Operators

Increment

Increment and Decrement in Java programming let you easily **add** 1 or **subtract** 1 from variable.

To achieve this, we have two different types of operators.

INCREMENT

In Java, the increment unary operator increases the value of the variable by one.

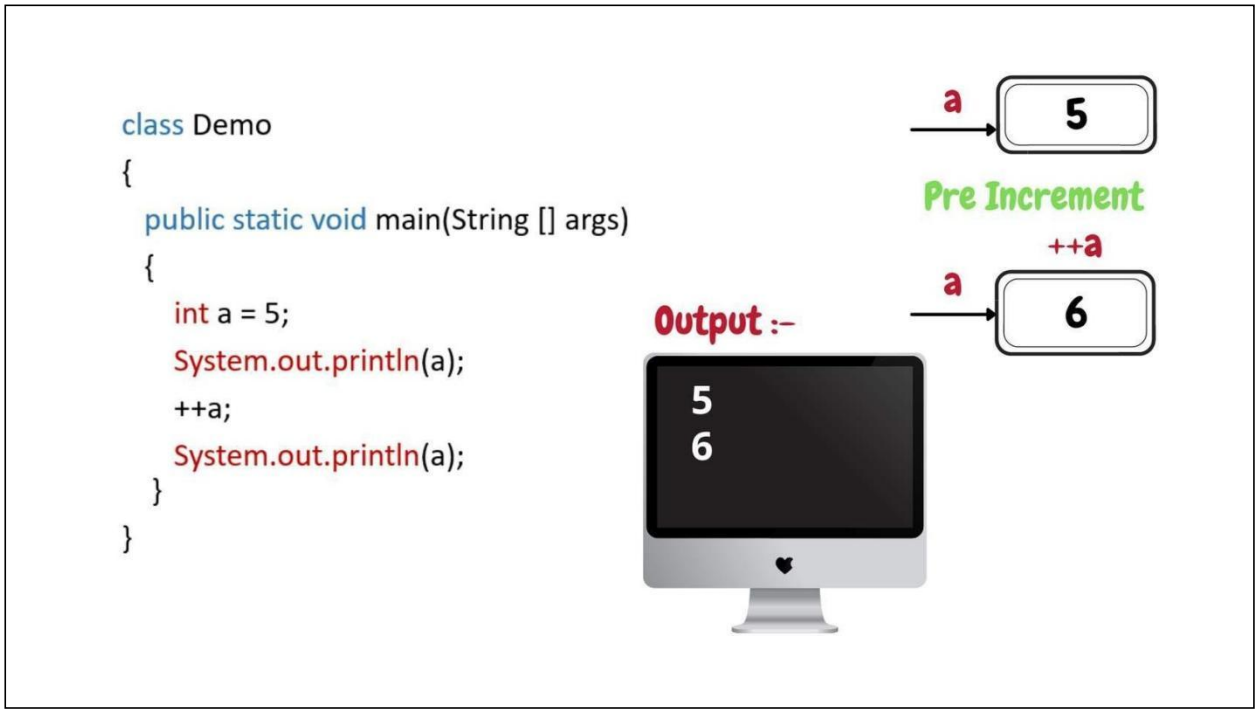
" ++ " is the operator used to increment.

There are two types of Increment

- Pre-Increment.
- Post-Increment.

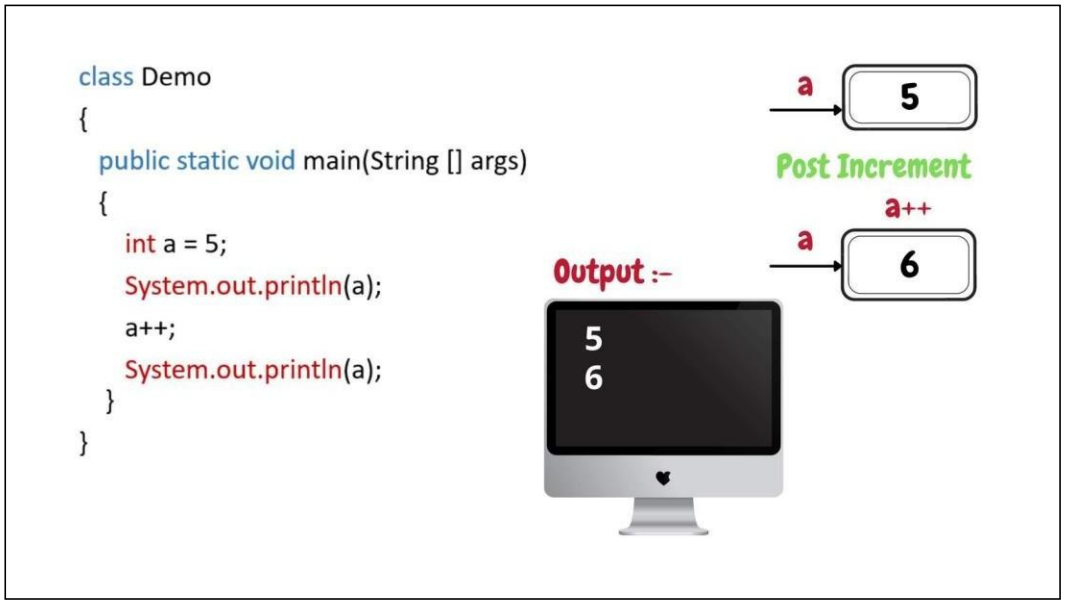
Pre-Increment-

- "++" is written before Variable name.
- Value will be Incremented First and then incremented value is used in expression.



Post-Increment-

- "++" is written after Variable name.
- Value is used in expression first and then gets incremented.



Decrement

In Java, the decrement unary operator decreases the value of the variable by one.

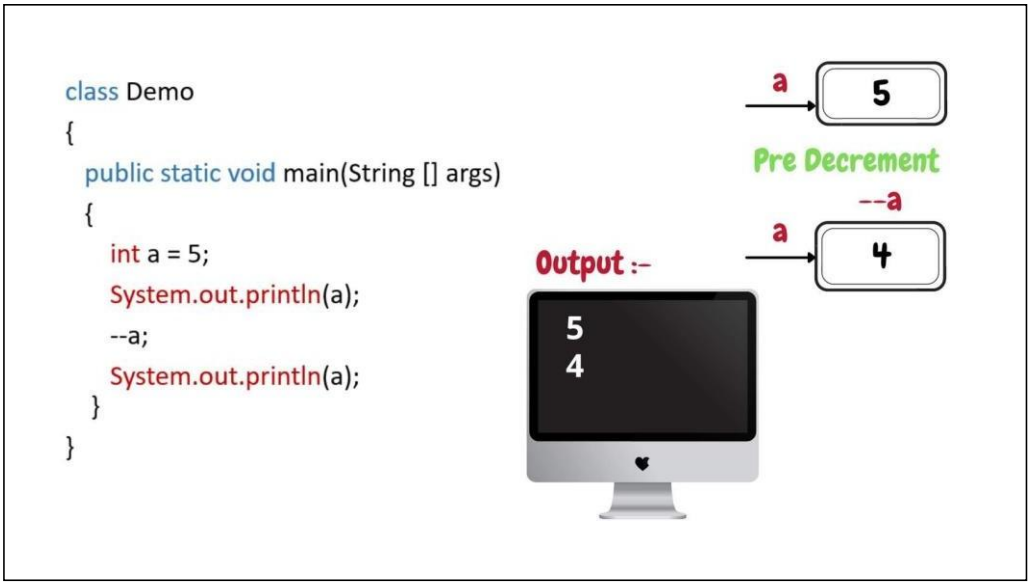
" -- " is the operator used to decrement.

There are two types of Increment

- Post-Decrement.
- Pre-Decrement.

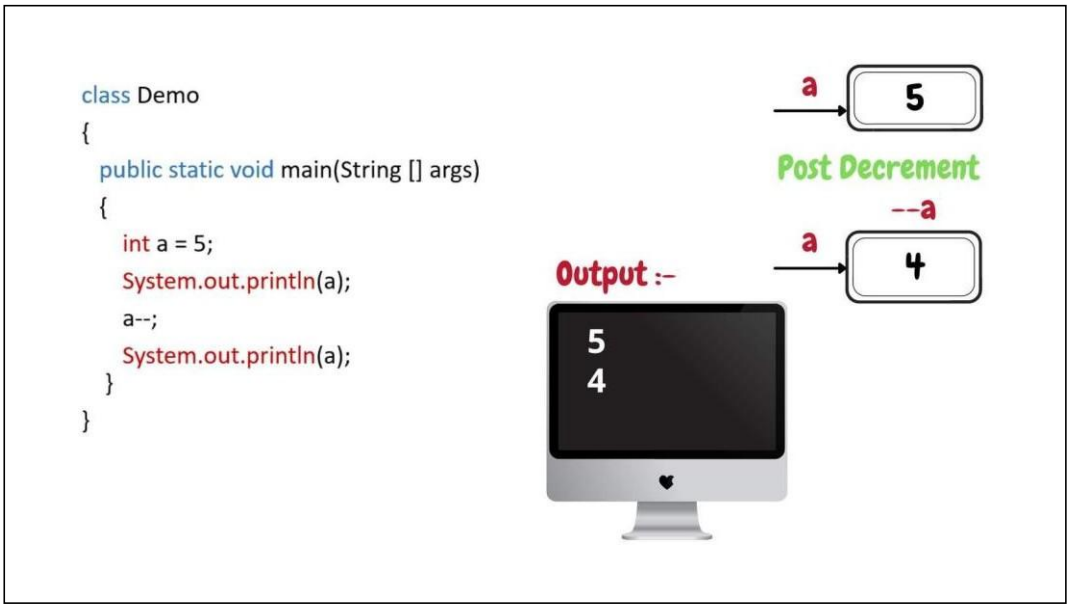
Pre-Decrement-

- "--" is written before Variable name.
- Value is decremented First and then decremented value is used in expression



Post-Decrement-

- "--" is written after Variable name.
- Value is used in expression first and then gets decremented.



Problems on Increment and Decrement.

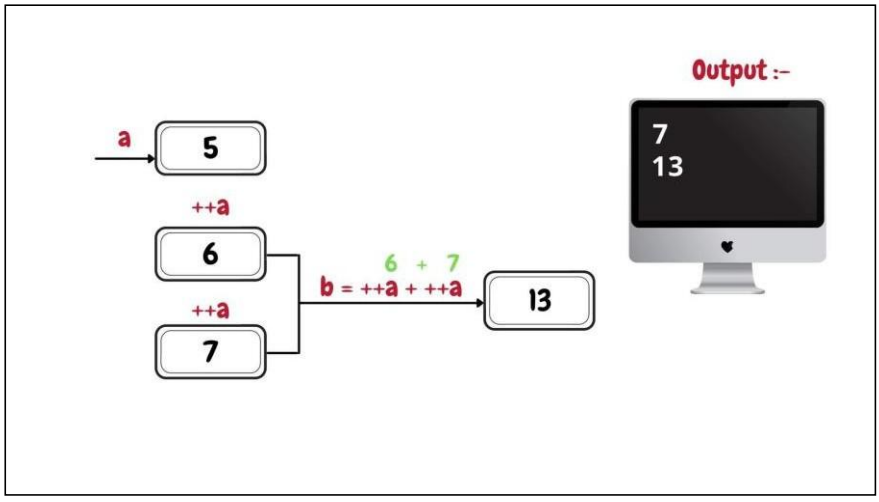
Question 01

Consider a = 5, print a and b

b = ++a + ++a;

Code:

```
class Demo
{
   public static void main (String [] args)
   {            int a = 5 ;
                int b;
                b = ++a + ++a;
                System.out.println(a);
                System.out.println(b);
    }
}
```



Question 02

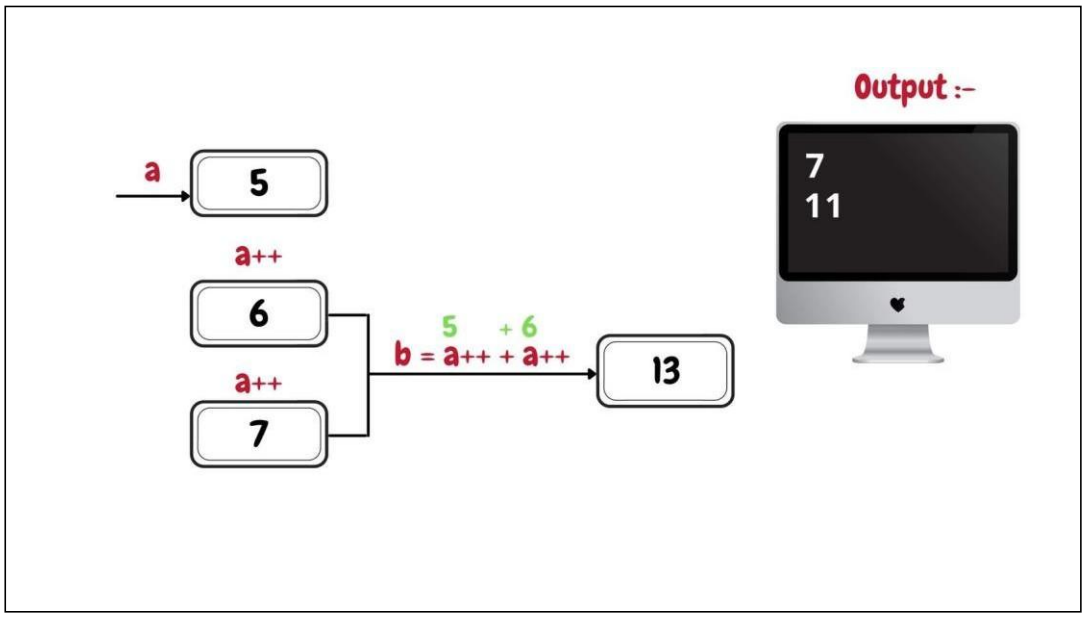
Consider a = 5, print a and b

b = a++ + a++;

Code:

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

```
   {            int a = 5 ;
                int b;
                b = a++ + a++;
                System.out.println(a);
                System.out.println(b);
    }
}
```



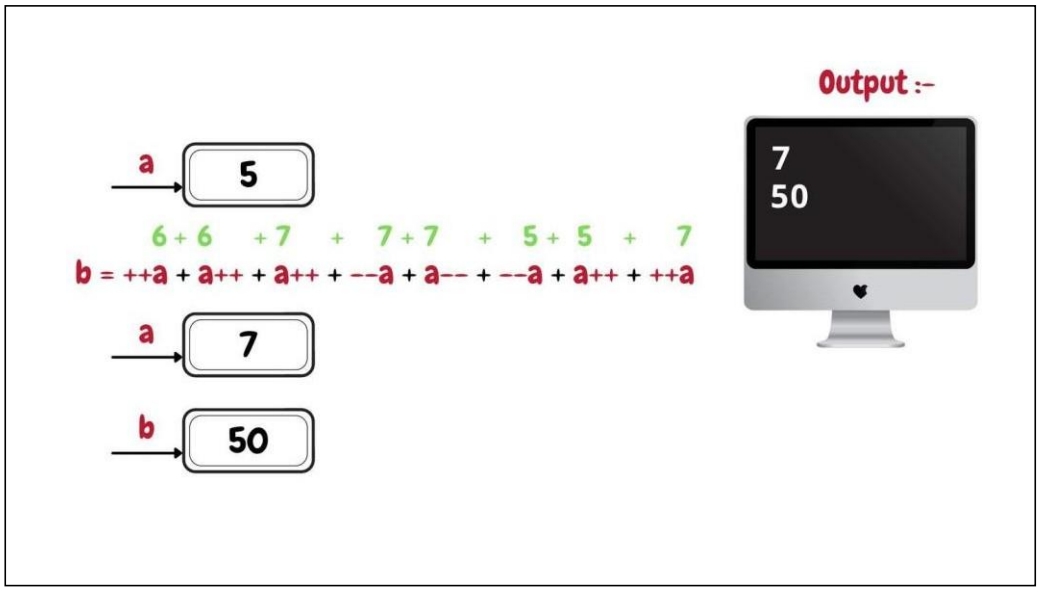
Question 03

Consider a = 5, print a and b

b = ++a + a++ + a++ + --a + a-- + --a + a++ + ++a;

Code:

```
class Demo
{
   public static void main (String [] args)
   {            int a = 5 ;
                int b;
                b = ++a + a++ + a++ + --a + a-- + --a + a++ + ++a;
                System.out.println(a);
                System.out.println(b);
    }
}
```


Output :-
a
5
6 + 6 + 7 + 7 + 7 + 5 + 5 + 7
b = ++a + a++ + a++ + --a + a-- + --a + a++ + ++a
a
7
b
50
7
50

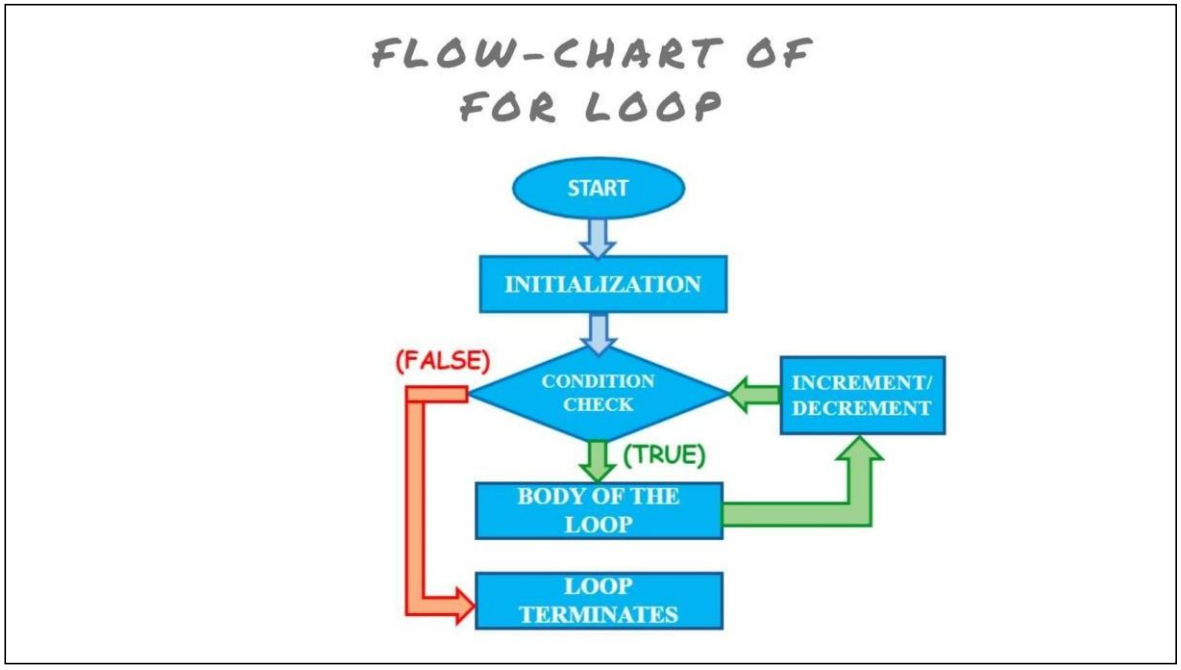
Loops


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

Imagine you have to print "JAVA" 100 times?

Are you going to type the statement 100 times?

Here comes the loop to rescue the programmer and make the job easy. Most fundamental and basic loop is known as FOR loop.



Syntax:

for (initialization; condition; increment/ decrement)

Code:

```
class Demo
{
   public static void main (String [] args)
   {
         int i;
         for ( i=1; i<=5; i++)
         {
            System.out.println("JAVA");
         }
    }
}
```