

## Plan

# 2023-09-07

## Homework

- Naming Conventions (Соглашения по именованию)
- Примитивные типы данных.
- Переменные.
- Арифметические операции.
- Практические задания:
- Общий сценарий:

## Объявление переменных

- присваивание им значений
- вычисления
- печать ответа

## Задания

- сложение, вычитание, умножение
- деление целых чисел
- целая часть от деления
- остаток от деления
- деление чисел типа double, вычисление среднего арифметического

## Lesson Plan №3 07.09.2023

## Homework

- Naming Conventions
- Primitive Data Types
- Variables
- Arithmetic Operations
- Practical Exercises:
- General Scenario:

## Variable Declaration

- assigning values to them
- calculations
- printing the answer

## Tasks

- addition, subtraction, multiplication
- division of integers
- integer part of division
- remainder of division
- division of double type numbers, calculating the arithmetic mean

### Theory

## По русский

### Часть 1: Naming Conventions (Соглашения по именованию)

#### Java naming

**Название файла программы** = название класса – наименование файла программы должно точно совпадать с именем класса. При сохранении файла, Вы должны сохранить его, используя имя класса (помните о чувствительности к регистру) и добавить «.java» в конце имени (если имена не совпадают, Ваша программа не будет компилироваться), например, «MyJavaProgram» – это название класса, тогда файл должен быть сохранен как «MyJavaProgram.java».

**public static void main(String[] args)** – обработка программы начинается с метода main(), который является обязательной частью каждой программы. Идентификаторы – это имена, используемые для классов, переменных и методов. Все компоненты Java требуют имена.

#### Существует несколько правил в синтаксисе языка Java, которые необходимо помнить об идентификаторе.

- Они следующие: Каждый идентификатор должен начинаться с «A» до «Z» или «a» до «z», «\$» или «\_». Имена классов - с большой буквы. Имена методов и переменных - с маленькой буквы. После первого символа может иметь любую комбинацию символов. Ключевое слово не может быть использовано в качестве идентификатора.
- Самое важное – идентификатор в Java чувствителен к регистру! Пример правильного написания: age, \$salary, \_value, \_\_ 1\_value. Пример неправильного написания: 123abc, - salary.

# Ключевые слова Всего зарезервировано 50 служебных (ключевых) слов в Java.

Список ключевых слов по алфавиту:

- abstract assert boolean break byte case catch char class const continue default do double else enum extends final finally float for goto if implements import instanceof int interface long native new package private protected public return short static strictfp super switch synchronized this throw throws transient try void volatile while

**Важно!** Ключевое слово не может быть употреблено в качестве константы или переменной, или любых других имен идентификаторов.

**Полезно!** Составить свой мини-словарь с переводом ключевых слов.

## Переменные

- Читаемость: Имя переменной должно быть понятным и отражать её назначение. Например, `totalAmount` лучше, чем просто `t`.
- Стил: В Java принято использовать `camelCase` для именования переменных. Например, `totalAmount`, `userName`.

## Методы

- Глаголы: Имя метода обычно начинается с глагола, так как метод — это действие. Например, `calculateTotalAmount()`, `getUserInfo()`.
- Читаемость: Имя должно быть понятным и отражать то, что делает метод.

## Константы

- Верхний регистр: Константы именуются в верхнем регистре с подчеркиваниями. Например, `MAX_VALUE`, `PI`.

## Классы

- Существительные: Имя класса обычно является существительным и начинается с заглавной буквы. Например, `User`, `ProductInfo`.
- Читаемость: Имя должно быть понятным и отражать назначение класса.

## Примеры

- Плохие имена: `a`, `str1`, `flag2`
- Хорошие имена: `accountBalance`, `isActive`, `calculateTotalAmount`

**ссылка** - <https://www.oracle.com/java/technologies/javase/codeconventions-namingconventions.html>

## Часть 2: Переменные и Типы Данных

# Переменные

- Переменные — это "контейнеры" для хранения информации, которую можно использовать и изменять в программе.

## Примеры из жизни

- **Кошелек:** Переменная — это как кошелек, в котором можно хранить деньги (значения). Вы можете положить туда деньги,   
  
взять их оттуда или проверить, сколько у вас денег.

## Примитивные типы и строки

- В Java есть разные типы данных:
- логические значения (`boolean`)
- целые числа (`byte`, `short`, `int`, `long`)
- дробные числа (`double`, `float`)
- символы (`char`)
- строки (`String`).

Тип данных определяет диапазон значений, которые может хранить переменная

Тип	Описание	Пример кода	Диапазон значений	Размер (в битах)	Размер в байтах
boolean	Хранит значение <code>true</code> или <code>false</code>	<code>boolean isActive = true; false;</code>	<code>true</code> , <code>false</code>	1 бит	
byte	Целое число	<code>byte a = 3;</code>	-128 до 127	8	1
short	Целое число	<code>short a = 3;</code>	-32,768 до 32,767	16	2
int	Целое число	<code>int a = 4;</code>	-2,147,483,648 до 2,147,483,647	32	4
long	Целое число	<code>long a = 5L;</code>	-9,223,372,036,854,775,808 до 9,223,372,036,854,775,807	64	8
double	Число с плавающей точкой	<code>double x = 8.5;</code>	$\pm 4.910^{-324}$ до $\pm 1.797693134862315710^{308}$	64	8
float	Число с плавающей точкой	<code>float x = 8.5F;</code>	$-3.410^{38}$ до $3.410^{38}$	32	4
char	Одиночный символ в	<code>char c = 'A';</code> <code>char c</code>	0 до 65,535	16	2

Тип	Описание	Пример кода	Диапазон значений	Размер (в битах)	Размер в байтах
	кодировке UTF-16	<code>= '178';</code>			

## Что такое Бит, Байт

- Бит — это минимальная единица информации, одна ячейка в памяти. Он может принимать одно из двух значений: 0 или 1.
- Байт — это набор из 8 битов. В одном байте можно хранить число от 0 до 255 (включительно) в десятичной системе счисления.

## Part 1: Naming Conventions

### Java Naming

Program file name = class name - The name of the program file must exactly match the class name. When saving the file, you should save it using the class name (case-sensitive) and add ".java" at the end (if the names don't match, your program won't compile). For example, if the class name is "MyJavaProgram," the file should be saved as " MyJavaProgram.java".

`public static void main(String[] args)` - The program execution starts from the `main()` method, which is a mandatory part of every program. Identifiers are names used for classes, variables, and methods. All Java components require names.

There are several syntax rules in Java language that you need to remember about identifiers. They are as follows: Each identifier must start with "A" to "Z" or "a" to "z", "\$" or "\_". Class names start with an uppercase letter. Method and variable names start with a lowercase letter. After the first character, it can have any combination of characters. A keyword cannot be used as an identifier.

The most important thing - Java identifiers are case-sensitive! Correct examples: age, \$salary, \_value, \_1value. Incorrect examples: 123abc, -salary.

There are 50 reserved keywords in Java. List of keywords alphabetically:

abstract assert boolean break byte case catch char class const continue default do double else  
enum extends final finally float for goto if implements import instanceof int interface long  
native new package private protected public return short static strictfp super switch  
synchronized this throw throws transient try void volatile while

Important! A keyword cannot be used as a constant or variable, or any other identifier names.

Useful! Create your own mini-dictionary with the translation of keywords.

### Variables

**Readability:** The variable name should be understandable and reflect its purpose. For example, `totalAmount` is better than just `t`. **Style:** In Java, it is customary to use camelCase for variable naming. For example, `totalAmount`, `userName`.

## Methods

**Verbs:** The method name usually starts with a verb, as a method is an action. For example, `calculateTotalAmount()`, `getUserInfo()`. **Readability:** The name should be understandable and reflect what the method does.

## Constants

**Uppercase:** Constants are named in uppercase with underscores. For example, `MAX_VALUE`, `PI`.

## Classes

**Nouns:** The class name is usually a noun and starts with a capital letter. For example, `User`, `ProductInfo`. **Readability:** The name should be understandable and reflect the purpose of the class.

## Examples

Bad names: `a`, `str1`, `flag2` Good names: `accountBalance`, `isUserActive`, `calculateTotalAmount`

[Link](#)

# Part 2: Variables and Data Types

## Variables

Variables are "containers" for storing information that can be used and modified in the program.

## Real-life Examples

**Wallet:** A variable is like a wallet where you can store money (values). You can put money in, take it out, or check how much you have.

## Primitive Types and Strings

In Java, there are different types of data:

- Boolean values (`boolean`)
- Integers (`byte`, `short`, `int`, `long`)
- Floating-point numbers (`double`, `float`)
- Characters (`char`)
- Strings (`String`).

The data type determines the range of values that a variable can store.

Type	Description	Example Code	Value Range	Size (in bits)	Size (in bytes)
boolean	Stores true or false	<pre>boolean isActive = false;</pre>	true, false	1 bit	
byte	Integer	<pre>byte a = 3;</pre>	-128 to 127	8	1
short	Integer	<pre>short a = 3;</pre>	-32,768 to 32,767	16	2
int	Integer	<pre>int a = 4;</pre>	-2,147,483,648 to 2,147,483,647	32	4
long	Integer	<pre>long a = 5L;</pre>	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	64	8
double	Floating-point number	<pre>double x = 8.5;</pre>	$\pm 4.9\text{e-}324$ to $\pm 1.8\text{e}308$	64	8
float	Floating-point number	<pre>float x = 8.5F;</pre>	$-3.4\text{e}38$ to $3.4\text{e}38$	32	4
char	Single character in UTF-16 encoding	<pre>char c = 'A'; char c = '\u00178';</pre>	0 to 65,535	16	2

## What is a Bit, Byte

A bit is the smallest unit of information, a single memory cell. It can take one of two values: 0 or 1.

A byte is a set of 8 bits. One byte can store a number from 0 to 255 (inclusive) in the decimal numbering system.

### Homework

## Написать аппликацию с методами для вычисления различных величин

- 1. Метод вычисляющий площадь круга;
- 2. Метод вычисляющий площадь квадрата;
- 3. Метод вычисляющий площадь прямоугольника (чуть сложнее чем 1. и 2.);

## Create an Application with Methods for Calculating Various Quantities

- 1. Method for calculating the area of a circle;
- 2. Method for calculating the area of a square;
- 3. Method for calculating the area of a rectangle (slightly more complex

## Code

code/Main/src/Variable.java

```
/*
Числовые:
Целочисленные:
byte от -128 до +127
short
int
long
С дробной частью (3,14; 1,23145687)
double - 3.14, 0.7, .7
float - 3.14, 0.7, .7
Символьные
char - хранит 1 символ 'a'; 'b', 'h'; '1'
не примитивный
String - хранить строку. "a b c", "Hello world", "9.89"
Логический
boolean = true/false (0/1)
*/
```

```
public class Variable {

    public static void main(String[] args) {
        /*
        // объявление переменной
        byte thisIsByte;
        // присвоение
        thisIsByte = 5;

        byte thisIsByteVTwo = 30;
        short s = 300;
        int i = 5000;
        long l = 80000;

        int maxInt = 2147483647;
        long longNumber = 2147483648L;

        System.out.println(thisIsByte);
        System.out.println(s);
    }
}
```



```
System.out.println(i);
System.out.println(l);

System.out.println(300);

double d = 4.35;
float f = 8.8987F;

System.out.println(d);
System.out.println(f);

char charA = 'A';
char charUnknown = 1256;

System.out.println(charA);
System.out.println(charUnknown);

String name = "Herr Vasja";

System.out.println("Herr Vasja");
System.out.println("Herr Vasja");

System.out.println(name);
System.out.println(name);
*/

    math();
}

public static void math() {
    byte a = 120;
    int x = 10;
    int y = 5;

    int result = x + y; // 10 + 5 = 15;
    System.out.println(result);

    result = x - y; // 10 - 5 = 5;
    System.out.println(result);

    result = x / y; // 10 : 5 = 2;
    System.out.println(result);
}
```

```
result = 0;
System.out.println(result);

result = x % y; // 10 % 5
System.out.println(result); // 0

result = 9 % 2;
// 9 / 2 = 4,5
// 4 * 2 = 8; = 1
System.out.println(result); // 1

result = 10 % 2;
System.out.println(result); // 0

result = 9 % 2;
System.out.println(result); // 1

result = 10 % 4;
// 10 / 4 = 2,5;
// 2 * 4 = 8;
// 10 - 8 = 2;

System.out.println(result); // 2
}
}
```

code/Main/src/Application.java

```
// camelCase - для методов и переменных (andreySayGoodBye)
// snake_case - для констант (ANDREJ_SAY_GOOD_BYE)
// PascalCase - для классов (AndreySayGoodBye)
```

```
public class Application {

    public static void main(String[] args) {

    }

    // andrey say good bye
    public static void andreySayGoodBye() {
```

```
}  
}
```