OPERATORS IN JAVA

Operators as the name suggests are something that operates on some other things to get things done. What do we mean by that? Say '+' is an operator. It will operate upon two numbers and thus it will give us the value of addition.

Here '+' is operator, and the objects it operates upon are the operands and the action it performs is the operation, here it is addition. So we have that in clear.

As programming deals with finding the solution to problems, we often need to do lot's of operation in order to do all of our calculation. So every programming language has lot's of operators to do different operation. So does java. Now the operators are described in various classifications. Here is a rough one not an absoulte type of classification:

- **Arithmetic Operators:** They perform arithmetic functions like addition, subtraction, multiplication etc.
- **Assignment Operators:** They perform the job of assigning one thing to another.
- **Unary Operators:** They are operators that operate upon single operands, we will learn in detail later about what are they.
- **Logical Operators:** They are operators that performs logical operations on operands, we will learn this later in great details.
- **Relational Operators:** They are operators that does the job of comparing different operands, that is how the oprands are related to each other, whether one is larger than the other or not or if one is equal to another or not.
- **Bitwise Operators:** They perfotrm bit by bit operations on numbers and variables
- **Shift Operators:** They perform the job of bit shifting, we will see them in details in a moment.
- **Ternary Operator:** They perform a special form of flow control in a really short way, we will see later.

Now we observe the operators in great details:

Arithmetic Operators

Operator Name	Operator	Example	Description
Addition	+	5 + 6, a + 7	Adds two operands
Subtraction	-	5 - 6, b - 9	Subtracts two operands

Operator Name	Operator	Example	Description
Multiplication	*	3 * 2, u * y	multiplies two operands
Division	/	4 / 2, q / 9	Divides two operands
Modulus	%	9%2, a%b	Gives us the modulus of two operands(details here)

Code Snippet:

```
int a = 1, b = 2, c = 3, d = 4;
//Addition operator
System.out.println(a+b);
System.out.println(2+b);
//Subtraction operator
System.out.println(c-b);
System.out.println(5-b);
//Multiplication operator
System.out.println(a*b);
System.out.println(3*6);
//Division operator
System.out.println(d/b);
System.out.println(6/b);
//Modulus operator
System.out.println(d%b);
System.out.println(d%c);
System.out.println(10%5);
```

Unary Operators

Operator Name	Operator	Example	Description
Increament	++	a++, ++a	a = a+1, Increases an operand by 1
Decreament	--	a--, --a	a = a-1, Decreases an operand by 1
Logical Not	!	!a,	inverts the logical value of an operand

Difference between post fix and prefix explained here

Code Snippet:

```
int x = 10, y = 100, z = 0, p = 200, q = 400, r = 300;
        boolean state = true;

        //Increament
        System.out.println(7++);
        System.out.println(++z);

        // pre-increment operator
        // first x = x+1 then assignment to z, z = a
        z = ++x;
        System.out.println(z);

        // post increment operator
        // first z = p then increaments the value, p = p+1
        z = p++;
        System.out.println(z);


        //Decreament
        System.out.println(7--);
        System.out.println(--z);

        // pre-decrement operator
        // first q = q-1 then z = q
        z = --q;
        System.out.println(z);

        // post-decrement operator
        // first z = r then increament r = r-1
        z = r--;
        System.out.println(z);

        // Logical not operator
        System.out.println(!state);
```

Logical Operators

Operator Name	Operator	Example	Description
Logical And	&&	1&&1 = true, 1&&0 = false	gives us true if both the operands are true
Logical OR	||	1||1 = true, 1||0 = true, 0||0 = false	gives us true if one of the operand is true

Code Snippet

```
int a = 10, b = 9;

System.out.println(1&&1);
System.out.println(1&&0);
System.out.println(0&&0);
System.out.println(a&&b);
System.out.println(1||1);
System.out.println(1||0);
System.out.println(0||0);
System.out.println(a||b);
```

More details about this in flow control such as "if else"

Relational Operators

Operator Name	Operator	Example	Description
Equal To	==	x == y, 1 == 2	gives us true if both the operands are equal to each other
Not Equal To	|=	x != y, 1 != 2	gives us true if both the oprands are not equal to each other
Greater Than	>	x > y, 1 >2	gives us true if left oprand is greater than right
Greater Than or Equal to	>=	x >= y, 1 >=2	gives us true if left oprand is greater than or equal to right
Less Than	<	x < y, 1 <2	gives us true if left oprand is less than right
Less Than or Equal to	<=	x <= y, 1 <=2	gives us true if left oprand is less than or equal to right

Code Snippet

```
int a = 1000, b = 2000;
boolean state = false;
System.out.println(a == b);
System.out.println(1 == 1);
System.out.println(a < b);
System.out.println(a <= b);
System.out.println(a > b);
System.out.println(a >= b);
System.out.println(30 >= 9);
System.out.println(a != b);
System.out.println(!condition);
```

Assignment Operators

Operator Name	Operator	Example	Description
Simple Assignment	=	a = b+1;	assigning values to variables
Addition and assignment	+=	a += b (a = a + b)	shorthand for adding and assigning
Subtraction and assignment	-=	a -= b (a == a - b)	shorthand for subtracting and assigning
Multiplication and assignment	*=	a *= b (a = a * b)	shorthand for multiplying and assigning
Division and assignment	/=	a /= b (a = a / b)	shorthand for dividing and assigning
Modulus and assignment	%=	a %= b (a = a % b)	shorthand for modulus and assigning

Code Snippet

```
int x = 230, y = 110, z = 9, w = 8, p = 99;

// simple assignment operator
d = b;
System.out.println(c);

//previously learned system of assignments
x = x + 1;
y = y - 1;
z = z * 2;
w = w / 2;
p = p % 3;
System.out.println(x + ", " + y + ", " + z + ", " + w + ", " + p);

//reassigning the variables
x = 230, y = 110, z = 9, w = 8, p = 99;

// shorthand assignment operator
x += 1;
y -= 1;
z *= 2;
w /= 2;
p %= 3;
System.out.println(x + ", " + y + ", " + z + ", " + w + ", " + p);
```

The codes are written without mentioning the main class and the main method, copying them in the main method should be enough to operate upon the code.

The three other type of operators will be discussed in upcoming videos.