



Cohort 34.2



basic\_programming



lesson\_02



▼ Plan

**2023-09-06**

## Lesson 2 Plan (English)

- Introduction

A brief introduction to the lesson's topics.

### Part 1: Creating a Project in the IDE

#### Creating a New Project

- Step-by-step instructions for creating a new Java project in IntelliJ IDEA.

### Part 2: First Program and Basics of Output

#### "Hello, World!"

- Creating and running the first program.

#### System.out.println

- How to display text and variables on the screen.





## Commenting Code

- Single-line and multi-line comments.

## Part 3: Basics of Syntax

### What is Syntax?

- Explanation of the concept of syntax in programming.

### Java Syntax

- Examples and features of Java syntax.
- Class name and file name.

## План 2 Урока (Русский)

### Введение

- Краткое введение в темы урока.

### Часть 1: Создание Проекта в IDE

#### Создание нового проекта

- Пошаговая инструкция по созданию нового Java-проекта в IntelliJ idea.

### Часть 2: Первая Программа и Основы Вывода

#### "Hello, World!"

- Создание и запуск первой программы.

#### System.out.println

- Как выводить текст и переменные на экран.

#### DRY (Не повторяйся)

- Что такое DRY?

#### Комментирование кода



## Что такое синтаксис?

- Объяснение понятия синтаксиса в программировании.

## Синтаксис Java

- Примеры и особенности синтаксиса в Java.
- имя класса и имя файла.

### ▼ Theory

## In English

### First Program "Hello, World!":

1. Open your IDE and create a new Java file.
2. Insert the following code:

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
    }  
}
```

### Notes:

- `public static void main(String[] args)`: the entry point to the program.
- `public static void main(String... args)`: also serves as the entry point to the program.
- `System.out.println()` is a method for outputting text to the console. Text or a variable can be specified in parentheses.

### Types of Comments:

- Single-line: `// This is a single-line comment`
- Multi-line: `/* This is a multi-line comment */`

### Notes:

- Comments are ignored and do not affect program execution.

## DRY



## Meaning

- Repeated code complicates maintenance and makes it more error-prone. If you need to change some logic, you may have to do it in several places, increasing the likelihood of errors.

Copy code

To get an output like this:

Wow, so many of us gathered, 16 people.

I'll go and meet everyone.

**\*\*Hi, my name is Andrey\*\***

Hi, my name is Maria

**\*\*Nice to meet you and thank you for the conversation.\*\***

**\*\*Hi, my name is Andrey\*\***

Hi, my name is Kostya

**\*\*Nice to meet you and thank you for the conversation.\*\***

**\*\*Hi, my name is Andrey\*\***

...

**\*\*Nice to meet you and thank you for the conversation.\*\***

## Part 3: Syntax and Structure Basics



### Semicolon ';'

- In Java, each expression (line of code) must end with a semicolon (;).

#### Real-Life Example:

- **Ending a sentence:** Like in real life, a sentence ends with a period, in Java an expression ends with a semicolon.

### Curly Braces { }

- Curly braces ({} ) are used to define a block of code.

#### Real-Life Example:

- **Rooms in a house:** Curly braces are like rooms in a house. Everything in the room belongs to that room.

## Placement of Classes, Methods, and Variables

- Classes contain methods, and methods contain variables. None of this can exist outside curly braces.

#### Real-Life Example:

# По русский

## Первая программа "Hello, World!":

1. Откройте вашу IDE и создайте новый Java-файл.
2. Вставьте следующий код:

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
    }  
}
```

### Заметки:

- `public static void main(String[] args)`: точка входа в программу.
- `public static void main(String... args)`: так же является точкой входа в программу.
- `System.out.println()` — это метод для вывода текста в консоль.
- В скобках можно указать текст или переменную.



## Типы комментариев:

- Однострочные: `//` Это однострочный комментарий
- Многострочные: `/*` Это многострочный комментарий `*/`

### Заметки:

- Комментарии игнорируются и не влияют на выполнение программы.

## DRY

- **DRY** расшифровывается как "Не повторяйся / "Don't Repeat Yourself"". Это принцип разработки программного обеспечения, направленный на уменьшение избыточности в коде.

## Значение

- Повторение кода затрудняет его поддержку и делает его более подверженным ошибкам. Если вам нужно изменить какую-то логику, возможно, это нужно будет сделать в нескольких местах, что увеличит вероятность ошибок.

```
**Привет, меня зовут Андрей**  
Привет, меня зовут Мария  
**Рад с вами познакомиться и спасибо за общение.**  
**Привет, меня зовут Андрей**  
Привет, меня зовут Костя  
**Рад с вами познакомиться и спасибо за общение.**  
**Привет, меня зовут Андрей**  
...  
**Рад с вами познакомиться и спасибо за общение.**
```

## Часть 3: Основы Синтаксиса и Структуры

### Точка с запятой ';'

- В Java каждое выражение (строка кода) должно заканчиваться точкой с запятой (;).

#### Пример из жизни:

- **Закончить предложение:** Как в реальной жизни предложение заканчивается точкой, в Java выражение заканчивается точкой с запятой.

### Фигурные скобки

- Фигурные скобки ({ }) используются для определения блока кода.

#### Пример из жизни:

- **Комнаты в доме:** Фигурные скобки как комнаты в доме. Всё, что находится в комнате, принадлежит этой комнате.

### Расположение классов, методов и переменных

- Классы содержат методы, а методы содержат переменные. Ничего из этого не может существовать вне фигурных скобок.

#### Пример из жизни:

- **Квартиры и вещи:** Методы как квартиры в доме (классе). Ящики (переменные) могут находиться только в квартирах (методах) или в общих помещениях дома (переменные класса), но не на улице (вне фигурных скобок).



## Пример из жизни:

- **Почтовые ящики:** Номер квартиры (имя класса) и номер на почтовом ящике (имя файла) должны совпадать для правильной доставки почты (компиляции).
- **Паспорт и ФИО:** Представьте, что имя файла — это ваш паспорт, а имя класса — это ваше ФИО в этом паспорте. Они должны совпадать, иначе будет путаница.

### ▼ Home work

## In English

Create a separate project for each task!

## Task 1: Tell About Yourself

### Code Skeleton

```
public class Main {  
    public static void currentCity() {  
    }  
  
    public static void yourNickname() {  
    }  
  
    public static void main(String[] args) {  
    }  
  
    public static void yourName() {  
    }  
  
    public static void casualTalk() {  
    }  
}
```

### Expected Output

```
What is your name?  
My name is Andrey  
How should I address you in class?  
Andrey
```



Berlin

## Task 2: Debugging Exercise

### Code Skeleton

[MainProgram.java](#)

### Expected Output

Hello, World!

## Task 3: Print Special Symbols

### Code Skeleton

```
public class CommonSymbols {
    public static void main(String[] args) {
        /*
        printSymbolsByGroups();
        printSymbolsInReverse();
        printSymbols();
        */
    }

    public static void printSymbols() {
        // code..
    }

    public static void printSymbolsInReverse() {
        // code..
    }

    public static void printSymbolsByGroups() {
        System.out.println("Symbols by Groups:");
        System.out.println("Arithmetic: ...");
        System.out.println("Comparison: ...");
        // ...
    }
}
```





## Task 4: Understanding Variables and Assignment

### Code Skeleton

```
public class VariableBasics {  
    public static void main(String[] args) {  
        // Declare an int variable named 'daysInDecember'  
        // Initialize 'daysInDecember' with the value 31  
        // Declare and initialize a double variable named 'pi' with the value  
        // Declare a String variable named 'favoriteDrink' and assign it the  
        // Print all the variables to the console  
    }  
}
```

### Expected Output

```
Days in December: 31  
Pi Number: 3.14159  
Favorite Drink: [Your Favorite Drink]
```

## По русский

Создайте отдельный проект для каждой задачи!

### Задача 1: Расскажите о себе

Код-скелет

```
public class Main {  
  
    public static void currentCity() {  
    }  
  
    public static void yourNickname() {  
    }  
  
    public static void main(String[] args) {  
    }  
}
```



```
}  
  
    public static void casualTalk() {  
    }  
}
```

**Ожидаемый вывод** (обратите внимание на порядок вывода)

Как вас зовут?  
Меня зовут Андрей  
Как к вам обращаться на занятиях?  
Андрей  
Вы не против со временем перейти на ты?  
Со временем, да  
В каком городе сейчас проживаете?  
Berlin

## Задача 2: Упражнение по отладке

Исправить не компилируемый код

Код-скелет

MainProgram.java

Ожидаемый вывод

Hello, World!

## Задача 3: Вывести специальные СИМВОЛЫ

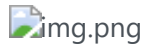
Вывести на консоль наиболее часто используемые специальные символы в одном выводе и затем в обратном порядке. Вывести символы по группам.

```
public class CommonSymbols {  
    public static void main(String[] args) {  
        /*  
        printSymbolsByGroups();  
        printSymbolsInReverse();  
        printSymbols();  
        */  
    }  
}
```



```
public static void printSymbolsInReverse() {  
    // code..  
  
}  
  
public static void printSymbolsByGroups() {  
    System.out.println("Символы по группам:");  
    System.out.println("Арифметические: ...");  
    System.out.println("Сравнение: ...");  
    // ...  
}  
}
```

**Ожидаемый вывод** (обратите внимание на порядок вывода)



▼ **Code**

code/hello\_world/src/Main.java

```
// однострочный комментарий  
/*  
  
*/  
  
public class Main {  
  
    public static void main(String... args) {  
  
        /*  
        room1();  
        room2();  
        room3();  
        room1();  
        room1();  
        room2();  
        room3();  
        */  
  
        /*  
        Тут я могу писать комментарий на несколько строк.  
        Вот к примеру строка 2  
        Вот к примеру строка 3  
        */  
    }  
}
```





```
System.out.println("I am in room 1"); // тут будет печать в 1
System.out.println("I am in room 2"); // Маша, что тут проис
System.out.println("I am in room 3");
// System.out.println("I am in room 1");
// System.out.println("I am in room 1");
// System.out.println("I am in room 2");
// System.out.println("I am in room 3");
}

public static void room1() {
    System.out.println("I am in room 1");
}

public static void room3() {
    System.out.println("I am in room 3");
}

public static void room2() {
    System.out.println("I am in room 2");
}
}
```

code/hello\_world/src/Dry.java

```
public class Dry {

    public static void main(String... arg) {

        greet();
        // System.out.println("Привет, меня зовут Андрей");
        meetMaria();
        // System.out.println("Привет, меня зовут Маша");
        sayGoodBye();
        // System.out.println("Приятно познакомиться, хорошего вечера");

        greet();
        // System.out.println("Привет, меня зовут Андрей");
        meetKostja();
        // System.out.println("Привет, меня зовут Костя");
        sayGoodBye();
        // System.out.println("Приятно познакомиться, хорошего вечера");

        greet();
        // System.out.println("Привет, меня зовут Андрей");
    }
}
```





```
// System.out.println("Приятно познакомиться, хорошего вечера");

greet();
// System.out.println("Привет, меня зовут Андрей");
meetAndrej();
// System.out.println("Привет, меня зовут Андрей");
sayGoodBye();
// System.out.println("Приятно познакомиться, хорошего вечера");
}

public static void greet() {
    System.out.println("Привет, меня зовут Андрей");
}

public static void meetMaria() {
    System.out.println("Привет, меня зовут Маша");
}

public static void meetKostja() {
    System.out.println("Привет, меня зовут Костя");
}

public static void meetNastja() {
    System.out.println("Привет, меня зовут Настя");
}

public static void meetAndrej() {
    System.out.println("Привет, меня зовут Андрей");
}

public static void sayGoodBye() {
    System.out.println("Приятно познакомиться, хорошего вечера");
}
}
```



[Legacy lesson view](#)