

# Оператор if. Области видимости

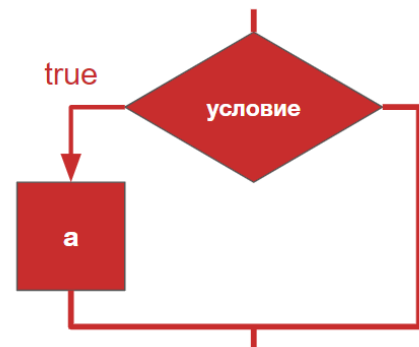
## if, if-else, if-else-if

Оператор `if` в Java - это основная конструкция для выполнения условной логики. Он позволяет программе выбирать путь выполнения на основе истинности или ложности логического выражения. Есть несколько вариантов использования оператора `if`

### Одиночный if

Это самая простая форма, где код внутри блока `if` выполняется только если условие истинно.

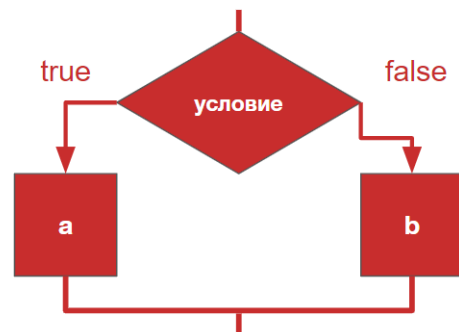
```
...  
if (условие) {  
    // блок кода a  
}  
...
```



### If-else

Если условие в `if` истинно, выполняется первый блок кода; если ложно - второй блок кода в `else`.

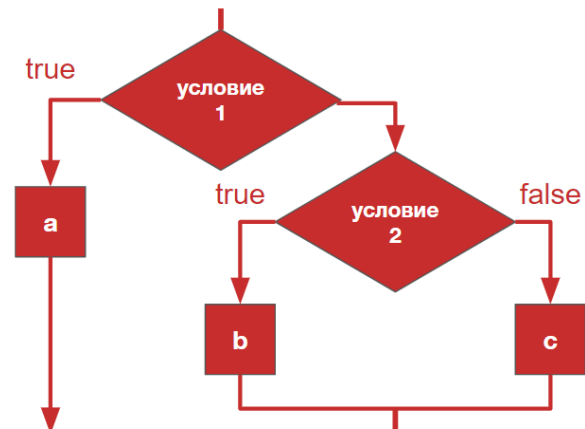
```
...  
if (условие) {  
    // блок кода a  
} else {  
    // блок кода b  
}  
...
```



## If-else-if

Используется для проверки нескольких условий последовательно. Если первое условие ложно, проверяется следующее, и так далее. Если какое-либо условие истинно, выполняется соответствующий блок кода, и остальные условия не проверяются.

```
...  
if (условие 1) {  
    // блок кода a  
} else if (условие 2) {  
    // блок кода b  
} else {  
    // блок кода c  
}  
...
```



## Области видимости

**Область видимости** в Java определяет, где переменные и методы доступны для использования в вашей программе. Область видимости ограничена блоками кода, такими как тела методов, циклы и блоки условных операторов.

```
public class ScopeExample {  
    public static void main(String[] args) {  
        int x = 5;  
  
        if (x > 0) {  
            // Эта переменная доступна только внутри этого блока if  
            int y = 10;  
  
            if (x == 5) {  
                // переменная z доступна только в этом вложенном блоке if  
                int z = 15;  
                System.out.println("x равно 5, y равно " + y + ", z равно " +  
z);  
            }  
            // Здесь z уже не доступна, но y доступна  
            // System.out.println(z); // Это вызовет ошибку компиляции  
            System.out.println("x больше 0, y равно " + y);  
        }  
        // Здесь y и z уже не доступны  
        // System.out.println(y); // Это вызовет ошибку компиляции  
    }  
}
```

```
// System.out.println(z); // Это тоже вызовет ошибку компиляции

// переменная x доступна во всем методе main
System.out.println("x равно " + x);
}
}
```

## Тернарный оператор

**Тернарный оператор** в Java — это сокращенная форма условного оператора, которая позволяет присвоить переменной одно из двух значений в зависимости от результата логического выражения. Тернарный оператор обозначается символами **?** и **:**

Структура тернарного оператора:

переменная = условие ? значение1 : значение2;

Здесь условие — это логическое выражение, которое оценивается как `true` или `false`. Если условие истинно (`true`), тернарный оператор возвращает **значение1**, иначе — **значение2**.

Пример использования тернарного оператора:

```
int a = 5;
int b = 10;
int max;

// Использование тернарного оператора для определения максимального значения
max = (a > b) ? a : b;

System.out.println("Максимальное значение: " + max);
```

В этом примере тернарный оператор используется для определения максимального из двух чисел. Если `a > b` истинно, то `max` присваивается значение `a`, в противном случае — значение `b`.

Тернарный оператор удобен для использования в простых условиях, когда необходимо выбрать одно из двух значений. Он помогает сократить количество кода по сравнению с использованием стандартных условных конструкций `if-else`.