

---

# ОПЕРАТОРЫ JAVA

---

# ОПЕРАЦИИ

- Арифметические `+, -, *, /, %, ++, --`
- Битовые `~, &, |, ^, <<, >>, >>>`
- Логические `!, &&, ||`
- Сравнения `<, <=, >, >=, ==, !=`
- Условный оператор `condition ? ifTrue : ifFalse`
- Присваивания `=, +=, -=, /=, *=, %=, ~=, &=, |=, ^=, <<=, >>=, >>>=`
- Вызов метода `method()`
- Обращение к полю `object.field`
- Создание объекта `new MyClass()`
- Приведение типа `(type)`
- Ссылка на метод `object::method`
- Lambda-выражение `(a, b) -> { ... }`

# ПРИОРИТЕТ И АССОЦИАТИВНОСТЬ

Приоритет	Ассоциативность	Операции
1	слева направо	() [] .
2	справа налево	~ ! ++ -- new (cast) + (унарный) - (унарный)
3	слева направо	* / %
4	слева направо	+ -
5	слева направо	>>> >> <<
6	слева направо	> >= < <=
7	слева направо	== !=
8	слева направо	&
9	слева направо	^
10	слева направо	
11	слева направо	&&
12	слева направо	
13	слева направо	?:
14	справа налево	= op=

# ИНКРЕМЕНТ И ДЕКРЕМЕНТ

Инкремент (++) увеличивает значение переменной на 1.

Декремент (--) уменьшает значение переменной на 1.

Инфиксная форма возвращает значение переменной после операции.

Постфиксная форма возвращает значение переменной до операции.

```
int a = 5;  
int b = a++; // b == 5, a == 6  
int c = ++a; // c == 7, a == 7  
  
int d = c--; // d == 7, c == 6  
int e = --c; // e == 5, c == 5
```

# ДВОИЧНОЕ ПРЕДСТАВЛЕНИЕ ЧИСЕЛ

десятичное число (byte)	двоичное число (byte)
1	00000001
2	00000010
3	00000011
4	00000100
8	00001000
127	01111111
-1	11111111
-2	11111110
-4	11111100
-128	10000000

# БИТОВЫЕ ОПЕРАЦИИ

A	B	$\sim B$	A & B	A   B	A ^ B
0000	0000	1111	0000	0000	0000
1111	1111	0000	1111	1111	0000
0000	1111	0000	0000	1111	1111
0001	1100	0011	0000	1101	1101
1000	1001	0110	1000	1001	0001
0101	1100	0011	0100	1101	1001
0110	1001	0110	0000	1111	1111

# БИТОВЫЕ ОПЕРАЦИИ

A	B	A << B	A >> B	A >>> B
0000	2	0000	0000	0000
0011	1	0110	0001	0001
1000	1	0000	1100	0100
0110	2	1000	0001	0001
1010	2	1000	1110	0010
1111	3	1000	1111	0001
1001	1	0010	1100	0100