Operand == on which we operate .

Operator == Who does the operation

Int num1 = 10, num2 = 5, num3 = 35;

Int result = num1 + num2;

num1 , num2 🡺 Operand | + 🡺 Operator

AND &

0 0 === 0

0 1 === 0

1 0 === 0

1 1 === 1

OR

0 0 === 0

0 1 === 1

1 0 === 1

1 1 === 1

int num = 10;

num = num + 1;

num++;

10 | 20 |2

20

(num1 > num2) && (num1 > num3)

false ==== num1 is not the largest number

(num2 > num1) ^ (num2 > num3) | ()

True true ==== num2 is the largest number

1. Unary Operator: require only one operand.
   1. Postfix: expr++, expr--;
   2. Prefix: ++expr, --expr;
2. Arithmetic Operator: addition, deletion, multiplication, division, modulation.
   1. + : num1 + num2;
   2. - : num1 - num2;
   3. \* : num1 \* num2;
   4. / : num1 / num2;
   5. % : num1 % num2;
3. Relational Operator:
   1. Comparison: <, >, <=, >= , instanceof.
   2. Equality: ==, !=
4. Bitwise :
   1. AND : &
   2. Exclusive OR: ^ (Shift + ^)
   3. Inclusive OR : | (Shift + \)
5. Logical:
   1. AND: &&
   2. OR: ||
6. Ternary:
   1. ? :
      1. (num1 > num2) ? num1 : num2;
      2. Weight <=500gm ? 600 : 1000;
7. Assignment:
   1. =
   2. += :
      1. num=20;
      2. num += 10;
      3. num = num + 10; == 30
      4. num -=10; 10
      5. num\*= 10 ;