

# Структура проекта. Примитивные и не примитивные типы данных.



ПРЕПОДАВАТЕЛЬ

**Фото  
преподавателя**

# Имя Фамилия

## Текущая должность

- Количество лет опыта
- Какой у Вас опыт - ключевые кейсы
- Самые яркие проекты
- Дополнительная информация по вашему усмотрению

[Корпоративный e-mail](#)

[Социальные сети \(по желанию\)](#)



# 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. Основной блок
4. Вопросы по основному блоку
5. Задание для закрепления
6. Практическая работа
7. Оставшиеся вопросы

1

# ПОВТОРЕНИЕ ИЗУЧЕННОГО

# Повторение

- Java-разработчик, QA-инженер, Front-end разработчик
- Что такое программа и что такое алгоритм
- Принципы Java. JVM
- Установка IDE, установка JDK
- Hello World
- Точка входа в любую Java-программу ('main' method), структура простой программы
- `System.out.println()`





TEL-RAN  
by Starta Institute

# 2

## ВОПРОСЫ ПО ПОВТОРЕНИЮ

# Введение

- Проект
- Пакет
- Класс
- Метод
- Прimitивные и не примитивные типы данных







TEL-RAN  
by Starta Institute

# 3

## ОСНОВНОЙ БЛОК

# Проект и пакет

Пакет в Java — это контейнер группы связанных классов.

Пакеты используются для:

- Предотвращения конфликтов имен.
  - Например, может быть два класса с именем cars в двух пакетах:  
com.auto.berlin.cars и com.auto.finland.cars
- Упрощения поиска/нахождения и использования классов
- Повторного использования существующих классов из пакетов столько раз, сколько нам нужно в нашей программе.



# Класс, объект и метод в Java

- Класс в Java - это шаблон для создания объекта, а объект - это экземпляр класса.
- Класс содержит переменные и методы, которые называются элементами класса.
- Методы используются для описания того, что объект класса умеет делать или что можно с ним сделать.
- Переменные - для описания свойств или характеристик объекта.



# Соглашения об именах в Java

- Проект:

**Имя собственное**, обычно с заглавной буквы, раскрывает суть всего проекта

*Telegram-Bot, Pizza-Service, Messenger*

- Пакеты и подпакеты:

**Имя существительные** в единственном числе, только в нижнем регистре, в составных именах слова разделяются подчеркиванием. Именуются в обратном порядке доменных имен, например: *org.telran.berlin.lesson.practice*.

*configuration, model, first\_package*



# Соглашения об именах в Java

- Класс:

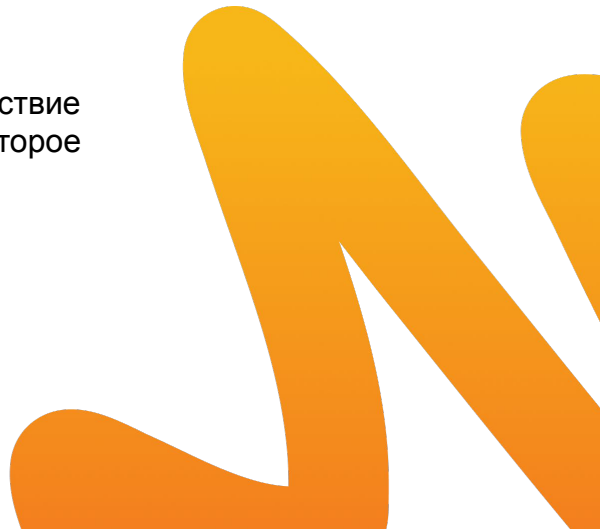
Если вы называете какой-либо класс, то это должно быть **существительное**, и поэтому оно должно быть названо в соответствии с целью, которая должна быть достигнута в программе. Первые буквы слов в верхнем регистре (CamelCasing)

```
class Student { }, class Scanner { }, class Cat { }, HttpResponse { }
```

- Метод:

**Глаголы** в нижнем регистре (print) или словосочетания, отражающие действие (printSum). Имя должно максимально точно описывать действие, которое выполняет метод.

```
method run (), method print (), method get ()
```



# Типы данных

Java - типизированный язык, в котором каждая переменная уже известна во время компиляции.

Как только переменная объявлена определенным типом данных, она не может содержать значения других типов данных.

Примитивный тип данных:

например → **boolean, char, int, short, byte, long, float и double**

Не примитивный тип данных или ссылочный тип данных:

например → **String, Array**



# Примитивные типы данных

- boolean - Логический тип данных представляет только (true, false) - один бит информации: либо истина, либо ложь
- byte - представляет собой число (-128 to 127) - 8 bits.
- short - представляет собой число (-32 768 до 32 767) – 16 bits
- int - представляет собой число (-2147483648 до 2147483647) – 32 bits.
- long – большое целое число (-9223372036854775808 до 9223372036854775807) – 64 bits.
- float - число с плавающей запятой (до 7 десятичных цифр) – 32 bits.
- double - число с плавающей запятой (до 16 десятичных цифр) – 64 bits.
- char - символ Unicode (от '\u0000' (0) до '\uffff' (65535)) – 16 bits.

# Не примитивные типы данных

- String - определяются как массив char-ов

W	o	r	l	d
---	---	---	---	---

- Классы обертки: Integer, Double, Char и т.д.





# Простые математические операторы

Это математические операторы, которые можно использовать для выполнения различных простых или сложных арифметических операций над типами данных.

Сложение (+)

Этот оператор является бинарным и используется для добавления двух операндов.

Вычитание (-)

Этот оператор является бинарным и используется для вычитания двух операндов.

Умножение (\*)

Этот оператор является бинарным и используется для умножения двух операндов.

Деление (/)

Это бинарный оператор, который используется для деления первого операнда (делимого) на второй операнд (делитель) и получения в результате частного.

Модуль (%)

Это бинарный оператор, который используется для возврата остатка, когда первый операнд (делимое) делится на второй операнд (делитель).

**Важно:** в языке Java результат деления целого числа на целое – это всегда целое число, остаток при делении отбрасывается.

# Экспресс-опрос

- **Вопрос 1.**

Определите корректное название для Класа описывающего погоду

- 1) Rain {}
- 2) getWeather {}
- 3) Weather {}
- 4) Cold {}

- **Вопрос 2.**

Какой тип больше float или int?



4

# ВОПРОСЫ ПО ОСНОВНОМУ БЛОКУ



TEL-RAN  
by Starta Institute

5

# ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

### ЗАДАНИЕ

Соотнесите название из 1 столбца с его определением из 2, ответы отправьте в чат:

1. <b>boolean</b>	A. бинарный оператор, который используется для возврата остатка, когда первый операнд (делимое) делится на второй операнд (делитель).
2. <b>модуль (%)</b>	B. определяемый пользователем план или прототип, на основе которого создаются объекты
3. <b>class</b>	C. символ Unicode (от '\u0000' (0) до '\uffff' (65535)) – 16 bits
4. <b>char</b>	D. логический тип данных представляет только (true, false) - один бит информации: либо истина, либо ложь



TEL-RAN  
by Starta Institute

# 6

# ПРАКТИЧЕСКАЯ РАБОТА

# Практическое задание 1

Создать все переменные примитивных и не примитивных типов.

Вывести

в

консоль.



# Реализация задания 1

```
public static void main(String[] args) {  
    byte myByte = 1;           // Byte (whole number from -128 to 127)  
    short myShort = 3;         // Short (whole number from -32768 to 32767)  
    int myNum = 5;             // Integer (whole number)  
    float myFloatNum = 5.99f;  // Floating point number  
    char myLetter = 'D';       // Character  
    boolean myBool = true;     // Boolean  
}
```



# Практическое задание 2

Дано: два целочисленных числа: 168 и 1500.

Примените все математические операторы, результат операций вывести в консоль.



# Реализация задания 2

```
public static void main(String[] args) {  
    int numberOne = 168;  
    int numberTwo = 1500;  
  
    System.out.println(numberOne+numberTwo);  
    System.out.println(numberOne*numberTwo);  
    System.out.println(numberOne/numberTwo);  
    System.out.println(numberOne-numberTwo);  
    System.out.println(numberOne%numberTwo);  
}
```



TEL-RAN  
by Starta Institute

7

# ОСТАВШИЕСЯ ВОПРОСЫ

# Домашнее задание №1

1. В методе main инициализировать все примитивные типы и вывести их результат в консоль:

**char: G**

**int: 89**

**byte: 4**

**short: 56**

**float: 4.7333436**

**double: 4.355453532**

**long: 12121**



# Домашнее задание №2

2. \* Дано трехзначное число. Вывести на экран все цифры этого числа

Пример: 345

Вывод в консоль: Число 345 → 3, 4, 5



# Полезные ссылки

- [Примитивные типы данных](#)
- [Строки в Java](#)

# ЗАКЛЮЧЕНИЕ



# Дополнительная практика

1. Создать пакет game
2. В пакете game создать пакет hero
3. Создать класс Elf
4. Создать поле int id
5. Создать еще 5 полей (любого типа)
6. Создать методы get<имяПоля> и set<имяПоля> для всех полей, подумать и добавить логику в эти методы.

Например для поля id: getId(); setId(int newId);