(Established under Haryana Private University Act, 2006 as amended by Act No. 8 of 2013)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

LABORATORY RECORD

Programme : UG

Specialization : Computer Science and Engineering

Semester & Year : IV & II

Name of Student : Avinash kumar

Registration. No : 10322210004

Section : A

Course Code : 21CS2116

Course Title : Java Programming Laboratory

Name of Teachers : Dr. M. Mohan

BONAFIDE CERTIFICATE

Name : Avinash kumar

Reg. No : 10322210004

Subject Code : 21CS2116

Subject : Java Programming Laboratory

Course : B.Tech / Computer Science and Engineering

Certified that this is the bonafide record of practical done as a part of IV semester during the academic year 2023 - 2024.

Faculty Incharge **Head of the Department**

(Dr. M. Mohan) **(Dr. M. Mohan)**

Submitted for University Practical Examination held on / / 2024

Internal Examiner-I **Internal Examiner-II**

21CS2116 – Java Programming Laboratory

LIST OF EXPERIMENTS

21CS2116 – Java Programming Laboratory

1. CLASS AND OBJECT

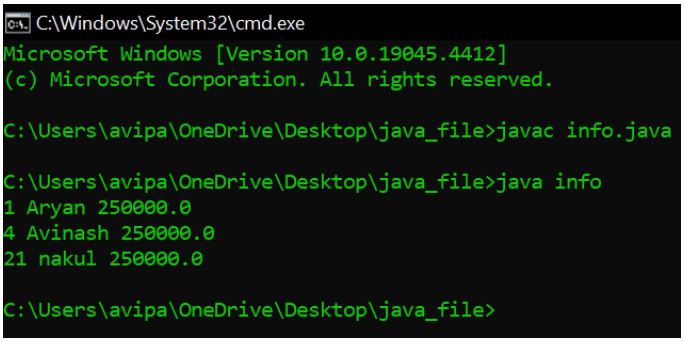
AIM:

To write a program, to understand how to make a class and object in java.

PROGRAM:

```
class Employee{
   int id;
   String name;
   float salary;
   void insert(int i, String n, float s) {
      id=i;
      name=n;
      salary=s;
   }
   void display(){System.out.println(id+" "+name+" "+salary);}
}
public class info {
public static void main(String[] args) {
   Employee e1=new Employee();
   Employee e2=new Employee();
   Employee e3=new Employee();
   e1.insert(001,"Aryan",250000);
   e2.insert(004,"Avinash",250000);
   e3.insert(025,"nakul",250000);
   e1.display();
   e2.display();
   e3.display();
}
}
```

OUTPUT:



RESULT:

2. INHERITANCE AND INTERFACE

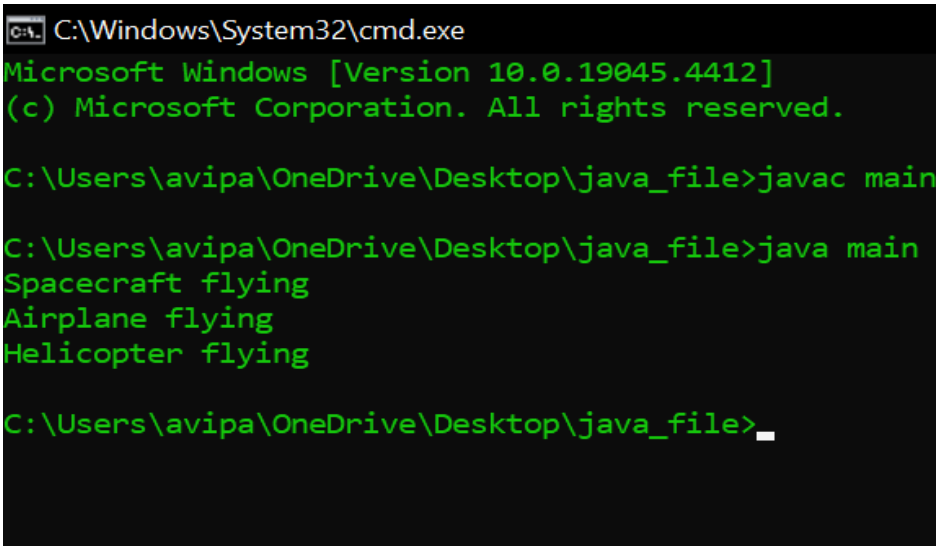
AIM:

To write a program, to understand how to make a inheritance and interface in java.

PROGRAM:

```
interface flyable{
void fly_obj();
}

class Spacecraft implements flyable{
public void fly_obj(){
System.out.println("Spacecraft flying");
}
}

class Airplane implements flyable{
public void fly_obj(){
System.out.println("Airplane flying");
}
}

class Helicopter implements flyable{
public void fly_obj(){
System.out.println("Helicopter flying");
}
}

public class main{
public static void main (String[] args){
Spacecraft space = new Spacecraft();
space.fly_obj();
Airplane air = new Airplane();
air.fly_obj();
Helicopter heli = new Helicopter();
heli.fly_obj();
}
}
```

OUTPUT:



RESULT:

21CS2116 – Java Programming Laboratory

3. EXCEPTION HANDLING

AIM:

To write a program, to understand how to implement Exception handling in java.

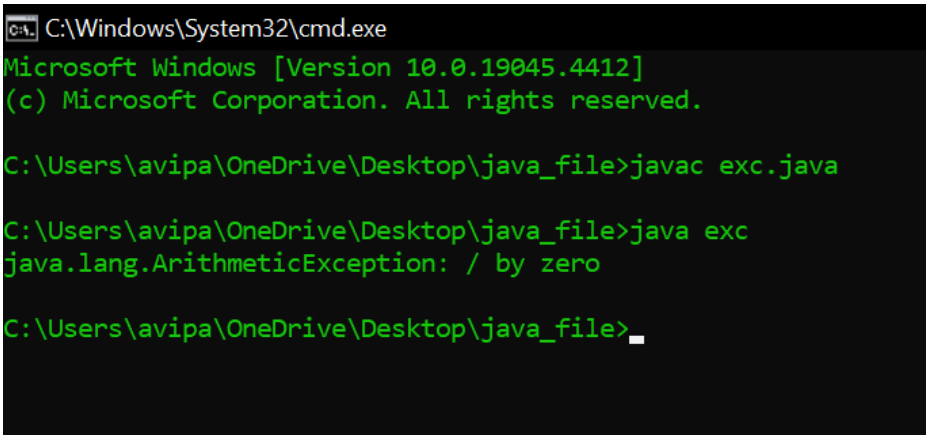
PROGRAM:

```
// Using try catch block
import java.io.*;
public class exc{
public static void main(String[] args){
try{
int a = 50, b = 0, c;
c = a/b;
System.out.println(c);
}
catch( ArithmeticException x){
System.out.println(x);
}
}
}
```

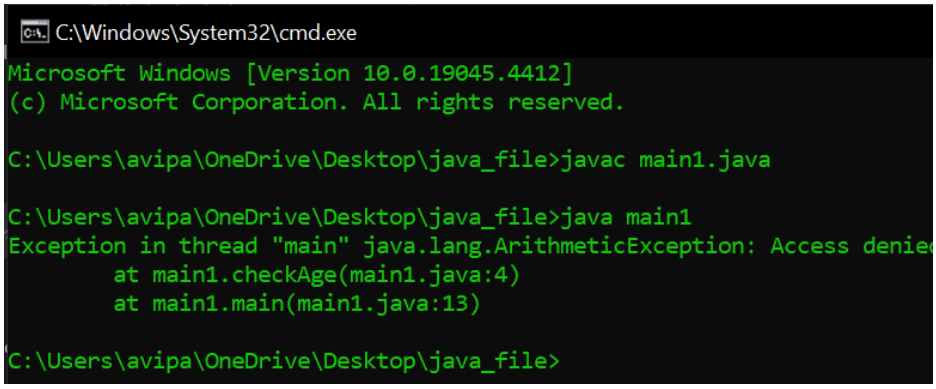
```
// Using throw and throws
public class main1 {
static void checkAge(int age) throws ArithmeticException {
if (age < 18) {
throw new ArithmeticException("Access denied - You must be at least 18 years old.");
}
else {
System.out.println("Access granted - You are old enough!");
}
}
public static void main(String[] args) {
checkAge(15);
}
}
```

OUTPUT:

Using Try-Catch block



Using Throw-Throws



RESULT:

4. EVENT HANDLING

AIM:

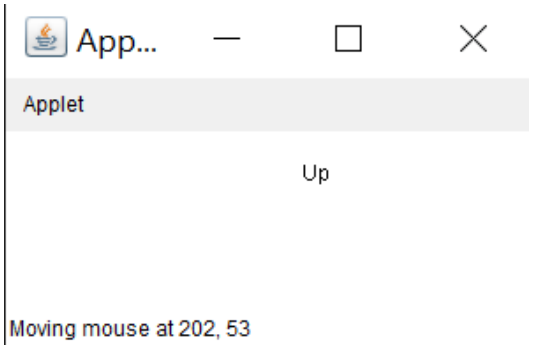
To write a program, to implement Event Handling in java.

PROGRAM:

```
// Demonstrate the mouse event handlers.
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="MouseEvents" width=300 height=100>
</applet>
*/
public class MouseEvents extends Applet
implements MouseListener, MouseMotionListener {
String msg = "";
int mouseX = 0, mouseY = 0; // coordinates of mouse
public void init() {
addMouseListener(this);
addMouseMotionListener(this);
}
public void mouseClicked(MouseEvent me) {
mouseX = 0;
mouseY = 10;
msg = "Mouse clicked.";
repaint();
}
// Handle mouse entered.
public void mouseEntered(MouseEvent me) {
// save coordinates
mouseX = 0;
mouseY = 10;
msg = "Mouse entered.";
repaint();
}
// Handle mouse exited.
public void mouseExited(MouseEvent me) {
// save coordinates
mouseX = 0;
mouseY = 10;
msg = "Mouse exited.";
repaint();
}
// Handle button pressed.
public void mousePressed(MouseEvent me) {
// save coordinates
mouseX = me.getX();
mouseY = me.getY();
```

```
msg = "Down";
repaint();
}
// Handle button released.
public void mouseReleased(MouseEvent me) {
// save coordinates
mouseX = me.getX();
mouseY = me.getY();
msg = "Up";
repaint();
}
public void mouseDragged(MouseEvent me) {
mouseX = me.getX();
mouseY = me.getY();
msg = "*";
showStatus("Dragging mouse at " + mouseX + ", " + mouseY);
repaint();
}
// Handle mouse moved.
public void mouseMoved(MouseEvent me) {
showStatus("Moving mouse at " + me.getX() + ", " + me.getY());
}
// Display msg in applet window at current X,Y location.
public void paint(Graphics g) {
g.drawString(msg, mouseX, mouseY);
}
}
```

OUTPUT:



RESULT:

5. MULTITHREADING

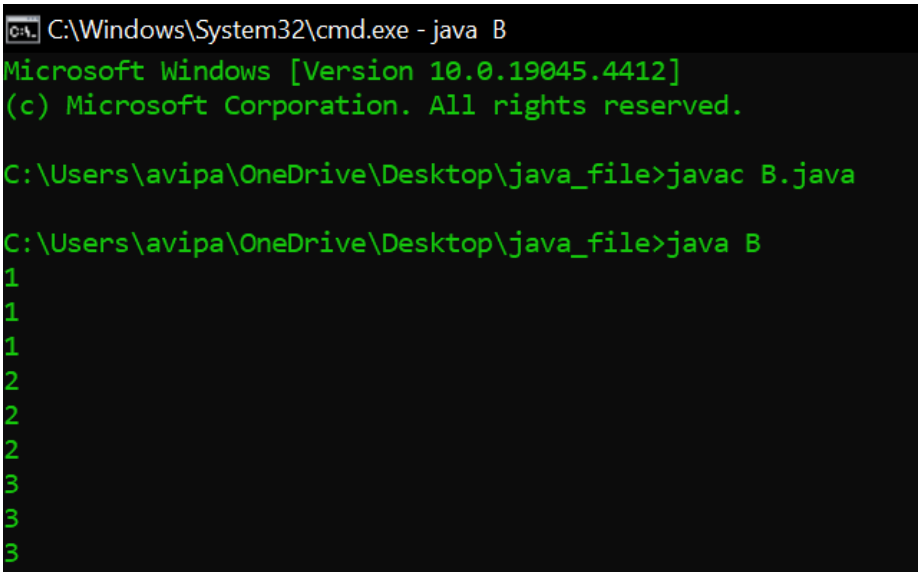
AIM:

To write a program, to implement the multithreading concept in java.

PROGRAM:

```
class A extends Thread{
public void run(){
String n = Thread.currentThread().getName();
for (int i=1;1<=3;i++){
System.out.println(i);
try{
Thread.sleep(1000);
}
catch(InterruptedException x){
}
}
}
}

class B{
public static void main(String[] args){
A t1 = new A();
A t2 = new A();
A t3 = new A();

t1.setName("Thread 1");
t2.setName("Thread 2");
t3.setName("Thread 3");

t1.start();
t2.start();
t3.start();
}
}
```

OUTPUT:



RESULT:

6. APPLET

AIM:

To write a program, to implement Applet in java.

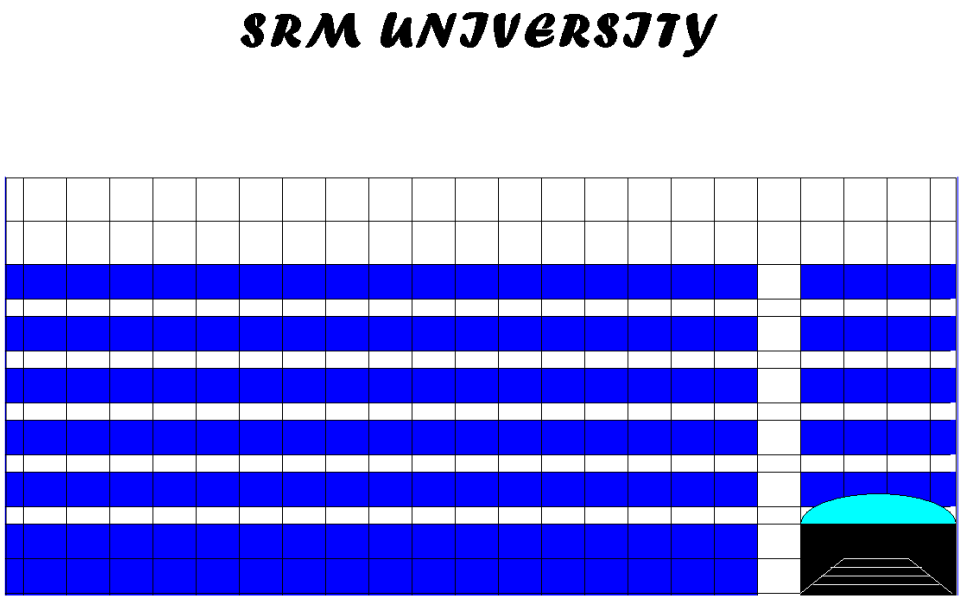
PROGRAM:

```
import java.awt.*;
import java.applet.*;
/*
 <applet complet code="eb" height="1000" width="2000") >
</applet>

*/
public class eb extends Applet
{

Font f3;

public void  init()
   {

     f3 = new Font("Forte",Font.BOLD,60);
   }
 public void paint(Graphics g)
 {

   g.setFont(f3);
   g.drawString("SRM UNIVERSITY" , 750 ,300 );
   g.setColor(Color.BLUE);
   g.fillRect(480, 450, 1100, 500);

   g.drawLine(479 , 449 ,479 ,1700); // firsthori
   g.drawLine(1582 , 449 ,1582, 1100); // secondhori


   g.setColor(Color.WHITE);
   g.fillRect(481 , 450 ,1100 ,400);

   g.setColor(Color.BLUE);
   g.fillRect(480, 450, 1100, 380);

   g.setColor(Color.WHITE);
   g.fillRect(481 , 450 ,1100 ,340);

   g.setColor(Color.BLUE);
   g.fillRect(480, 450, 1100, 320);

   g.setColor(Color.WHITE);
   g.fillRect(481 , 450 ,1100 ,280);
```

```
    g.setColor(Color.BLUE);
    g.fillRect(480, 450, 1100, 260);

    g.setColor(Color.WHITE);
    g.fillRect(481 , 450 ,1100 ,220);

    g.setColor(Color.BLUE);
    g.fillRect(480, 450, 1100, 200);

    g.setColor(Color.WHITE);
    g.fillRect(481 , 450 ,1100 ,160);

    g.setColor(Color.BLUE);
    g.fillRect(480, 450, 1100, 140);

    g.setColor(Color.WHITE);
    g.fillRect(481 , 450 ,1100 ,100);


    g.setColor(Color.WHITE);
    g.fillRect(1350, 450, 50, 1000);

     //GATE
    g.setColor(Color.BLACK);
    g.fillRect(1400, 850, 180, 100);


    // Border
    g.drawRect(480 , 450 ,1100 ,50);
    g.drawRect(480 , 450 ,1100 ,500);

  for(int i = 500; i < 1600 ; i+=50){
   g.drawLine(i , 450 ,i ,1700);
  }
   g.setColor(Color.BLACK);
g.drawLine(479 , 890 ,1400 ,890);

g.drawLine(479 , 550 ,1573 ,550);
g.drawLine(479 , 590 ,1573 ,590);
g.drawLine(479 , 610 ,1573 ,610);
g.drawLine(479 , 650 ,1573 ,650);
g.drawLine(479 , 670 ,1573 ,670);
g.drawLine(479 , 710 ,1573 ,710);
g.drawLine(479 , 730 ,1573 ,730);
g.drawLine(479 , 770 ,1573 ,770);
g.drawLine(479 , 790 ,1573 ,790);
g.drawLine(479 , 830 ,1573 ,830);
g.drawLine(479 , 850 ,1573 ,850);
g.drawLine(479 , 930 ,1573 ,930);
```

```
 g.setColor(Color.WHITE);
g.drawLine(1450 , 890 ,1525 ,890);
g.drawLine(1450 , 890 ,1400 ,930);
g.drawLine(1525 , 890 ,1575 ,930);
g.drawLine(1424 , 910 ,1550 ,910);
g.drawLine(1435 , 900 ,1540 ,900);
g.drawLine(1415 , 920 ,1560 ,920);

  g.setColor(Color.CYAN);
  g.fillArc(1400,815,180,70,0,180);

g.setColor(Color.BLACK);
  g.drawArc(1400,815,180,70,0,180);
  }
}
```

OUTPUT:



RESULT:

21CS2116 – Java Programming Laboratory

7. JDBC (JAVA DATABASE CONNECTIVITY)

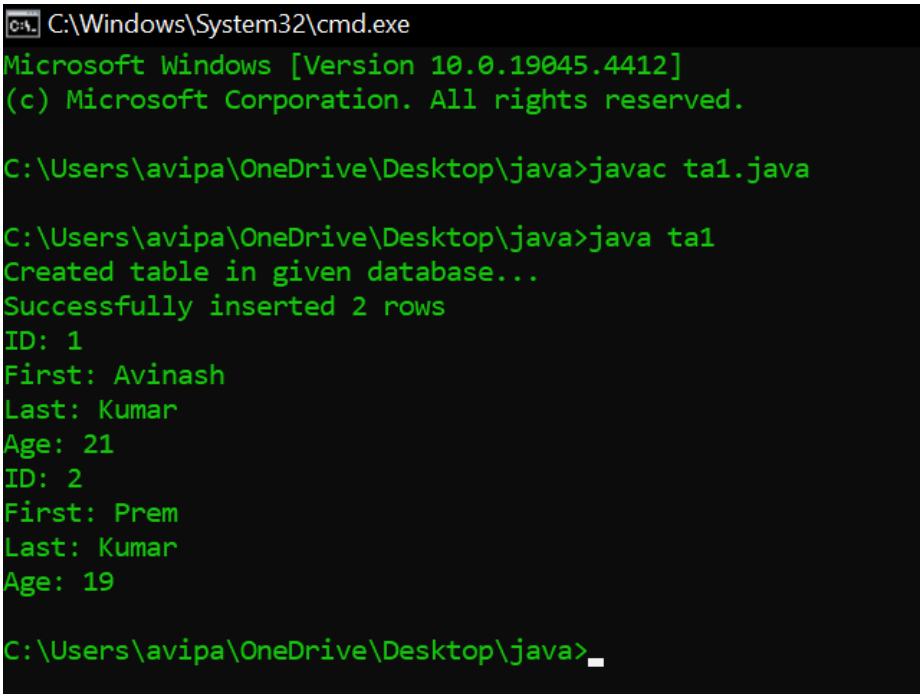
AIM:

To write a program, to understand how to connect, write and execute mysql using java.

PROGRAM:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;
import java.sql.ResultSet;

public class ta1 {
static final String DB_URL = "jdbc:mysql://localhost:3306/company";
static final String USER = "root"; //USERNAME
static final String PASS = "12345"; //PASSWORD
static final String QUERY = "SELECT * FROM Registration";
public static void main(String[] args) {
    try(Connection conn = DriverManager.getConnection(DB_URL, USER, PASS);
      Statement stmt = conn.createStatement();)
      {
       String sql = "CREATE TABLE REGISTRATION " +
              "(id INTEGER not NULL, " +
              " first VARCHAR(255), " +
              " last VARCHAR(255), " +
              " age INTEGER, " +
              " PRIMARY KEY ( id ))";

      stmt.executeUpdate(sql);
      System.out.println("Created table in given database...");
      int result = stmt.executeUpdate( "insert into registration(id,first,last,age)
                        values('1','Avinash','kumar','21'),
                            ('2','Prem','kumar','19')");

      System.out.println("successfully inserted");
      ResultSet rs = stmt.executeQuery(QUERY);
      while(rs.next()){
        System.out.print("ID: " + rs.getInt("id") + "\n");
        System.out.print(", First: " + rs.getString("first") + "\n");
        System.out.println(", Last: " + rs.getString("last")  );
        System.out.print(", Age: " + rs.getInt("age") + "\n" );
      }
        conn.close();
    }
    catch (SQLException e) {
      e.printStackTrace();
    }
  }
}
```

OUTPUT:



RESULT:

8. ENUMERATION, GENERICS AND AUTOBOX

AIM:

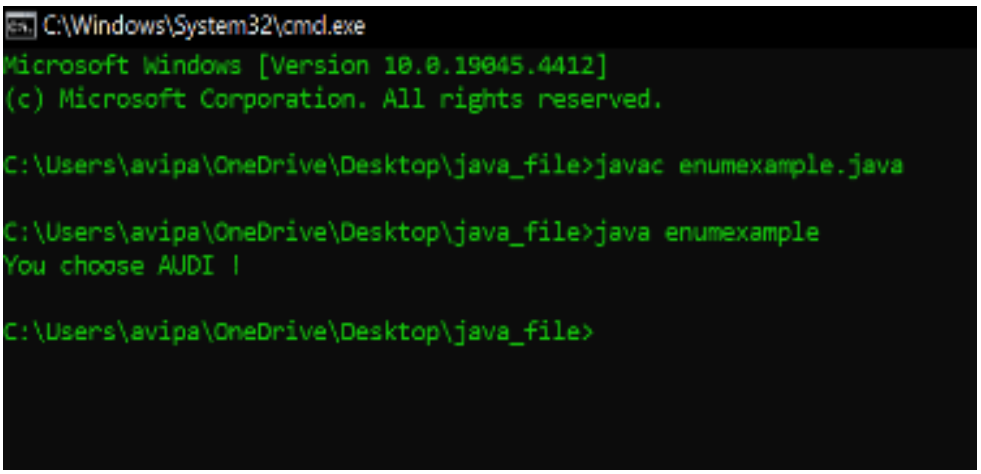
To write a program, to understand the concept of Enumeration, Generics and Autobox in java.

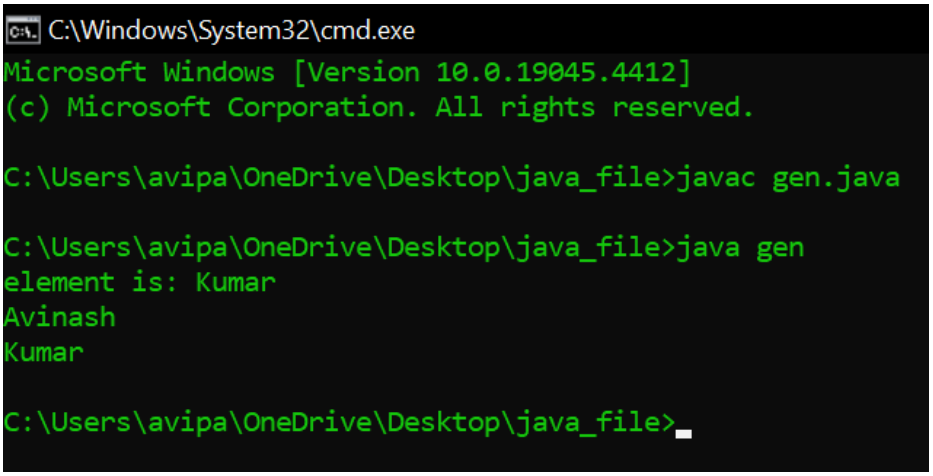
PROGRAM:

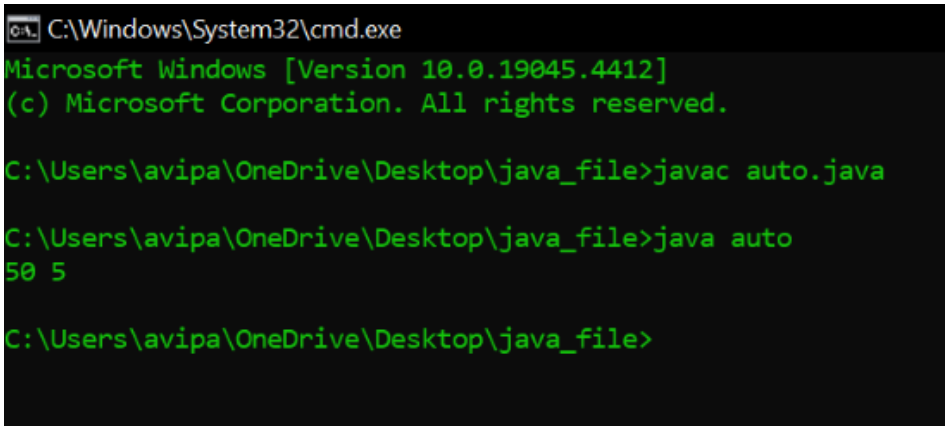
```
// Enumeration
enum Cars {
        BMW,
        JEEP,
        AUDI,
        VOLKSWAGEN,
        NANO,
        FIAT;
}


public class enumexample {
        public static void main(String args[])
        {
                Cars c;
                c = Cars.AUDI;
                switch (c) {
                case BMW:
                        System.out.println("You choose BMW !");
                        break;
                case JEEP:
                        System.out.println("You choose JEEP !");
                        break;
                case AUDI:
                        System.out.println("You choose AUDI !");
                        break;
                case VOLKSWAGEN:
                        System.out.println("You choose VOLKSWAGEN !");
                        break;
                case NANO:
                        System.out.println("You choose NANO !");
                        break;
                case FIAT:
                        System.out.println("You choose FIAT !");

                default:
                        System.out.println("NEW BRAND'S CAR.");
                        break;
                }
        }
}
```

```
// Generics
import java.util.*;
class gen{
public static void main(String args[]){
ArrayList<String> list=new ArrayList<String>();
list.add("Avinash");
list.add("Kumar");
//list.add(32);//compile time error

String s=list.get(1);//type casting is not required
System.out.println("element is: "+s);

Iterator<String> itr=list.iterator();
while(itr.hasNext()){
System.out.println(itr.next());
}
}
}

// Autoboxing
class auto{
public static void main(String args[]){
int a=50;
Integer a2=new Integer(a);
Integer a3=5;
System.out.println(a2+" "+a3);
}
}
```

OUTPUT:




C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.
C:\Users\avipa\OneDrive\Desktop\java_file>javac gen.java
C:\Users\avipa\OneDrive\Desktop\java_file>java gen
element is: Kumar
Avinash
Kumar
C:\Users\avipa\OneDrive\Desktop\java_file>


C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.4412]
(c) Microsoft Corporation. All rights reserved.
C:\Users\avipa\OneDrive\Desktop\java_file>javac auto.java
C:\Users\avipa\OneDrive\Desktop\java_file>java auto
50 5
C:\Users\avipa\OneDrive\Desktop\java_file>

RESULT: