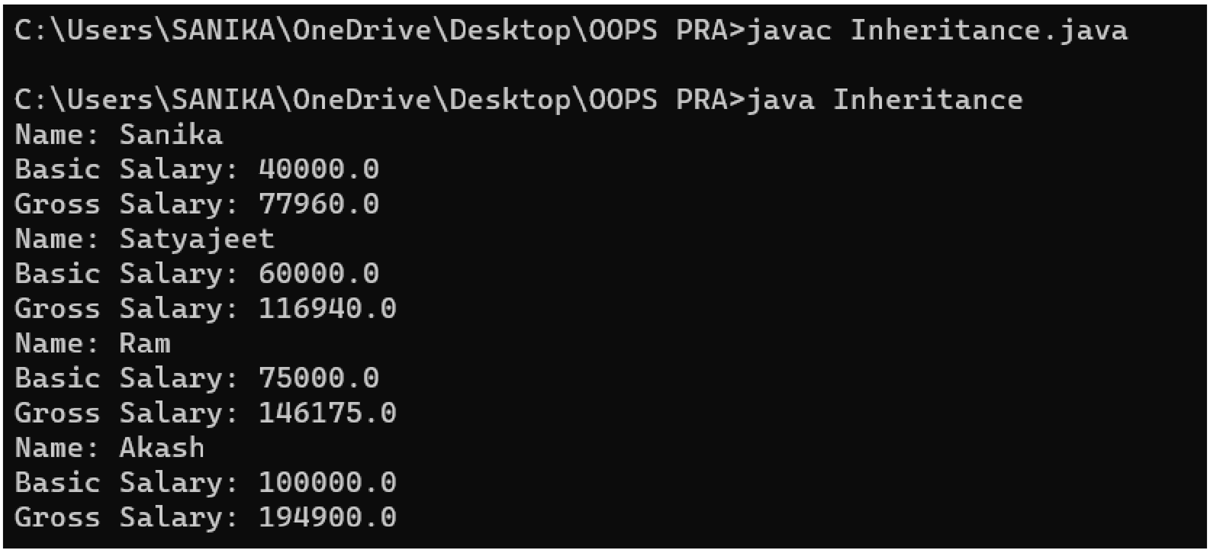
```
/*Identify commonalities and differences between Publication, Book and

Magazine classes. Title, Price, Copies are common instance variables and

saleCopy is common method. The differences are, Bookclass has author and

order Copies(). Magazine Class has orderQty, Currentissue, reciveissue().Write

a program to find how many copies of the given books are ordered and display

total sale of publication.*/

import java.util.Scanner;

class Publication

{

        String title;

        int copies,price;

        void saleCopy()

        {

        }

}

class Book

{
        String title,author;
        int copies,price;
        void saleCopy()
        {
            price = 250;
            System.out.println("Totalsale: "+copies*price);
        }
        void orderCopies()
        {
            Scanner in = new Scanner(System.in);
            System.out.println("Enter no. of copies");
            copies = in.nextInt();
        }
}
class Magazine
{
        String title;
        int copies,price;
        void saleCopy()
        {
        }
        void orderQty()
        {
        }
        void currentIssue()
        {
        }
        void receiveIssue()
        {
        }
}
```

```
public class P
{
        public static void main(String args[])
        {
                Book b = new Book();
                b.orderCopies();
                b.saleCopy();
        }
}
```



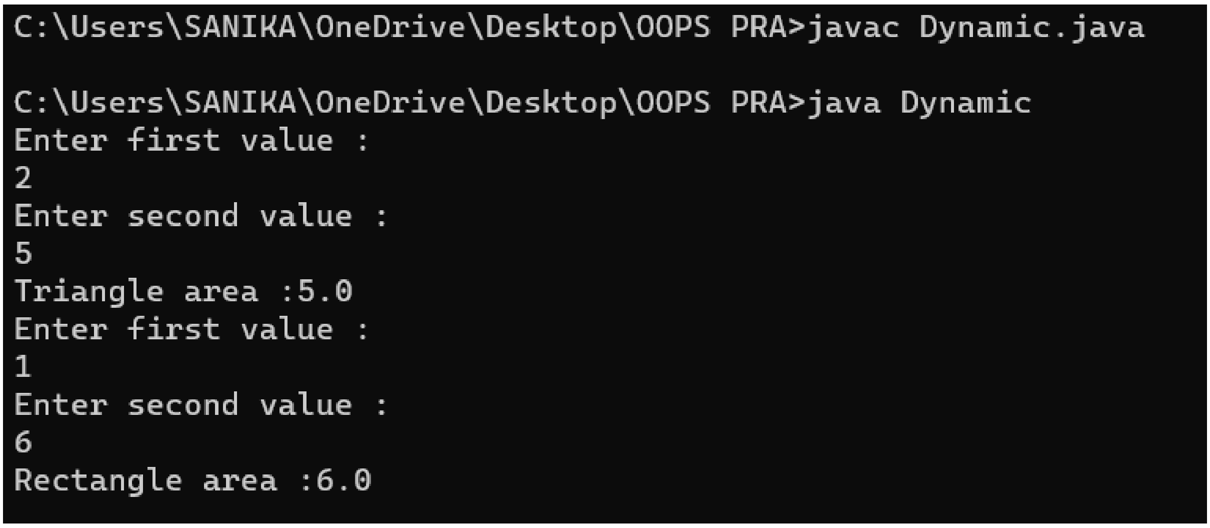
```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac P.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java P
Enter no. of copies
2
Totalsale: 500
```

```
/*
Design and develop inheritance for a given case study, identify objects
and relationships and implement inheritance wherever applicable. Employee
class with Emp_name, Emp_id, Address, Mail_id, and Mobile_no as members.
Inherit the classes, Programmer, Assistant Professor, Associate Professor and
Professor from employee class. Add Basic Pay (BP) as the member of all the
inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF,
0.1% of BP for staff club fund. Generate pay slips for the employees with their
gross and net salary.
*/
class Employee
{
        String name, add, mail;
        float id, mobile,basic;
        void salary()
        {
                float da,hra,pf,cf,gross;
                da = basic * 97/100;
        hra = basic * 10/100;
        pf = basic * 12/100;
        cf = basic * 0.1f/100;
        gross = basic + da + hra - pf - cf;
        System.out.println("Name: " + name);
        System.out.println("Basic Salary: " + basic);
        System.out.println("Gross Salary: " + gross);
    }
}
class Programmer extends Employee
{
    Programmer(String name,int sal)
    {
        this.name = name;
        basic = sal;
    }
}
class Assistant_Professor extends Employee
{
    Assistant_Professor(String name,int sal)
    {
        this.name = name;
        basic = sal;
    }
}
class Associate_Professor extends Employee
{
    Associate_Professor(String name,int sal)
    {
        this.name = name;
        basic = sal;
        }
}
class Professor extends Employee
{
        Professor(String name,int sal)
        {
                this.name = name;
                basic = sal;
        }
}
public class Inheritance
{
        public static void main(String args[])
        {
                Assistant_Professor ast = new Assistant_Professor("Sanika",40000);
                ast.salary();
                Associate_Professor aso = new Associate_Professor("Satyajeet",60000);
                aso.salary();
                Professor pro = new Professor("Ram",75000);
                pro.salary();
                Programmer pm = new Programmer("Akash",100000);
                pm.salary();
        }
}
```



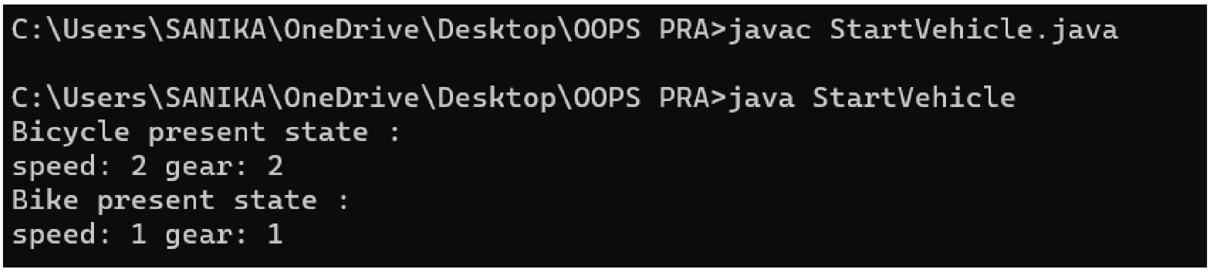
```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac Inheritance.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Inheritance
Name: Sanika
Basic Salary: 40000.0
Gross Salary: 77960.0
Name: Satyajeet
Basic Salary: 60000.0
Gross Salary: 116940.0
Name: Ram
Basic Salary: 75000.0
Gross Salary: 146175.0
Name: Akash
Basic Salary: 100000.0
Gross Salary: 194900.0
```

```
/*
Design a base class shape with two double type values and member
functions to input the data and compute_area() for calculating area of figure.
Derive two classes’ triangle and rectangle. Make compute_area() as abstract
function and redefine this function in the derived class to suit their requirements.
Write a program that accepts dimensions of triangle/rectangle and display
calculated area. Implement dynamic binding for given case study.
*/
import java.util.*;
abstract class Shape
{
        double val1,val2;
        void input()
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter first value :");
                val1=sc.nextDouble();
                System.out.println("Enter second value :");
                val2=sc.nextDouble();
        }
        abstract void compute_area();
}
class Triangle extends Shape
{
        void compute_area()
        {
                double area;
                area = 1.0f/2.0f*val1*val2;
                System.out.println("Triangle area :"+area);
        }
}
class Rectangle extends Shape
{
        void compute_area()
        {
                double area;
                area=val1*val2;
                System.out.println("Rectangle area :"+area);
        }
}
class Dynamic
{
        public static void main(String args[])
        {
                Shape s;
                Triangle t=new Triangle();
                Rectangle r=new Rectangle();
                s=t;
                s.input();
                s.compute_area();
                s=r;
                s.input();

                s.compute_area();

        }

}
```



```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac Dynamic.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Dynamic
Enter first value :
2
Enter second value :
5
Triangle area :5.0
Enter first value :
1
Enter second value :
6
Rectangle area :6.0
```

```
/*

Design and develop a context for given case study and implement an

interface for Vehicles Consider the example of vehicles like bicycle, car, and bike.

All Vehicles have common functionalities such as Gear Change, Speed up and

apply breaks. Make an interface and put all these common functionalities.

Bicycle, Bike, Car classes should be implemented for all these functionalities in

their own class in their own way.

*/

import java.io.*;

interface Vehicle

{

        void changeGear(int a);

        void speedUp(int a);

        void applyBrakes(int a);

}

class Bicycle implements Vehicle

{
        int speed;
        int gear;
        public void changeGear(int newGear)
        {
                gear = newGear;
        }
        public void speedUp(int increment)
        {
                speed = speed + increment;
        }
        public void applyBrakes(int decrement)
        {
                speed = speed - decrement;
        }
        public void printStates()
        {
                System.out.println("speed: " + speed+ " gear: " + gear);
        }
}
class Bike implements Vehicle
{
        int speed;
        int gear;
        public void changeGear(int newGear)
        {
                gear = newGear;
        }
        public void speedUp(int increment)
        {
                speed = speed + increment;
        }
        public void applyBrakes(int decrement)
        {
                speed = speed - decrement;
        }
        public void printStates()
        {
                System.out.println("speed: " + speed+ " gear: " + gear);
        }
}
class StartVehicle
{
        public static void main (String[] args)
        {
                Bicycle bicycle = new Bicycle();
                bicycle.changeGear(2);
                bicycle.speedUp(3);
                bicycle.applyBrakes(1);
                System.out.println("Bicycle present state :");
                bicycle.printStates();
                Bike bike = new Bike();
                bike.changeGear(1);
                bike.speedUp(4);
                bike.applyBrakes(3);
                System.out.println("Bike present state :");
                bike.printStates();
        }
}
```



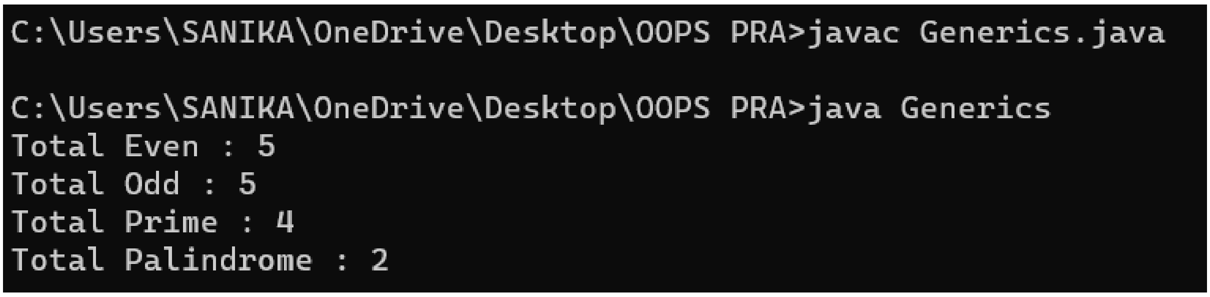
```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac StartVehicle.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java StartVehicle
Bicycle present state :
speed: 2 gear: 2
Bike present state :
speed: 1 gear: 1
```

Implement a generic program using any collection class to count the number of elements in a collection that have a specific property such as even numbers, odd number, prime number and palindromes.

```
public class Generics

{

  static boolean isPrime(int num){

    int flag =0;

    for(int i = 2;i<num;i++)

      if(num%i==0)

      {

        flag = 1;

        break;

      }

    if(flag==0)

      return true;

    return false;

  }

  static <T> void count(String str,T[] element){

    int even=0,odd=0,prime=0,palin=0;

    if(str.equals("even")){

      for(T value:element)

        if(Integer.parseInt(value.toString())%2==0)

          even++;

    System.out.println("Total Even : "+even);

    }

    if(str.equals("odd")){

      for(T value:element)
```

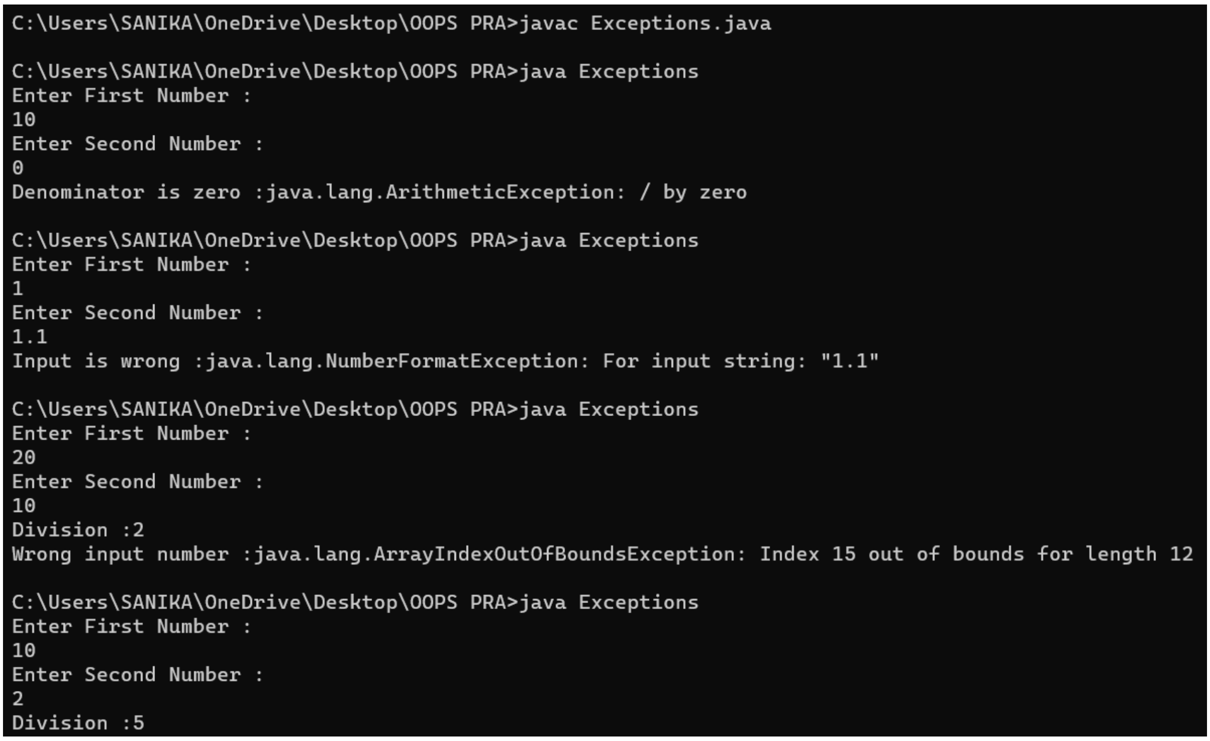
```
            if(Integer.parseInt(value.toString())%2!=0)
               odd++;
      System.out.println("Total Odd : "+odd);
      }
      if(str.equals("prime")){
         for(T value:element)
            if(isPrime(Integer.parseInt(value.toString())))
               prime++;
      System.out.println("Total Prime : "+prime);
      }
      if(str.equals("palindrome")){
         for(T value:element){
            StringBuffer rev = new StringBuffer(value.toString());
            if(value.toString().equals(new String(rev.reverse())))
               palin++;
         }
         System.out.println("Total Palindrome : "+palin);
      }
   }
   public static void main(String[] args){
      Integer iarray[] = {45,70,12,84,38,151,29,30,19,11};
      count("even",iarray);
      count("odd",iarray);
      count("prime",iarray);
      count("palindrome",iarray);
   }
}
```

Output :



```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac Generics.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Generics
Total Even : 5
Total Odd : 5
Total Prime : 4
Total Palindrome : 2
```

```
/*
Implement a program to handle Arithmetic exception, Array Index Out Of
Bounds. The user enters two numbers Num1 and Num2. The division of Num1
and Num2 is displayed. If Num1 and Num2 were not integers, the program would
throw a Number Format Exception. If Num2 were zero, the program would throw
an Arithmetic Exception. Display the exception
*/
import java.util.*;
public class Exceptions
{
    public static void main(String argsp[])
    {
        int num1,num2,result;
        Scanner sc=new Scanner(System.in);
        try
        {
            System.out.println("Enter First Number :");
            num1=Integer.parseInt(sc.next());
            System.out.println("Enter Second Number :");
            num2=Integer.parseInt(sc.next());
            result=num1/num2;
            System.out.println("Division :"+result);
            int x[]=new int[12];
            x[15]=10;
            System.out.println("x[15]="+x[15]);
        }
        catch(ArithmeticException e)
        {
            System.out.println("Denominator is zero :"+e);
        }
        catch(NumberFormatException e)
        {
            System.out.println("Input is wrong :"+e);
        }
        catch(ArrayIndexOutOfBoundsException e)
        {
            System.out.println("Wrong input number :"+e);
        }
        finally
    }
}
```



```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac Exceptions.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Exceptions
Enter First Number :
10
Enter Second Number :
0
Denominator is zero :java.lang.ArithmeticException: / by zero

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Exceptions
Enter First Number :
1
Enter Second Number :
1.1
Input is wrong :java.lang.NumberFormatException: For input string: "1.1"

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Exceptions
Enter First Number :
20
Enter Second Number :
10
Division :2
Wrong input number :java.lang.ArrayIndexOutOfBoundsException: Index 15 out of bounds for length 12

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java Exceptions
Enter First Number :
10
Enter Second Number :
2
Division :5
```

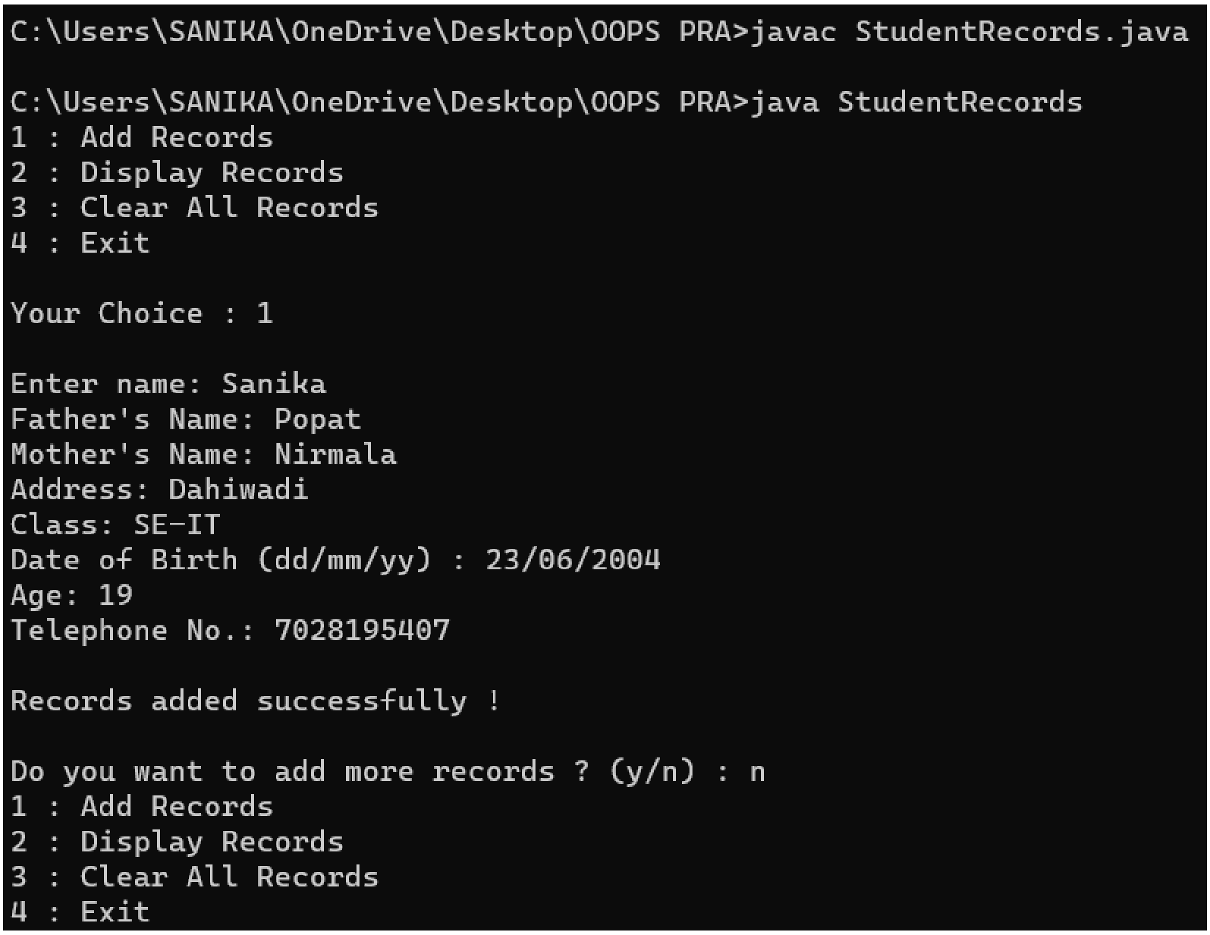
```
/*Implement a program for maintaining a student records database using

File Handling. Student has student_id, name, Roll_no, Class, marks and address.
```

```
Display the data for five students.

i) Create Database

ii) Display Database

iii) Clear Records

iv) Modify record

v) Search Record

*/


import java.io.*;

class StudentRecords

{

        static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

         public void addRecords() throws IOException

         {

         // Create or Modify a file for Database

         PrintWriter pw = new PrintWriter(new BufferedWriter(new

         FileWriter("studentRecords.txt",true)));

         String name, Class, fname, mname, address, dob;

         int age;

         long telephoneNo;

         String s;

         boolean addMore = false;

         // Read Data

         do

         {

          System.out.print("\nEnter name: ");

          name = br.readLine();

          System.out.print("Father's Name: ");

          fname = br.readLine();

          System.out.print("Mother's Name: ");

          mname = br.readLine();
          System.out.print("Address: ");
          address = br.readLine();
          System.out.print("Class: ");
          Class = br.readLine();
          System.out.print("Date of Birth (dd/mm/yy) : ");
          dob = br.readLine();
          System.out.print("Age: ");
          age = Integer.parseInt(br.readLine());

          System.out.print("Telephone No.: ");
          telephoneNo = Long.parseLong(br.readLine());
          // Print to File
          pw.println(name);
          pw.println(fname);
          pw.println(mname);
          pw.println(address);
          pw.println(Class);
          pw.println(dob);
          pw.println(age);
          pw.println(telephoneNo);
                System.out.print("\nRecords added successfully !\n\nDo you want to add more
records ? (y/n) : ");
          s = br.readLine();
          if(s.equalsIgnoreCase("y"))
          {
           addMore = true;
           System.out.println();
          }
          else
           addMore = false;
         }
```

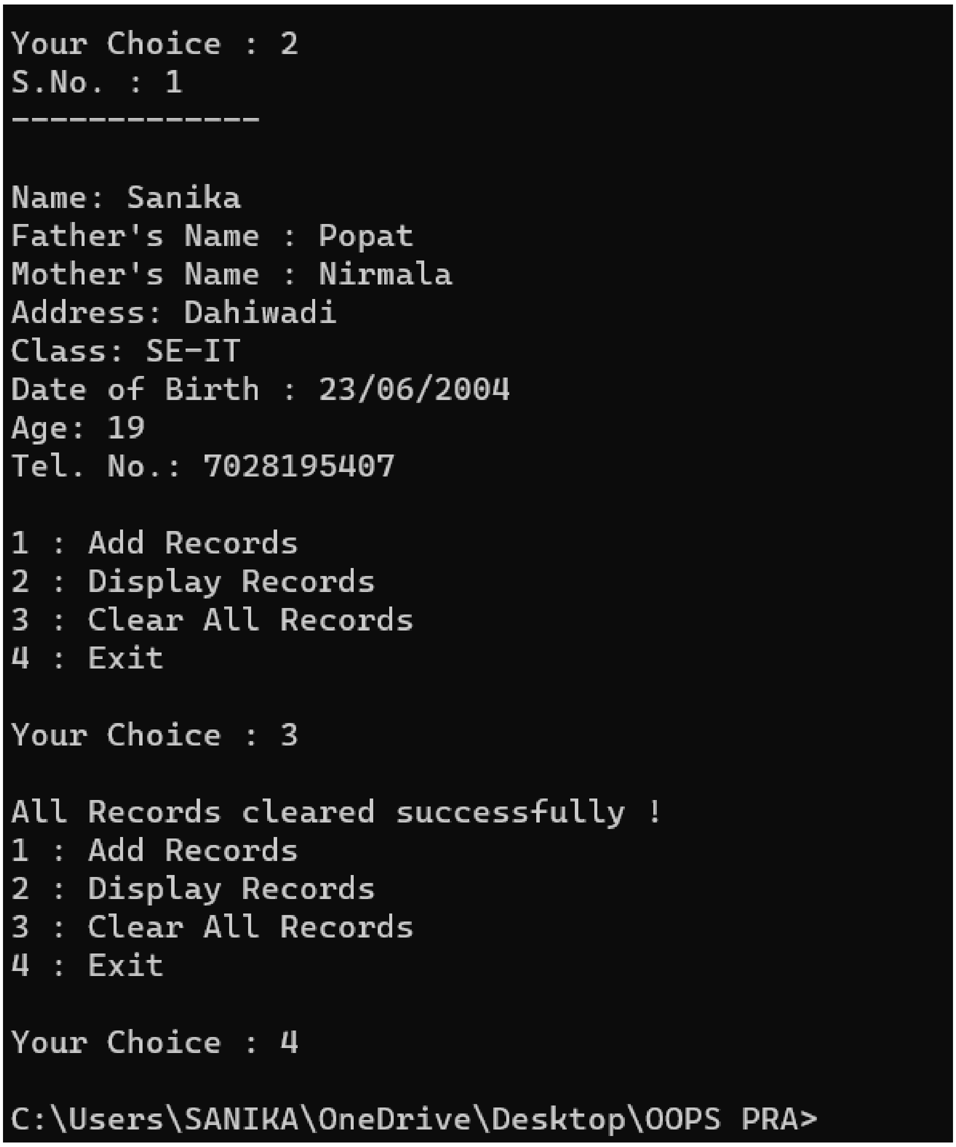
```
while(addMore);
pw.close();
showMenu();
}
public void readRecords() throws IOException
{
try
{
 // Open the file
 BufferedReader file = new BufferedReader(new
 FileReader("studentRecords.txt"));
 String name;
 int i=1;
 // Read records from the file
 while((name = file.readLine()) != null)
 {
  System.out.println("S.No. : " +(i++));
  System.out.println("-------------");
  System.out.println("\nName: " +name);
  System.out.println("Father's Name : "+file.readLine());
  System.out.println("Mother's Name : "+file.readLine());
  System.out.println("Address: "+file.readLine());
  System.out.println("Class: "+file.readLine());
  System.out.println("Date of Birth : "+file.readLine());
  System.out.println("Age: "+Integer.parseInt(file.readLine()));
  System.out.println("Tel. No.: "+Long.parseLong(file.readLine()));
  System.out.println();
 }
 file.close();
 showMenu();
}
```

```
catch(FileNotFoundException e)
{
 System.out.println("\nERROR : File not Found !!!");
}
}
public void clear() throws IOException
{
// Create a blank file
PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("studentRecords.txt")));
pw.close();
System.out.println("\nAll Records cleared successfully !");
for(int i=0;i<999999999;i++); // Wait for some time
showMenu();
}
public void showMenu() throws IOException
{
System.out.print("1 : Add Records\n2 : Display Records\n3 : Clear All Records\n4 : Exit\n\nYour Choice : ");
int choice = Integer.parseInt(br.readLine());
switch(choice)
{
 case 1:
  addRecords();
  break;
 case 2:
  readRecords();
  break;
 case 3:
  clear();
  break;
```

```
            case 4:
             System.exit(1);
             break;
            default:
             System.out.println("\nInvalid Choice !");
             break;
           }
           }
           public static void main(String args[]) throws IOException
           {
           StudentRecords call = new StudentRecords();
           call.showMenu();
           }
}
```



```
C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>javac StudentRecords.java

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>java StudentRecords
1 : Add Records
2 : Display Records
3 : Clear All Records
4 : Exit

Your Choice : 1

Enter name: Sanika
Father's Name: Popat
Mother's Name: Nirmala
Address: Dahiwadi
Class: SE-IT
Date of Birth (dd/mm/yy) : 23/06/2004
Age: 19
Telephone No.: 7028195407

Records added successfully !

Do you want to add more records ? (y/n) : n
1 : Add Records
2 : Display Records
3 : Clear All Records
4 : Exit
```



```
Your Choice : 2
S.No. : 1
-------------

Name: Sanika
Father's Name : Popat
Mother's Name : Nirmala
Address: Dahiwadi
Class: SE-IT
Date of Birth : 23/06/2004
Age: 19
Tel. No.: 7028195407

1 : Add Records
2 : Display Records
3 : Clear All Records
4 : Exit

Your Choice : 3

All Records cleared successfully !
1 : Add Records
2 : Display Records
3 : Clear All Records
4 : Exit

Your Choice : 4

C:\Users\SANIKA\OneDrive\Desktop\OOPS PRA>
```