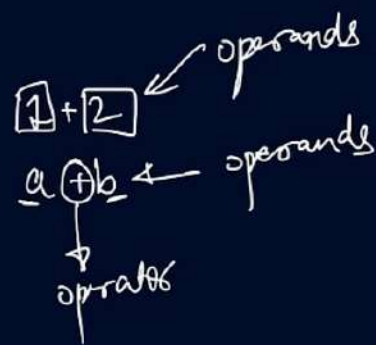


OPERATORS

Symbols that tell compiler to perform some operation



Arithmetic Operators

A = 10, B = 5

Binary

+ ✓ $\rightarrow A + B$ $(10 + 5) \rightarrow 15$

- ✓ $\rightarrow A - B$ $(10 - 5) \rightarrow 5$

* ✓ $\rightarrow 10 * 5$ $(10 * 5) \rightarrow 50$

/ ✓ $\rightarrow (10 / 5) \rightarrow 2$

% \rightarrow modulo $(10 \% 5)$

$$\begin{array}{r} 2 \\ 5 \overline{) 10} \\ \underline{10} \\ 0 \end{array}$$

Unary

++

--

Pre Increment

4^c

++a

```
int a = 10;
```

```
int b = 0;
```

```
b = a++;
```

```
System.out.println(a);
```

```
System.out.println(b);
```

```
int a = 10; }  
int b = 0;
```

```
b = a(++);
```

b = 10 , a = 11

```
System.out.println(a); → 11
```

```
System.out.println(b); → 10
```

Post Increment

पहले $\frac{1}{n}$

① use value
② change ✓

a++

```
int a = 10;
```

```
int b = 0;
```

```
b = ++a;
```

```
System.out.println(a);
```

```
System.out.println(b);
```

a=11 b=11

APN
COLL

Relational Operators

↓ boolean

Operators

$==$ → equal to

$!=$ → not equal to

$>$ → greater

$<$ →

$>=$

$<=$ → $A < B$ → true
→ false

$A = 10$, $B = 5$

$A == B$ → ~~false~~ true

$A > B$ → true
→ false true

$A < B$ → $10 < 5$ → false

$A = 15$ $B = 15$

$A <= B$ → true

$A < B$ → false

Logical Operators

Operators

&& (Logical AND)

|| (Logical OR)

! (Logical NOT)

Bitwise Operators

Operators

A = 0101, B = 0110

& (Binary AND)

<< (Binary Left Shift)

| (Binary OR)

>> (Binary Right Shift)

^ (Binary XOR)

~ (Binary One's Complement)

Bitwise Operators

Operators

→ **&** (Binary AND)

$$\begin{array}{cccc} 0 & 1 & 0 & 1 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 1 & 1 & 0 \\ \hline 0 & 1 & 0 & 0 \\ \hline \end{array}$$

→ **|** (Binary OR)

$$\begin{array}{cccc} 0 & 1 & 0 & 1 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 1 & 1 & 0 \\ \hline 0 & 1 & 1 & 1 \end{array}$$

→ **^** (Binary XOR)

$$\begin{array}{cccc} 0 & 1 & 0 & 1 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 1 & 1 & 0 \\ \hline 0 & 1 & 1 & 1 \\ \hline \end{array}$$

A

0101

~

(Binary One's
Complement)

~A

1010

~

→ <<

(Binary Left Shift)

Number operator position

$A \ll 1$

0101
←

>>

(Binary Right Shift)

$A \gg 1 \rightarrow$ 0101

↓

0010

$B \gg 1 \rightarrow$ 0110