

# if-else-if



ПРЕПОДАВАТЕЛЬ



# Артем Гордийчук

**Full-stack software engineer**

- Более 8 лет опыта работы
- Java, Spring, Hibernate, AWS, Oracle, PostgreSQL
- Проекты связанные с банковской, финансовой деятельностью, e-commerce

[artemsgor@gmail.com](mailto:artemsgor@gmail.com)

[www.linkedin.com/in/artem-g-48071a61](https://www.linkedin.com/in/artem-g-48071a61)



# ВАЖНО:

- Камера должна быть включена на протяжении всего занятия.
- Если у Вас возник вопрос в процессе занятия, пожалуйста, поднимите руку и дождитесь, пока преподаватель закончит мысль и спросит Вас, также можно задать вопрос в чате или когда преподаватель скажет, что начался блок вопросов.
- Организационные вопросы по обучению решаются с кураторами, а не на тематических занятиях.
- Вести себя уважительно и этично по отношению к остальным участникам занятия.
- Во время занятия будут интерактивные задания, будьте готовы включить камеру или демонстрацию экрана по просьбе преподавателя.

# ПЛАН ЗАНЯТИЯ

1. Повторение
2. Вопросы по повторению
3. Разбор теста
4. Основной блок
5. Вопросы по основному блоку
6. Задание для закрепления основного блока
7. Практическая работа
8. Оставшиеся вопросы

1

# ПОВТОРЕНИЕ ИЗУЧЕННОГО

# Повторение

- Основные понятия
- Реляционные операторы Java
- Логические операторы
  - Оператор логического И (&&)
  - Оператор «логическое ИЛИ» (||):
  - Оператор логического НЕ(!):
- Унарные операторы





TEL-RAN  
by Starta Institute

# 2

## ВОПРОСЫ ПО ПОВТОРЕНИЮ

# Введение

- Принятие решений в Java
- if
- if-else
- nested if
- if-else-if ladder
- Instruction – return





3

# РАЗБОР ТЕСТА

# Правильные ответы

1. Разработчик - это...
  - a. компьютер, который контролирует выполнение кода.
  - b. кто-то, кто пишет код на любом языке программирования.
  - c. тот, кто вносит изменения в написанный код.
2. Кто такой Джеймс Гослинг?
  - a. Телезвезда.
  - b. Автор объектно-ориентированного языка программирования.
  - c. Генеральный директор корпорации Oracle.
3. Компьютерная программа - это...
  - a. комбинация данных.
  - b. комбинация компьютерных инструкций и данных.
  - c. правила компьютера.
4. Алгоритм - это...
  - a. то, что делает компьютер.
  - b. конечный набор четко определенных правил.
  - c. слово из Википедии.



# Правильные ответы

5. JVM...
- a. позволяет запускать Java-программы на любом устройстве.
  - b. это набор инструментов разработчика Java.
  - c. управляет и оптимизирует программную память.
  - d. компилятор, который преобразует код Java в байт-код.
6. Класс в Java - это...
- a. шаблон для создания объекта.
  - b. шаблон для создания метода.
  - c. что-то, чему мы не научились.
  - d. то, что мы запускаем, когда пишем в консоли '-java Class.'
7. Выберите типы переменных в Java:
- a. Локальные переменные
  - b. Переменные константы
  - c. Частные переменные
  - d. Переменные экземпляра
  - e. Статические переменные



# Правильные ответы

8. Выберите примитивные типы данных в Java:
- a. Boolean
  - b. long
  - c. Object
  - d. int
  - e. Cat
  - f. char
  - g. String
9. Математическая операция в Java  $44 / 33 = ?$
- a. 33
  - b. 44
  - c. 0
  - d. 0.15637
  - e. 1
  - f. 1.0



# Правильные ответы

10. Математическая операция в Java  $44 \% 33 = ?$
- a. 33
  - b. 1
  - c. 11
  - d. 0.11
  - e. 1.0
11. Какой тип данных самый большой?
- a. boolean
  - b. int
  - c. long
  - d. char
12. Строка - это...
- a. массив символов
  - b. тип данных
  - c. объект
  - d. класс в Java
  - e. все вышеперечисленное



# Правильные ответы

13. Объект - это..
- a. класс
  - b. метод
  - c. программа
  - d. алгоритм
14. Как передается тип int в Java?
- a. по ссылке
  - b. по типу
  - c. по значению
  - d. по Wi-Fi
  - e. через Интернет
15. Что это за система счисления? -> 888
- a. Octo
  - b. Decimal
  - c. Binary
  - d. Hex



# Правильные ответы

16. Выберите правильное имя класса (в соответствии с соглашениями):

- a. Student
- b. \_Student
- c. studentRun
- d. RunStudent
- e. S
- f. STUDENT\_RUN
- g. 123Student

17. Выберите правильное имя метода (в соответствии с соглашениями):

- a. Student
- b. \_student
- c. runStudent
- d. run\_student
- e. RunStudent
- f. run

18. Модификаторы доступа в Java...

- a. public
- b. private
- c. final
- d. static



# Правильные ответы

19. Что верно для статических переменных?
- a. единственная копия
  - b. много копий
  - c. доступ через имя класса
  - d. изменения не отражаются
20. Выберите явное преобразование типа (сужение):
- a. int -> long
  - b. long -> int
  - c. double -> byte
  - d. float -> int
21. "I am doing a test".charAt(3);
- a. " " (space)
  - b. a
  - c. m
  - d. d



# Правильные ответы

22. "I am doing a test".substring(3, 6);
- a. md
  - b. m
  - c. am d
  - d. am
  - e. md
  - f. I am doing a test
23. Что может произойти при вызове метода?
- a. метод достигает оператора возврата (return)
  - b. метод не достигает оператора возврата
  - c. ничего не случится
  - d. что-то плохое происходит внутри программы
  - e. метод вызовет исключение



# Правильные ответы

24. Выберите, какой метод будет точкой входа?

a.

```
public void main(String[] args) {  
    int idx = 0;  
    idx++;  
    idx--;  
    out.println(idx);  
}
```

b.

```
public static void main(String[] strings) {  
    int idx = 0;  
    idx++;  
    idx--;  
    out.println(idx);  
}
```

c.

```
public static int main(String[] args) {  
    int idx = 0;  
    idx++;  
    idx--;  
    out.println(idx);  
    return idx;  
}
```

d.

```
public static void myMain(String[] args) {  
    int idx = 0;  
    idx++;  
    idx--;  
    out.println(idx);  
}
```

# Правильные ответы

25. Что напечатает этот метод?
- a. java.lang.ArithmeticException
  - b. 1
  - c. 0
  - d. 1.0
  - e. 2.0
  - f. 0.0
  - g. -1.0
  - h. complete nonsense, it won't work

```
public class Test {  
    public void test() {  
        int idx = 0;  
        idx++;  
        idx--;  
        idx = idx + 1;  
        idx = -1;  
        idx = idx % idx;  
  
        if (idx >= 0) {  
            --idx;  
        } else {  
            idx++;  
        }  
        System.out.println((double) idx);  
    }  
}
```

4

# ОСНОВНОЙ БЛОК

# Принятие решений в Java

- Принятие решений в программировании похоже на принятие решений в реальной жизни.
- В программировании также встречаются ситуации, когда мы хотим, чтобы определенный блок кода выполнялся при выполнении некоторого условия.
- Язык программирования использует операторы управления для управления потоком выполнения программы на основе определенных условий.
- Они используются для того, чтобы поток выполнения продвигался и разветвлялся в зависимости от изменений состояния программы.



# Операторы выбора

if

if-else

if-else-if

nested-if

instruction – return



# Оператор if

Оператор if является самым простым оператором принятия решения.

Он используется для принятия решения о том, будет ли выполняться определенный оператор или блок операторов, т. е. если определенное условие истинно, то блок операторов выполняется, в противном случае нет.

Синтаксис:

**If (условие) {**

**// Операторы для выполнения**

**}**



5

# ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ



## ЗАДАНИЕ

1. Создайте и инициализируйте переменную `int = 10`
2. В блоке `if` сравните созданную переменную со значением 15
3. После блока добавьте инструкцию `System.out.println("I am Not in if");`
4. Проанализируйте вывод

# Оператор if-else

Оператор if говорит нам, что если условие истинно, он выполнит блок операторов, а если условие ложно, то нет.

Но что, если мы хотим сделать что-то еще, если условие ложно.

Мы можем использовать оператор else с оператором if для выполнения блока кода, когда условие ложно.

Синтаксис:

```
if (условие) {
```

```
    // Выполняет этот блок, если условие  
    истинно
```

```
} else {
```

```
    // Выполняет этот блок, если условие  
    ложно
```

```
}
```





TEL-RAN  
by Starta Institute

# 6

## ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

## ЗАДАНИЕ

1. Создайте и инициализируйте переменную `int = 10`
2. В блоке `if` сравните созданную переменную со значением 15
3. Внутри блока добавьте инструкцию `System.out.println("i is smaller than 15");`
4. В блоке `else` добавьте инструкцию `System.out.println("i is greater than 15");`
5. Проанализируйте вывод

# Оператор nested-if

Вложенный if – это оператор если, который является целью другого если или еще.

Вложенные операторы if означают оператор if внутри оператора if.

Java позволяет нам вкладывать операторы if в операторы if. т. е. мы можем поместить оператор if внутри другого оператора if.

Синтаксис:

```
if (условие1) {
```

```
    // Выполняется, когда условие1  
    истинно
```

```
    if (условие2) {
```

```
        // Выполняется, когда условие2  
        истинно
```

```
    }
```

```
}
```



7

# ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

## ЗАДАНИЕ

1. Создайте и инициализируйте переменную `int = 10`
2. В блоке `if` сравните созданную переменную со значением 10 (`==`)
3. Внутри блока добавьте блок `if` и сравните переменную со значением 15 , внутри блока добавить инструкцию `System.out.println("i is smaller than 15");`
4. Добавьте еще один блок `if-else` и сравните переменную со значением 12
5. В блоке `if` добавить инструкцию `System.out.println("i is smaller than 12 too")`
6. В блоке `else` добавьте инструкцию `System.out.println("i is greater than 12");`
7. Проанализируйте вывод

# Оператор if-else-if ladder

Здесь выбираем один из нескольких вариантов.

Операторы if выполняются сверху вниз.

Как только одно из условий, управляющих if, становится истинным, выполняется оператор, связанный с этим if, и остальная часть лестницы игнорируется.

Если ни одно из условий не выполняется, будет выполнен последний оператор else.

Синтаксис:

```
if (condition) {  
    statement;  
} else if (condition) {  
    statement;  
} else {  
    statement;  
}
```







TEL-RAN  
by Starta Institute

# 8

## ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

## ЗАДАНИЕ

1. Создайте и инициализируйте переменную `int = 20`
2. В блоке `if` сравните созданную переменную со значением 10 (`==`)
  - а. В блоке `if` добавить инструкцию `System.out.println("i is 10")`
3. Добавьте блок `else if` и сравните переменную со значением 15 (`==`) , внутри блока добавить инструкцию `System.out.println("i is 15");`
4. Добавьте еще один блок `else if` и сравните переменную со значением 20 (`==`)
5. Добавьте блок `else` добавьте инструкцию `System.out.println("i is not present");`
6. Проанализируйте вывод

# инструкция – return

Java поддерживает оператор (инструкцию) перехода: return.  
Этот оператор передает управление другой части программы.

return

Используется для явного  
возврата из метода -  
управление программой  
передается обратно  
вызывающему методу.



TEL-RAN  
by Starta Institute

# 9

## ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

## ЗАДАНИЕ

1. Создайте и инициализируйте переменную `boolean t = true;`
2. Добавьте инструкцию `System.out.println("Before the return instruction");`
3. В блоке `if` определите оператор `return`
4. Добавьте инструкцию после блока `if` `System.out.println("This won't execute");`
5. Запустите код
6. Проанализируйте вывод используя режим дебаггера
7. Переопределите переменную `t = false`
8. Проанализируйте вывод используя режим дебаггера

# Экспресс-опрос

- **Вопрос 1.**

Объясните, как вы понимаете определение “вложенное ветвление”

- **Вопрос 2.**

При применении инструкции return, куда вернется исполнение программы?



1

# ВОПРОСЫ ПО ПОДБЛОКУ

0



TEL-RAN  
by Starta Institute

1

# ПРАКТИЧЕСКАЯ РАБОТА

1



# Практическое задание 1

## 1. Проанализировать if блоки

```
if (x > 30 && x < 50 && x != 5) {  
    System.out.println ("number from 30 to 50, and not equal to 5");  
}
```

```
if (x < 10 || x > 40) {  
    System.out.println ("number is or less than 10 or greater than 40");  
}
```

```
if (y == 5.5 || y == 10 || y == 8.3) {  
    System.out.println ("number is or 5.5 or 10 or 8.3");  
}
```

```
if (y == 5.5 && y == 10) {  
    System.out.println ("Is there something wrong?");  
}
```



# Экспресс-опрос

- **Вопрос 1.**

Может ли быть так, что в операторе if-else не выполнится ни if, ни else?

- **Вопрос 2.**

Чем отличается условие  $x < 10$  от  $x \leq 10$ , то есть "строго меньше" от "меньше либо равно" ?





TEL-RAN  
by Starta Institute

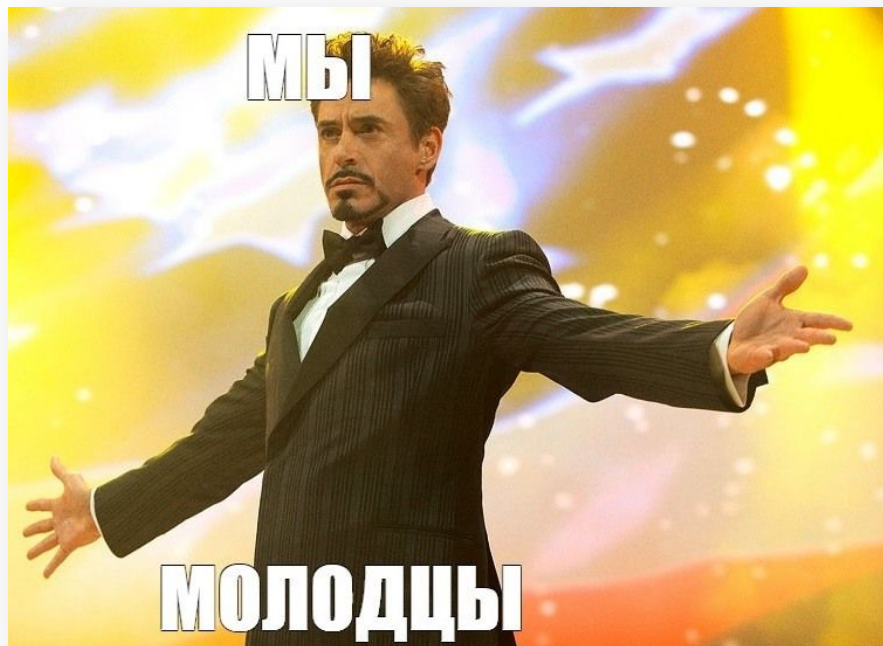
6

# ОСТАВШИЕСЯ ВОПРОСЫ

# Полезные ссылки

- [The if-then and if-then-else Statements \(The Java™ Tutorials > Learning the Java Language > Language Basics\) \(oracle.com\)](#)
- [Conditional \(computer programming\) - Wikipedia](#)

# ЗАКЛЮЧЕНИЕ



# Дополнительная практика

1. Создайте две переменные `*isWeekend*` и `*isRain*`.
2. Создайте переменную `*canWalk*`, значение которой должно быть истинным, если это выходной день (`isWeekend = true`) и не идет дождь (`isRain = false`).
3. Создайте метод, который принимает два параметра и возвращает значение `true/false`
4. Результат сохраните в `canWalk`
5. Распечатайте в консоль.