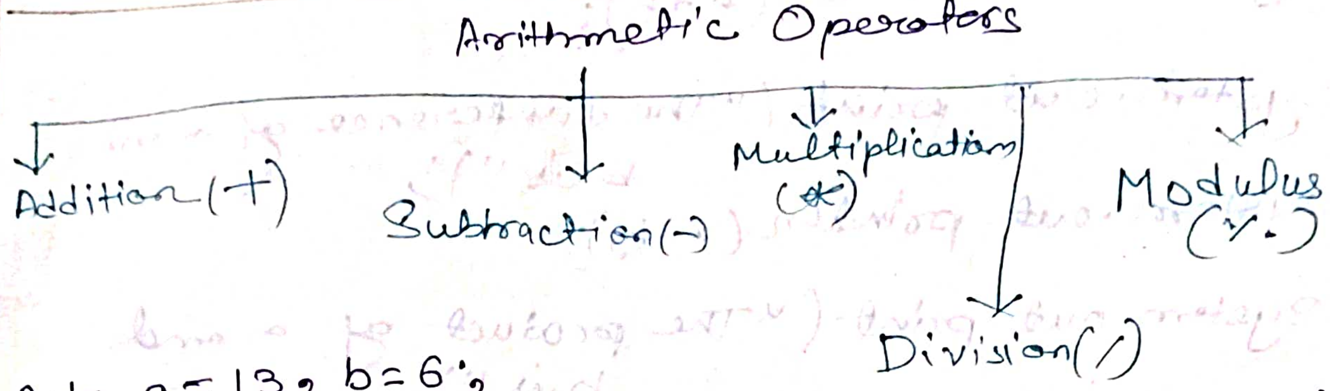
Java Operators

Operand	Operator	Operand	Operator	Result (Operand)
2	+	5	=	7

Arithmetic Operators


Arithmetic Operators

Addition (+)

Subtraction (-)

Multiplication (*)

Division (/)

Modulus (%)

```
int a=13, b=6;
System.out.print("The sum of a and b is");
System.out.println(a+b); //19
System.out.print("The difference of a and b
                  is");
System.out.println(a-b); //7
System.out.print("The product of a and b is")
System.out.println(a*b); 78
System.out.print("The quotient when a is
                  divided by b is");
System.out.println(a/b); // 2
System.out.print("The remainder when a is
                  divided by b is");
System.out.println(a%b); // 1
```

$\frac{int}{int} = int$ int ← transparent water

$\frac{int}{float} = float$ float ← int

$\frac{float}{int} = float$

$\frac{float}{float} = float$

```
int a=13, b=6;
float c=6.0f;
float d = 13.0f;
System.out.print("The sum of a and b is");
System.out.println(a+b);

System.out.print("The difference of a and
                b is");
System.out.println(a-b);
System.out.print("The product of a and
                b is");
System.out.println(a*b);
System.out.print("The quotient when a
        is divided by b is");
System.out.println(a/b);
System.out.println(a/c);
System.out.println(c/a);
System.out.println(c/d);

System.out.print("The remainder when
        a is divided by b is");
System.out.println(a%b);
```

Unary Operators

Unary plus +a

Unary Minus -a $\quad$ $2 \oplus 3$ → binary

Increment Operater ++a $\quad$ +2 $\quad$ -2

Unary

Decrement Operater --a

Logical compement Operator ! !! !A

```
int a= -10;
int b = +10;
booleam c=true;
booleam d = !c;
int e =13;
System.out.println(a); // -10
System.out.println(b); // +10
System.out.println(c); // true
System.out.println(d); // false
System.out.println(e); // 13
System.out.println(++e); //14
System.out.println(e++); // 14
System.out.println(--e); // 14
System.out.println(e--); // 14
System.out.println(e); // 13
```

Rough

15

13

```
booleam y= true;
System.out.println(!y); // false
```

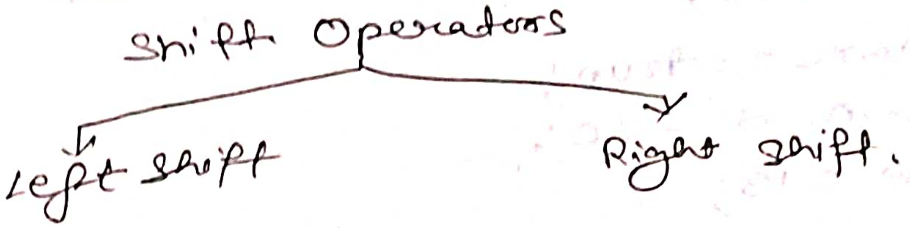
Ternary Operator

(Condition) ? (Expression 1) : (Expression 2)

↑ TRUE part ↑ FALSE part

```
int n1 =5, n2=10, max;
max = (n1 >n2)? n1:n2;
System.out.println("Maximum is = " +
                   max);
                              //10
```

Shift Operators



Right Shift

bits
→ convert it into binary

8>>1

1 0 0 0

0 1 0 0 0 ④

0 0 1 0 0 ②

0 0 0 1 0 ①

divided by 2

a >>b

$$\boxed{\frac{a}{2^b}}$$

Left Shift

3 << 5

3 ⇒ 0 1 1

0 1 1 0 ⑥

1 1 1 0 0 ⑫

a << b

$$\boxed{a \times 2^b}$$

11 → 1 0 1 1 → 5

0 1 0 1

0 0 1 0 → ② ↘ 4

$\frac{n}{2} = 5$

$\frac{5}{2}$

Relational Operators

Operators	Operations
==	Equal to
!=	Not Equal to
>	Greater than
>=	Greater than equal to
<	Less than
<=	Less than equal to

```
int a=5, b=4;
boolean c= a>b;
System.out.println(c);
```

Logical Operators

Logical Operator	Java Operator
AND	&&
OR	||
NOT	!

AND

a	b	a.b
0	0	0
0	1	0
1	0	0
1	1	1

OR

a	b	a+b
0	0	0
0	1	1
1	0	1
1	1	1

NOT

a	a^c
0	1
1	0

1)
```
Start
Integer a=1;
 Integer b=1;
  Integer c=a || --b;
  Integer d =--a && --b;
   Point  a b c d
 End
```

	a	b	c	d
	~~1~~	1	1	0
As →	0	1	1	0

6

$c = a \;||\; --b$

$= \textcircled{1} \;||\; \textcircled{X}$

$d = --a \&\& --b;$

$= \textcircled{0} \;\&\&\; X$

②
```
Start
Integer a = 1;
Integer b = 1;
Integer c = a || --b;
Integer d = a-- && --b;
Print a b c d
End
```

a	b	c	d
~~1~~	1	1	0
0	~~1~~ 0	1	0

~~ro~~=

c = a || -- b
= 1

d = a--&& --b;
= 1 && 0
= 0 → 0

As → 0 0 1 0 → 2.

boolean a = true;
System.out.println(!a); // false

Bitwise Operators

8 4 2 1

Operator	Name	Example	Result
&	Bitwise AND	6&3	2
|	Bitwise OR	10|10	10
^	Bitwise XOR	2^2	0
<<	Left - shift	10<<2	40
>>	Right - shift	10>>2	2

6&3

6 → 0 1 1 0
3 → 0 0 1 1
& → 0 0 1 0 → 2.

A	B	A^B
0	0	0
0	1	1
1	0	1
1	1	0

10 || 10

10 → 1 0 1 0
10 → 1 0 1 0
|| 1 0 1 0 → 10

10 << 2

10×2^2
$= 40$

10>>2

$\frac{10}{2^2} = \frac{\cancel{10}\,5}{\cancel{4}\,2}$

= ②.⑤ → ~~⑤~~
↓
2

```
System.out.println(6&3);
System.out.println(10||10);
System.out.println(2^2);
System.out.println(10<<2);
System.out.println(10>>2);
```

Assignment Operators

a = 5; assignment

a == 5
relational

$a =+ 5 \longrightarrow a = a+5$

2 operator   3 operand

age

age = age+1 ⟶ age =+ 1

1 Jan 2018 }
1 Jan 2019

Operator	Example	Equivalent Expression
=	m = 10	m = 10
+=	m += 10	m = m+10
-=	m -= 10	m = m - 10
*=	m *= 10	m = m*10
/=	m /= 10	m = m / 10
%=	m %= 10	m = m % 10
<<=	a <<= b	a = a<<b
>>=	a >>= b	a = a>>b
>>>=	a >>>= b	a = a>>>b
&=	a &= b	a = a&b
^=	a ^= b	a = a^b
|=	a |= b	a = a|b

Java Operator Priority & Associativity

2 + $\underline{3 * 4}$ → priority high

2 + 12 → 14

2 + 4 - 3 → Associativity left to right because + and - both are same priority

6 - 3 → 3

Operator	Description	Associativi
( )	Parentheses or function call	left to right
[ ]	Brackets or array subscipt	
.	Dot member selection Operator	
->	Arrow Operator	
++ --	Postfix increment/decrement	
++ --	Prefix increment/decrement	right to left
+ -	Unary plus and minus	
! ~	not operator and bitwise ~~operator~~ complement	
(type)	type cast	
&	Address of operator	
sizeof	Determine size in bytes	
* / %	Multiplication, division and modulus	lefto to right
+ -	Addition and subtraction	left to rig
<< >>	Bitwise left shift and right shift	left to rig
< <=	relational less than/less than equal to	left to right
> >=	relational greater than/greater than or equal to	

== .!=	Relational equal to or not equal to	L to R
&&	Bitwise AND	LtoR
^	Bitwise exclusive OR	LtoR
|	Bitwise inclusive OR	L toR
&&	Logical AND	L to R
||	Logical OR	L toR
? :	Ternary operator	R to L
=	Assignment Operator	R to L
+= -=	Addition/subtraction assignment	
*= /=	Multiplication/division assignment	
%= &=	Modulus and bitwise assignment	
^= |=	Bitwise exclusive/inclusive OR assignment	
<<= >>=		
,	Comma operator	L to R

PUMA'S REBL TAC

AUT

3