

**National Academy of Science and Technology (NAST)**

Accredited by University Grants Commission (UGC), Nepal (2022)

***(Affiliated to Pokhara University)***

Uttar Behadi-04, Dhangadhi, Kailali

**A   
Lab Report on**

**Object Oriented Programming using Java**

Submitted To

**Department of Computer Application**

**National Academy of Science and Technology**

Submitted By:  
Lila Sagar Ghimire (23530011)

Roll No: 11

Program: BCA

Semester: Third

|  |
| --- |
| Table of Contents |
| [Lab 1 1](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc5626 ) |
| [1.1 WAP to display hello World 1](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc10587 ) |
| [1.2 Wap to Add two number 2](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc27627 ) |
| [Lab 2 3](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc11540 ) |
| [2.1.WAP to demonstrate Data Type 3](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc4675 ) |
| [Lab 3 5](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc24356 ) |
| [3.1.WAP using switch case where user input 1 for addition 2 for subtraction 3 for multiplication 4 for division and if user hit any other option show invalid message. 5](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc24548 ) |
| [Lab 4 8](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc20847 ) |
| [4.1. WAP to illustrate relational operators. 8](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc25502 ) |
| [4.2. WAP to illustrate conditional operator 9](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc29202 ) |
| [4.3.Program to illustrate Ternary Operator and pre and post increment/decrement operator. 10](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc19298 ) |
| [Lab 5 11](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc30299 ) |
| [5.1.WAP to take 10 number input from user and sort in ascending order and display. 11](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc16866 ) |
| [5.2.WAP to take two 3X3 matrix sum on third matrix. 12](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc26219 ) |
| [5.3.WAP to take age of fifty person and display the age from older to younger and count the number of person whose age is greater than or equal to average age. 14](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc24101 ) |
| [Lab 6 15](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc5869 ) |
| [6.1.WAP to take a number input from user and determine either the input number is odd or even. 16](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc23778 ) |
| [6.2.WAP to take age input from user and display eligible for voting if age is greater or equal to 18. 17](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc2444 ) |
| [6.3.WAP using switch case statement when user input a name of day it will display the number of respective number 18](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc22947 ) |
| [6.4.WAP using switch case statement when user input a number from 1 to 7 it will display respective day name. 19](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc12811 ) |
| [6.5. WAP to take input mark of java program and display the respective grade based on PU grading system. 21](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc11769 ) |
| [Lab 7 23](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc19758 ) |
| [7.1.WAP to display number from 1 to input number using for, while and do while loop. 23](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc3909 ) |
| [7.2.WAP to take 10 city name in any array from user and display using for each loop. 24](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc8145 ) |
| [Lab 8 26](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc2022 ) |
| [8.1.WAP to crate two constructor to initialize default value of person name and age and parameterized value from another constructor 26](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc26600 ) |
| [8.2. Create a class Box having width and height as a instance variable initialize default and parameterized value 27](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc3794 ) |
| [8.3.Create a class Box and one method to set value to width and height property and another method to calculate area. 28](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc22068 ) |
| [Lab 9 29](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc22746 ) |
| [9.1. Wap to handle string related functionality using built-in function 29](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc24309 ) |
| [Lab 10 33](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc6443 ) |
| [10.1.WAP to illustrate an example of method overloading 33](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc24805 ) |
| [10.2.WAP that will overload a method area to calculate area of triangle. 34](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc21200 ) |
| [Lab 11 36](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc23864 ) |
| [11.1.WAP to show simple example of inheritance create a class person and a method work. Create subclass employee and a method getSalary and call necessary method 36](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc11066 ) |
| [11.2 Create a class student having two properties name and age. A subclass PayFee which have fee property both function have two functions to set and display data and create a class CollectFee from where student fee can be set and display 37](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc4216 ) |
| [11.3.Write a Java program to create a class known as "BankAccount" with methods called deposit(double amt) and withdraw(double amt). Create a subclass called Savings Account that overrides the withdraw(double amt) method to prevent withdrawals if the account balance falls below one hundred. 39](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc28189 ) |
| [Lab 12 41](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc31613 ) |
| [12.1.WAP to illlustrate Multilevel inheritance 41](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc13605 ) |
| [12.2.WAP to illlustrate Multilevel inheritance 42](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc9414 ) |
| [12.3.WAP to illustrate Hierarchy inheritance 44](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc31304 ) |
| [12.4. Write a Java program to create a vehicle class hierarchy. The base class should be Vehicle,with subclasses Truck, Car and Motorcycle.Each subclass should have properties such as make, model, year, and fuel type. Implement methods for calculating fuel efficiency, distance traveled, and maximum speed. 47](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc24066 ) |
| [Lab 13 50](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc20935 ) |
| [13.1.WAP a program to implement the concept of Polymorphism. Create a class Shape and a method draw and two properties name and color. Create a subclass Square and overwrite draw method. Create another subclass Triangle and overwrite draw method which will implement the concept of plymorphism. When user supply sides of Shape if he/she input three it should set and display name and color of triangle and draw a triangle if user supply four sides it should draw Square, otherwise it should call method of Shape 51](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc30352 ) |
| [13.2.Write down the step to create and use package with eg. 53](file:///D:\My%20Works\Lab%20Report%20from%20Others\Java%20Lab%20Report%20Naresh.docx#_Toc12571 ) |

# Lab 1

## **WAP to display hello World**

public class HelloWorld {

public static void main(String... args){

System.out.println("Hello world");

}

}

**Procedure to Generate Output:**

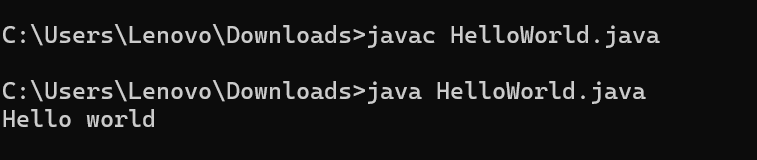
1.Save the file HelloWorld.java

2. Compile the HelloWorld.java file:

**Command**: javac HelloWorld.java

**3**.Run the HelloWorld.class file

command: java HelloWorld

**Output:**

## **Wap to Add two number**

import java.util.Scanner;

public class Add{

public static void main(String... args){

Scanner in=new Scanner(System.in);

System.out.println("enter any two number");

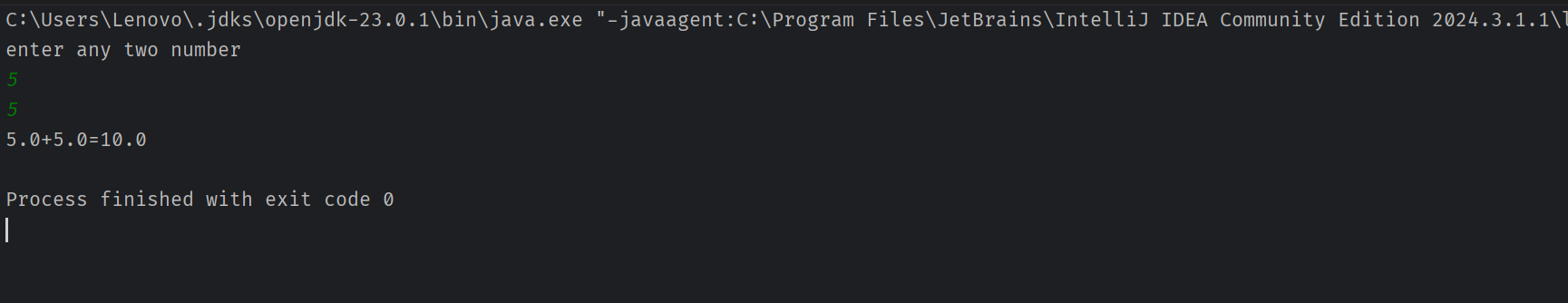
double a=in.nextDouble();

double b=in.nextDouble();

System.out.println(a+"+"+b +"="+(a+b));

}

}

**Output:**

# **Lab 2**

## **2.1.WAP to demonstrate Data Type**

public class DataType {

public static void main(String[] args) {

//byte

System.out.println("Data type: Byte");

byte b='a';

byte byt=67;

System.out.println(b+byt);

System.out.println(Byte.BYTES);

System.out.println(Byte.SIZE);

System.out.println(Byte.MAX\_VALUE);

System.out.println(Byte.MIN\_VALUE);

//char

System.out.println("Data type: Char");

char c='D';

System.out.println(c);

//short

System.out.println("Data type:short");

short st=60;

System.out.println(st);

System.out.println(Short.MAX\_VALUE);

System.out.println(Short.MIN\_VALUE);

System.out.println(Short.BYTES);

System.out.println(Short.SIZE);

//int

System.out.println(Integer.MAX\_VALUE);

System.out.println(Integer.MIN\_VALUE);

System.out.println(Integer.BYTES);

System.out.println(Short.SIZE);

//long

System.out.println("Data type:Long");

System.out.println(Long.MIN\_VALUE);

System.out.println(Long.MAX\_VALUE);

//Float

System.out.println("Data type:Float");

System.out.println(Float.MIN\_VALUE);

System.out.println(Float.MAX\_VALUE);

//Double

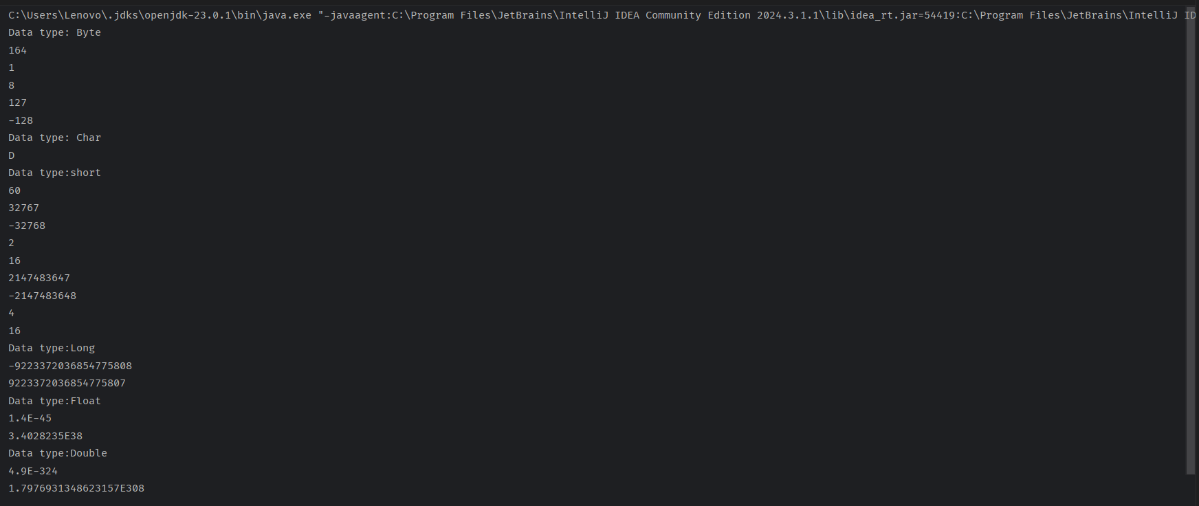
System.out.println("Data type:Double");

System.out.println(Double.MIN\_VALUE);

System.out.println(Double.MAX\_VALUE);

}

}

**Output:**

# **Lab 3**

## **3.1.WAP using switch case where user input 1 for addition 2 for subtraction 3 for multiplication 4 for division and if user hit any other option show invalid message.**

import java.util.Scanner;

public class Variable {

static int firstNumber, secondNumber;

static Scanner in;

public static void main(String[] args) {

in = new Scanner(System.in);

while (true) {

System.out.println("1.Addition");

System.out.println("2.Subtraction");

System.out.println("3.Multiplication");

System.out.println("4.Division");

System.out.println("0.exit");

int choice = in.nextInt();

System.out.println("Enter any two number");

firstNumber = in.nextInt();

secondNumber = in.nextInt();

switch (choice) {

case 1:

add();

break;

case 2:

sub();

break;

case 3:

mul();

break;

case 4:

div();

break;

default:

System.out.println("INVALID");

}

if (choice == 0) {

break;

}

}

}

public static void add() {

System.out.println(“Addition”+(firstNumber + secondNumber));

}

public static void sub() {

System.out.println(“”Subtraction”+(firstNumber - secondNumber));

}

public static void mul() {

System.out.println(“Multiplication”+(firstNumber \* secondNumber));

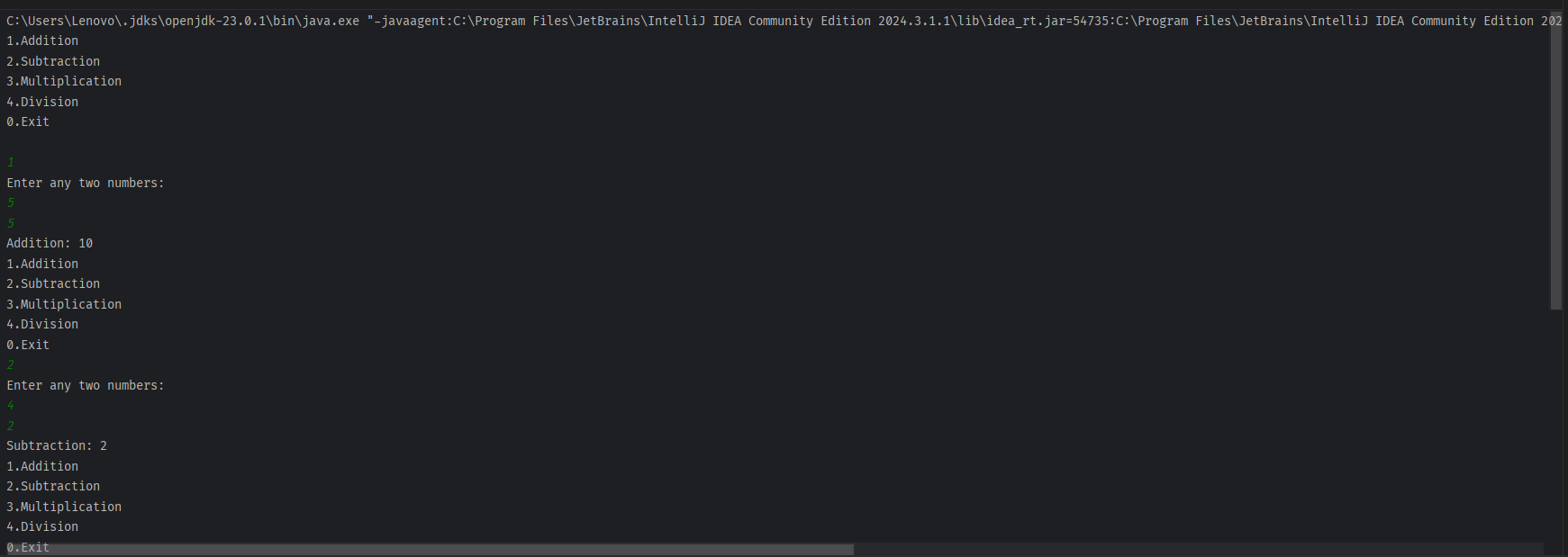
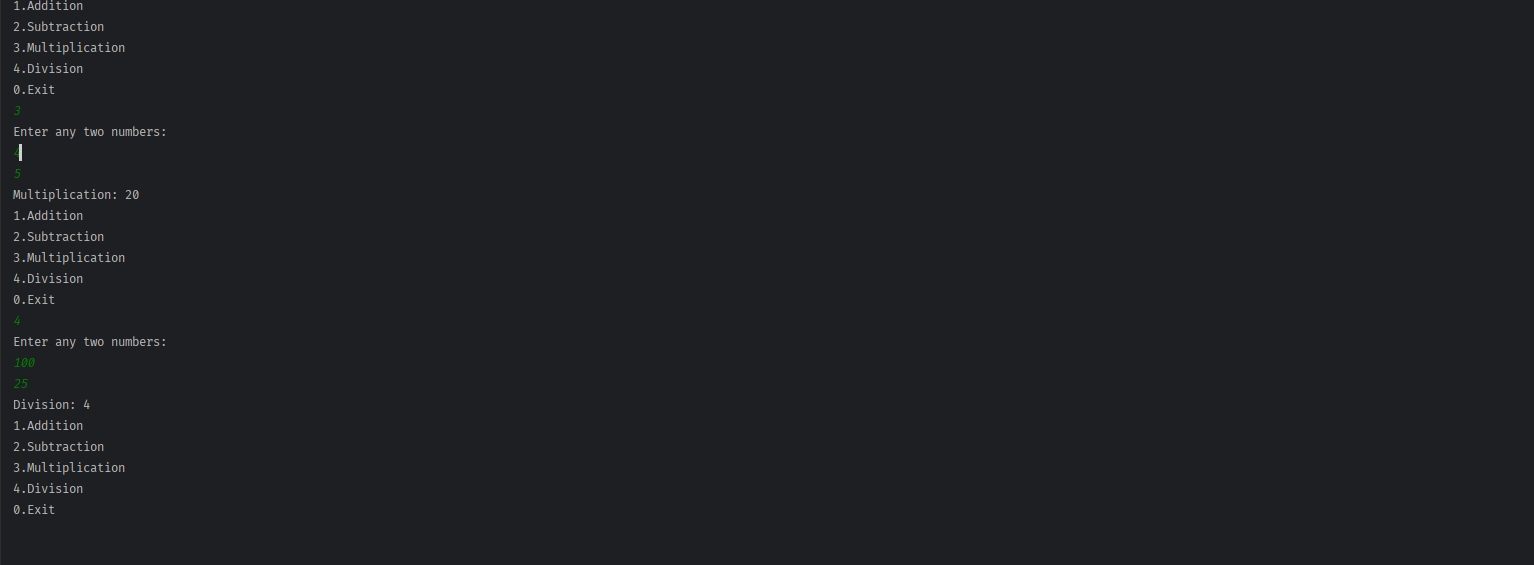
}

public static void div() {

System.out.println(“Division”+(firstNumber / secondNumber));

}

}

**Output**:

# **Lab 4**

## **4.1. WAP to illustrate relational operators.**

import java.util.Scanner;

public class RelationalOperator {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("enter two number");

int first = in.nextInt();

int second = in.nextInt();

boolean r = first > second;

System.out.println(first + " > " + second + ":" + r);

r = first < second;

System.out.println(first + " < " + second + ":" + r);

r = first >= second;

System.out.println(first + " >= " + second + ":" + r);

r = first <= second;

System.out.println(first + " <= " + second + ":" + r);

r = first == second;

System.out.println(first + " == " + second + ":" + r);

r = first != second;

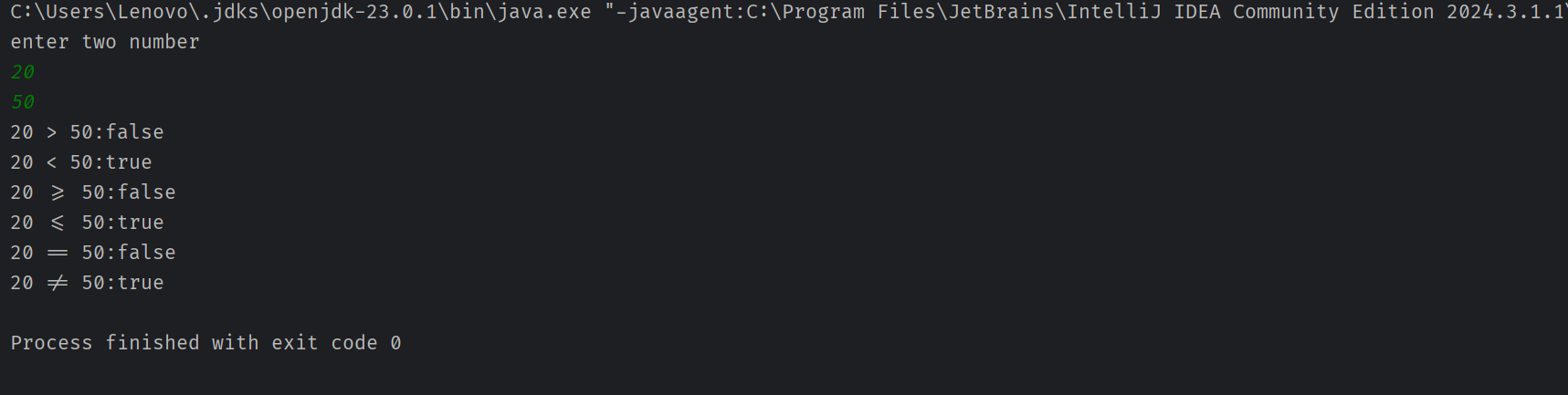
System.out.println(first + " != " + second + ":" + r);

in.close();

}

}

**Output:**



## **4.2**. **WAP to illustrate conditional operator**

import java.util.Scanner;

public class ConditionalOperator {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("enter two number");

int first = in.nextInt();

int second = in.nextInt();

int r = first > second ? first : second;

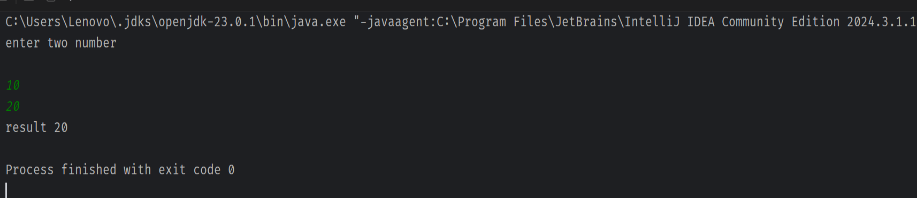
System.out.println("result " + r);

in.close();

}

}

**Output:**



## **4.3.**

## **Program to illustrate Ternary Operator and pre and post increment/decrement operator.**

public class TernaryDemo {

public static void main(String[] args) {

int a, b, c; //a=0,b=0,c=0

a = 10; //a=10

b = ++a; // a=11, b=11

c = a++; // c=11, a=12

System.out.println(a + b + c); //34 (12+11+11)

c++; //c=12

b = --c; // b=11, c=11

a = --b; //a=10, b=10

System.out.println(--a); //8 (9)

b=a>b?a:b; //9>10 b=10

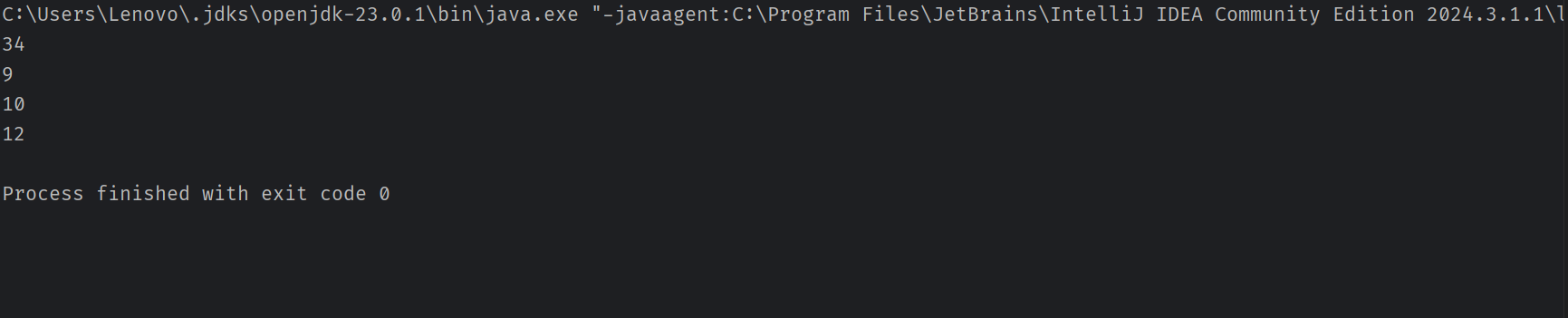
System.out.println(b--); //print 10, b=9

System.out.println(++c); //c=12

}

}

**Output**:



# **Lab 5**

# 

## **5.1. WAP to take 10 number input from user and sort in ascending order and display**

import java.util.Scanner;

public class Ascending {

public static void main(String[] args) {

Scanner in=new Scanner(System.in);

int[] numbers=new int[10];

System.out.println("enter any 10 numbers");

for(int i=0;i<10;i++){

numbers[i]=in.nextInt();

}

int temp;

for(int i = 0; i<10; i++){

for(int j=i+1;j<10;j++){

if(numbers[i]>numbers[j]){

temp=numbers[i];

numbers[i]=numbers[j];

numbers[j]=temp;

}

}

}

System.out.println("In Ascending order");

for(int i=0;i<10;i++){

System.out.println(numbers[i]);

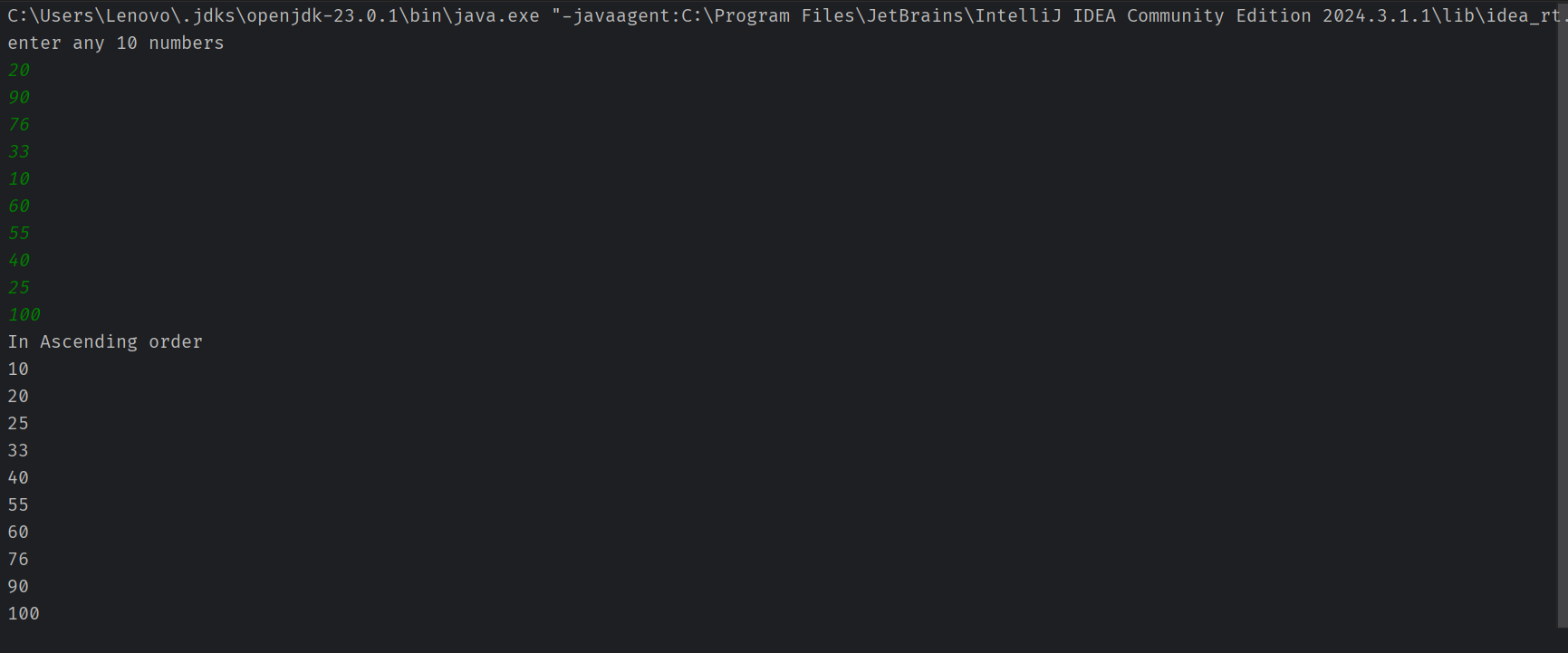
}

in.close();

}

}

output:



## **5.2. WAP to take two 3X3 matrix sum on third matrix**

import java.util.Scanner;

public class MatrixSum {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

int[][] matrix1 = new int[3][3];

int[][] matrix2 = new int[3][3];

int[][] matrixSum = new int[3][3];

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

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

matrix1[i][j] = in.nextInt();

}

}

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

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

matrix2[i][j] = in.nextInt();

}

}

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

matrixSum[i][j] = matrix1[i][j] + matrix2[i][j];

}

}

System.out.println("Sum of matrix");

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

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

}

System.out.println(" ");

}

in.close();

}

}

**Output**:

