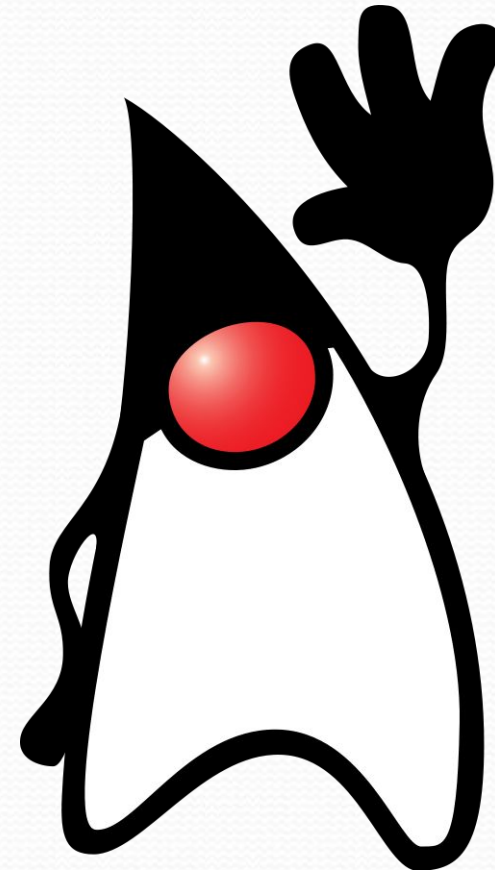


# Курс: «Программирование на Java»



Java™

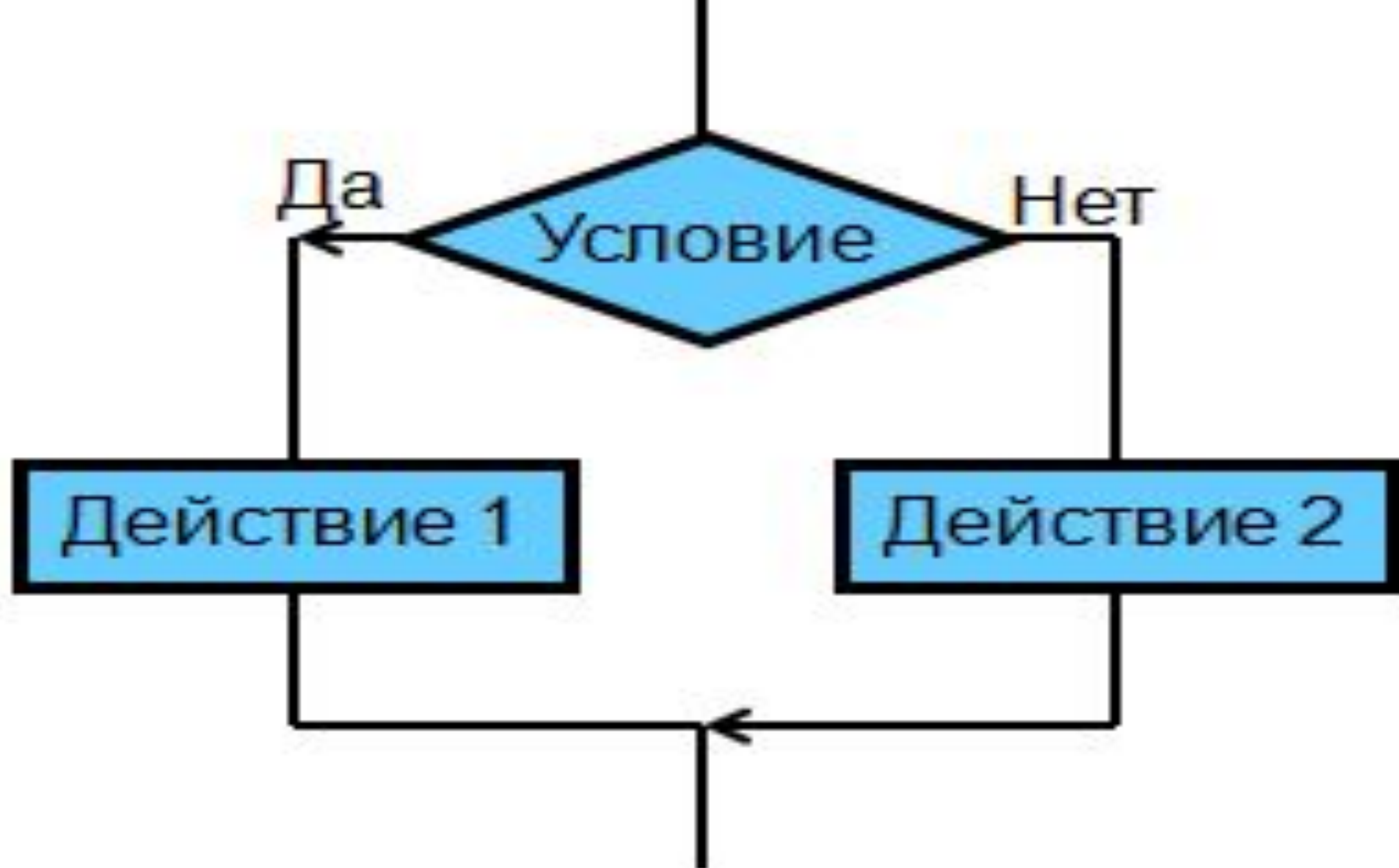




- Рогаткин  
Роман  
Олегович
- [vk.com/teacher\\_1cb](https://vk.com/teacher_1cb)
- [teacher.1cb@yandex.ru](mailto:teacher.1cb@yandex.ru)
- +7 (926) 151-39-25 (Мегафон)

## Условные конструкции (ветвления)

- **Условная конструкция (ветвление)** — алгоритмическая конструкция, в которой, в зависимости от результата проверки условия («истина» или «ложь»), предусмотрен **выбор одной из двух последовательностей действий (ветвей)**.
- Команда ветвления записывается так:
- ```
Если (Условие)           if(Условие){  
(Если условие правдиво)  То<Действие1>           Действие 1  
(Если условие ложно)  Иначе<Действие2>       } else {  
                           Действие 2  
                           }
```



# Операторы сравнения в условных конструкциях

- « > » - оператор «**больше?**» Проверяет является ли правый операнд больше левого операнда.
- « < » - оператор «**меньше?**» Проверяет является ли правый операнд меньше левого операнда.
- « == » - оператор «**равно?**». Проверяет равны ли правый операнд и левый операнд.
- « != » - оператор «**не равно?**» Проверяет не равны ли правый операнд и левый операнд.
- « >= » - оператор «**больше либо равно?**» Проверяет является ли правый операнд больше левого операнда или равен ему.
- « <= » - оператор «**меньше либо равно?**» Проверяет является ли левый операнд меньше правого операнда или равен ему.

## Условные конструкции : Объявление

- Чтобы начать работу с условной конструкцией, ее необходимо объявить. **Объявление** начинается с ключевого слова **if**.
- После ключевого слова необходимо записать условие-вопрос в круглых скобках, которое мы хотим проверить с помощью операторов сравнения. **if(5 >= 3)**
- Потом необходимо поставить фигурные скобки и внутри фигурных скобок дать действие-ответ(тело условия) на поставленное условие.  

```
if(5 >= 3){ // условие-вопрос 5 >= 3 ?  
    System.out.println("Условие верно"); // ответ на условие  
}
```

## Условные конструкции : Ветка Else (Иначе)

- Иногда, встречаются условия, проверив которые мы можем получить при проверке **ложь**. Для таких случаев придумали ветку **Иначе**, которая выполнятся тогда и только тогда, когда условие **ложно**.
- Ветка **Иначе** пишется сразу **после** закрывающей фигурной скобки **if** и начинается с ключевого слова **else** (В переводе – иначе).
- После ключевого слова ставятся фигурные скобки внутри которых пишется действие-ответ, если условие ложно.
- **!!! ВАЖНО** – ветку **Иначе(else)** не обязательно добавлять, порой достаточно только **if**.

## Пример использования else

- ```
if(5 < 3){ // условие-вопрос 5 < 3 ?  
    System.out.println("Условие верно"); // ответ на условие  
} else { // начало ветки ИНАЧЕ  
    System.out.println("Условие не верно"); //ответ ИНАЧЕ  
}
```



## Пример итоговой программы.

```
public class Proj2 {  
    public static void main(String[] args){  
        if(5 < 3){ // условие-вопрос 5 < 3 ?  
            System.out.println("Условие верно"); // ответ на условие  
        } else { // начало ветки ИНАЧЕ  
            System.out.println("Условие не верно"); //ответ ИНАЧЕ  
        }  
    }  
}
```

# Пример программы, которая проверяет, равны ли переменные *a* и *b*

```
public class Proj2 {  
    public static void main(String[] args){  
        int a = 5;  
        int b = 5;  
        if ( a == b) {  
            System.out.println("Переменная a равна переменной b");  
        } else {  
            System.out.println("Переменная a не равна переменной b");  
        }  
    }  
}
```

## Составные условные выражения.

- Составные логические выражения являются подвидом простых условных выражений , таких как:  $a > b$ ,  $a == b$ ,  $a <= b$  и т.д. и также записываются в круглых скобках условных конструкций и циклов.
- Сложные (составные) условия строятся из простых с помощью логических операций: **И**, **ИЛИ**. Все они соответствуют связкам, употребляемым в естественном языке.

# Составные условные выражения. Логические операторы.

- **&&** - логическое «**И**» - проверяет правдивы ли выражения в условии. Если оба правдивы, то все условие правдиво и выполняется ветка **if**, иначе выполняется ветка **else**. Например  
Если (ложь && истина) – результат проверки - **ложь**  
`if ( 5 < 3 && 5 < 7 )` – результат проверки – **ложь** выполняется **else**
- **||** - логическое «**ИЛИ**». Проверяет правдивы ли выражения в условии. Если хотя бы одно истинно, то все условие истинно и выполняется ветка **if**, иначе выполняется ветка **else**.
- Если (ложь || истина) – результат проверки - **истина**  
`if ( 5 < 3 || 5 < 7 )` – результат проверки – **истина** **else** не выполняется

# Составные логические выражения. Примеры.

- Пример двух простых выражений:  $k > b$        $b > c$
- Используя операторы составных выражений их можно объединить в одно целое:
  - 1) Для того, чтобы проверить правдивость двух простых выражений  $k > b$  “И”  $b > c$  в Java будет выглядеть, как  $k > b \ \&\& \ b > c$
  - 2) Для того, чтобы проверить правдивость одного из двух простых выражений  $k > b$  “Или”  $b > c$  в Java будет выглядеть, как  $k > b \ || \ b > c$



# Составные логические выражения. Пример кода.

```
public class Proj2 {  
    public static void main(String[] args){  
        int a = 5;  
        int b = 7;  
        if(a < 10 && b < 10){  
            System.out.println("a меньше 10 И b меньше 10");  
        }  
        if(a < 10 && b > 5){  
            System.out.println("a меньше 10 И b больше 5");  
        }  
        if(a < 10 || b < 10){  
            System.out.println("a меньше 10 ИЛИ b меньше 10");  
        }  
        if(a < 10 || b > 5){  
            System.out.println("a меньше 10 ИЛИ b больше 5");  
        }  
    }  
}
```