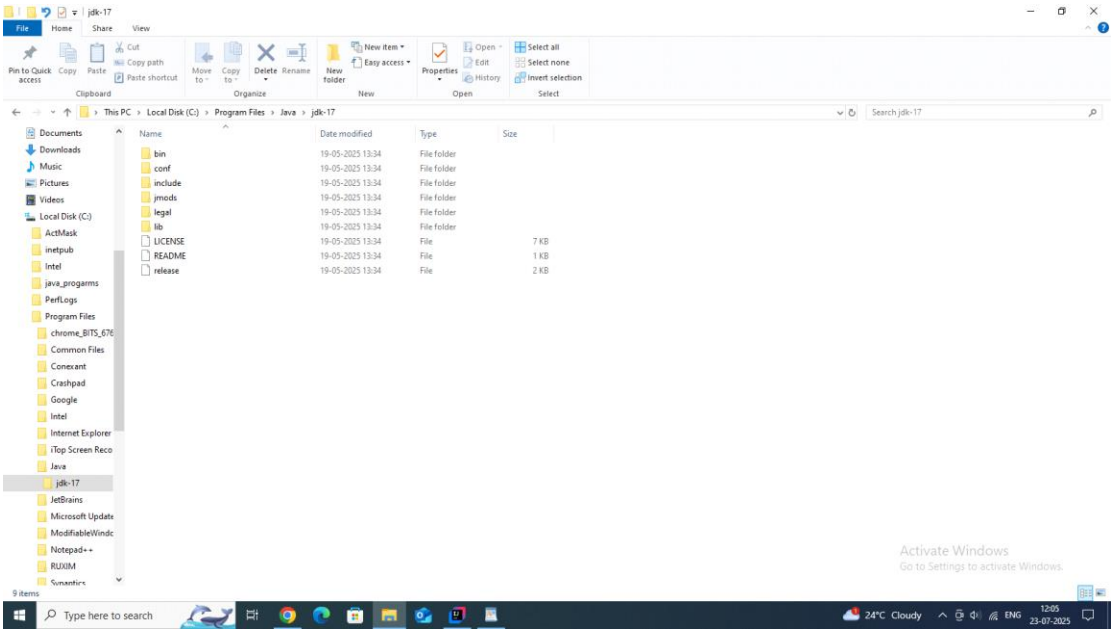
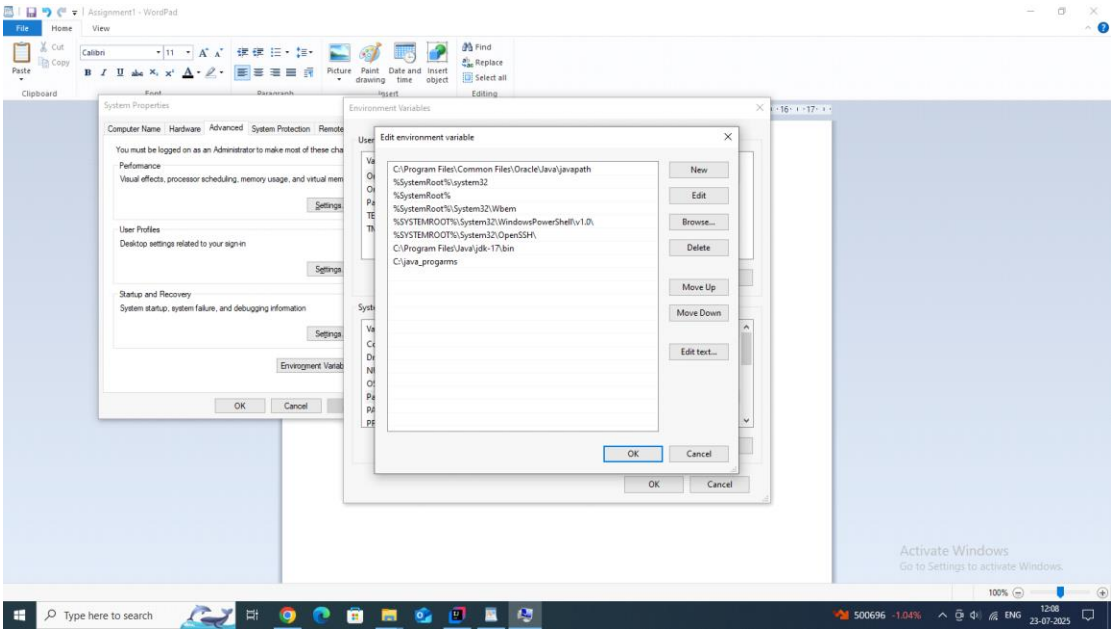
1 Introduction To Java

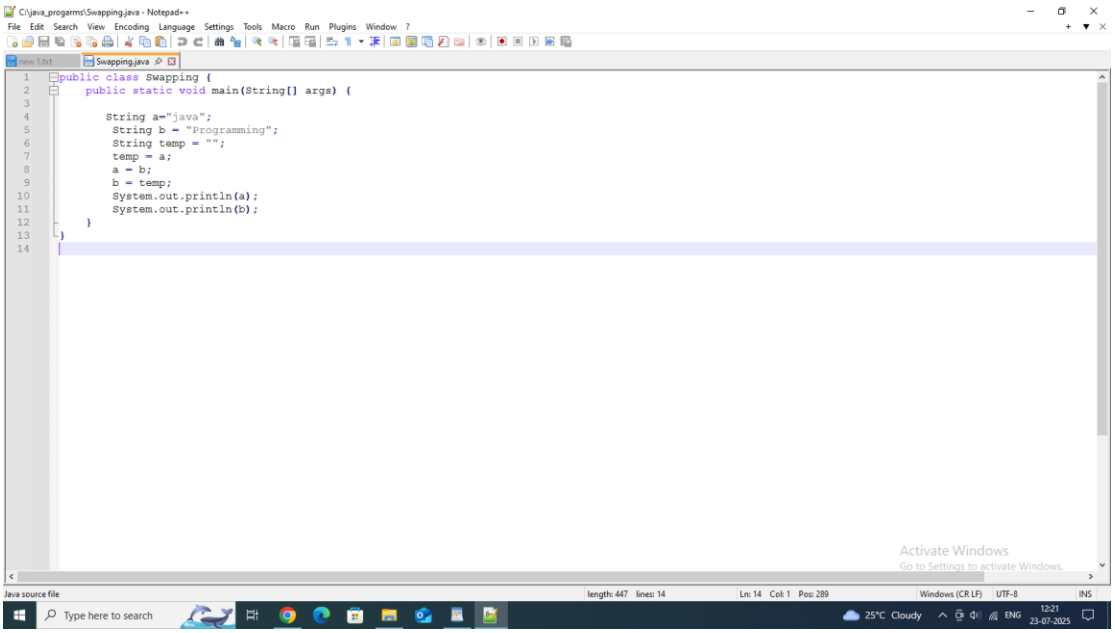
1) Install JDk .



2) Set Environment Variable (path and classpath).



3) Try to write basic program in Notepad++.



2 Java Basic Programming Elements

Ex1: Write Program with one user defined method calling another user defined method.

```
class User_define {
	public static void main(String[] args) {
		m1();	// calling another method
	}
	public static void m1() {
		System.out.println("Method m1() called");
	}
}
```

Ex2: Write Program without main method and execute that program

```
public class Without_main {
	static {
		System.out.println("Executed without main method!");
		System.exit(0); // Terminates the JVM after printing
	}
}
```

Ex3: Write a program with same method in different class.

```
class A {
	public static void m1() {
		System.out.println("A's m1");
	}
}

class B {
	public static void m1() {
		System.out.println("B's m1");
	}
}
```

Ex4: In single Java File write main method in all classes

```
class A {
	public static void main(String[] args) {
		System.out.println("Class A");
```

```
	}
}

class B {
	public static void main(String[] args) {
		System.out.println("Class B");
	}
}
```

EX 5: Write a program to call A main method from B class main method?

```
class A {
	public static void main(String[] args) {
		System.out.println("Inside A main");
	}
}

class B {
	public static void main(String[] args) {
		A.main(args);
	}
}
```

Ex 6: Write program where we pass negative number as size to array

```
public class Test {
	public static void main(String[] args) {
```

```
            int[] arr = new int[-5];
        }
}
```

EX 7: What is the output of below program?

```
public class A
{
    public static void main(String []args)
    {
        System.out.println("A main");
        m1();
    }
    public static void m1()
    {
        System.out.println("A m1()");
        main(new String [0]);
    }
}
```

Output

A main

A m1()

A main

A m1()

A main()

Error.

Ans:

A main

A m1()

A main

A m1()

A main

A m1()

...

StackOverflowError

Ex 8What is the output of the below program?

```
public class A
{
        public static void main(String[]args)
      {
            System.out.println("A main string array");
      }
     public static void main    (String args)
     {
        System.out.println("A main only String");
     }
     public static void main(int []args)
```

```
        {
            System.out.println("A main int array");
        }

    }
```

Output:

A main string array

2.3 Operators

EX:1 Write Program to implement concept

i) Preincrement operator

ii) Postincrement operator

```
public class OperatorsDemo
{
        public static void main(String[]args)
        {
                int x=4;
                int y=++x;
                System.out.println(x);
                System.out.println(y);
                  y=x++;
                  System.out.println(y);//5
                  y=--x;
```

```
            System.out.println(y);//5
        }
}
```

Conclusion:We can apply increment and decrement operators only for variables and not for constant value.

Otherwise we will gate compile time error

Ex:

```
public class OpertorsDemo2
{
        public static void main(String[]args)
        {
                int x:
                int y = --4;    //invalid argument to operation ++/--
                System.out.println(x);
                  y=x--;
                  System.out.println(y);
          }
}
```

******In integer Arithmetic (byte,short,int,long) there is no way to represent infinity. Hence ,if infinity is the result we will get the Arithmetic Exception in the Integral Arithmetic**

Ex:

```
public class OpertaorsDemo
{
       public static void main(String[]args)
```

```
        {
                System.out.println(10/0);
         }
}
```

Output: Exception in thread main "java.lang.ArithmaticException:/by zero

*****But in floating point arithmetic (float and double) there is a way to represent infinity as this float and double class contains following**

1) POSITIVE_INFINITY

2) NEGATIVE_INFINITY

Here we get the result infinity, we don't get any result

```
1) public class OpertaorsDemo2
{
          public static void main(String[]args)
        {
                System.out.println(10.0/0);
          }
}
```

Output : Infinity

```
2) public class OperatorsDemo3
{
                public static void main(String[]args)
```

```
        {
                System.out.println(0.0/0);          }
}
```

Output: NAN

*****String concatenation Operator : '+'**

```
public class OperatorsDemo4
{
    public static void main(String[]args)
   {
        String s="Tdit";
          int b =10;
          int c= 20;
        int d= 30;
        System.out.println(s+b+c);
       System.out.println(b+c+s);
       System.out.println(b+s+c+d);
       }
}
```

output:

Tdit1020

30Tdit

10Tdit2030

***** Equality Operators**

We can apply equality operator for any primitive type including Boolean type:

```
public class OperatorsDemo
{
        public static void main(String[]args)
       {
                System.out.println(10==20);
                System.out.println('a'==97.0);
                System.out.println(10==10.5);
                System.out.println(true==true);
                 System.out.println(false!=true);
        }
}
```

Output:

```
false
true
true
true
true
```