

Настройка среды разработки. Синтаксис языка. Основные типы данных и операции в Java

[Настройка среды разработки](#)

[Установка среды разработки для Java](#)

[Создание проекта в IntelliJ IDEA](#)

[Структура проекта](#)

[Структура программы на Java](#)

[Основные типы данных и операторы](#)

[Задания](#)

Настройка среды разработки

Для разработки на Java вам потребуется:

- **JDK (Java Development Kit):** Это набор инструментов, необходимых для компиляции и запуска Java-программ. Вы можете скачать его с официального сайта [Oracle](#).
- **IDE (Integrated Development Environment):** Это программа, которая упрощает процесс написания, компиляции и отладки кода. Популярные IDE для Java:
 - [IntelliJ IDEA](#)
 - Eclipse
 - NetBeans
 - Visual Studio Code (с расширением для Java)

Самая популярная среда разработки для Java – **IntelliJ IDEA**. В ней есть все необходимое для создания проектов: текстовый редактор, компилятор,

отладчик и другие инструменты. У IntelliJ IDEA есть бесплатная и платная версии. На первое время хватает возможностей бесплатной версии – IntelliJ IDEA Community Edition.

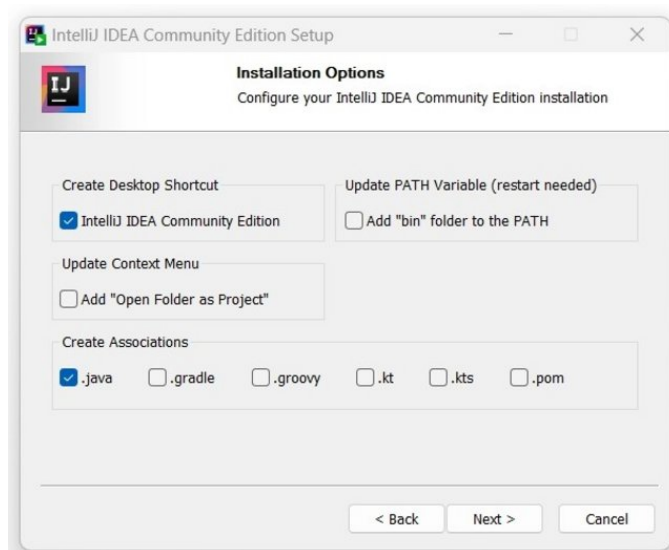
Установка среды разработки для Java

Скачайте версию Community для Windows с официального [сайта JetBrains](https://www.jetbrains.com/idea/).

Запустите скачанный исполняемый файл и выберите путь для установки. По умолчанию это папка *Program Files/JetBrains*.

Настройте параметры установки. Общие настройки:

- 64-bit launcher – добавление на рабочий стол ярлыка IntelliJ IDEA.
- Add «**Open Folder as Project**» – открытие папки с исходниками в редакторе через контекстное меню.
- **.java** – файлы с таким расширением по умолчанию будут открываться через IntelliJ IDEA.

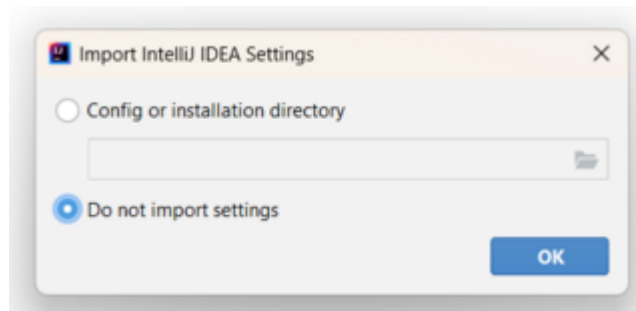


Выберите папку для ярлыков (по умолчанию) JetBrains. Затем нажмите Install и Finish. Установка IntelliJ IDEA завершена.

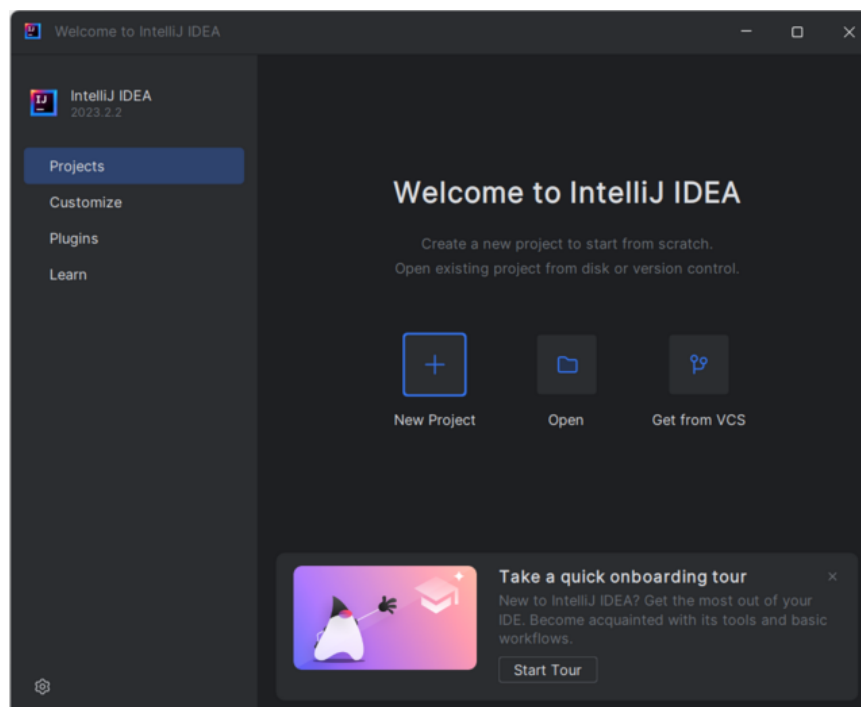
Создание проекта в IntelliJ IDEA

При первом запуске IntelliJ IDEA перед вами выскочит диалоговое окно с требованием указать путь до файла с настройками. Так как это наш первый опыт знакомства с IDE, то выберите пункт *Do not import settings*.

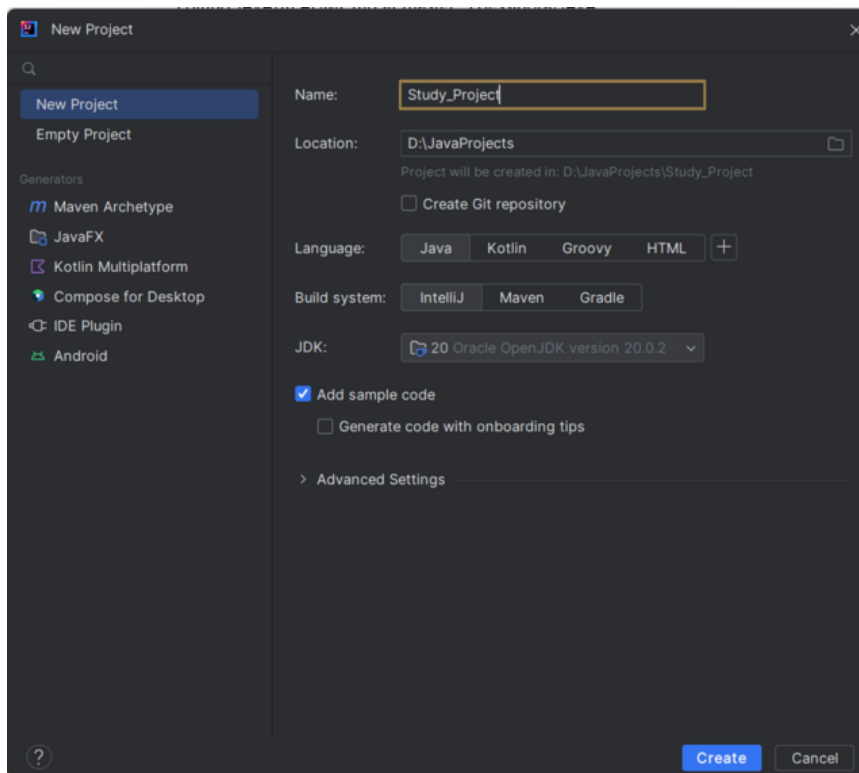
Если данное окно вылезло после обновления или переустановки – выберите исходную директорию.



Далее откроется следующее окно, из которого мы сразу можем создать новый проект.



Создайте ваш первый тестовый проект на Java в IntelliJ IDEA. Для этого в Idea на главном экране выберем **Create new project**. Откроется окно с настройкой конфигураций нового проекта:

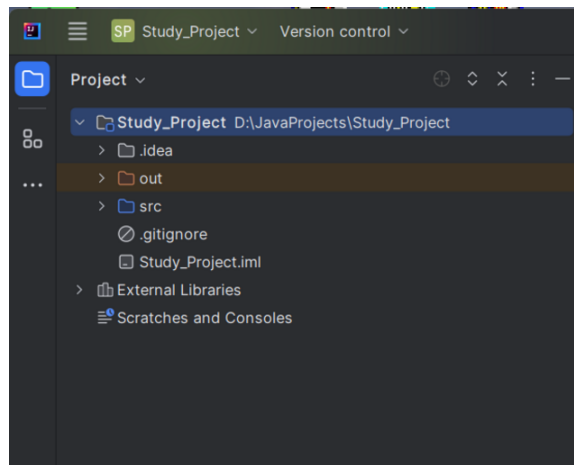


- Придумайте название проекта на латинице.
- Укажите язык Java
- Выберите систему сборки Java
- Укажите нужную версию JDK
- Нажмите **Create**, после чего IntelliJ сделает сборку вашего проекта

Структура проекта

Разверните название вашего проекта в корневой директории, в ней будет:

- скрытая директория с расширением **.idea**, в которой хранятся настройки Idea
- директория **src** – в этой папке находятся все те файлы, с которыми вы работаете. В этой папке уже есть файл **main.java**.
- файл с названием вашего проекта и расширением **.iml** – он содержит различную информацию о вашем проекте (о том, что он на языке Java, различные дополнительные характеристики)



Структура программы на Java

Поскольку Java является **объектно-ориентированным** языком программирования, то все файлы представляют из себя **классы**. Для обращения к классам необходимо создавать объекты и через объекты можно получать данные из класса. Наименование класса должно начинаться с **заглавной буквы** и строго совпадать с **названием файла**.

Просмотрите следующий код:

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

Вывод фразы «Hello World!» представлен ниже:

```
System.out.print("Hello World!");
```

Сперва происходит обращение к классу System, после чего берем его объекты и методы. Поскольку в System создан объект out, а у него есть метод **print** и **println**, то мы можем обратиться к ним через точку.

Основные типы данных и операторы

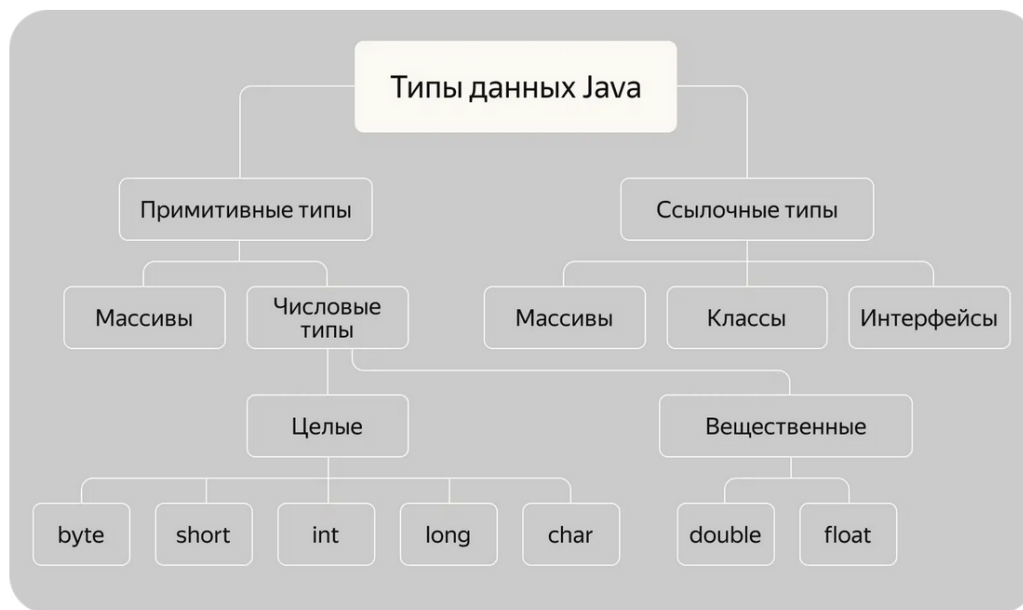
В Java типы данных делят на две большие группы: примитивные и ссылочные.

В состав примитивных типов (или просто примитивов) входят четыре

подвида

и восемь типов данных:

1. целые числа (`byte` , `short` , `int` , `long`);
2. числа с плавающей точкой (`float` , `double`);
3. логический (`boolean`);
4. символьный (`char`).



Задания

1. Создайте новый проект и напишите программу, которая:
 - Объявляет переменные разных типов данных.
 - Выполняет арифметические и логические операции с этими переменными.
 - Выводит результаты на консоль.
2. Напишите программу, которая раскладывает заданное число на простые множители.
3. Напишите программу, которая запрашивает у пользователя исходную цену товара и процент скидки, а затем вычисляет новую цену со скидкой.

4. Напишите программу, которая запрашивает у пользователя температуру в градусах Цельсия (целое число) и преобразует ее в градусы Фаренгейта. Формула преобразования: $F = (C * 9/5) + 32$. Выведите результат в виде числа с плавающей точкой.