

Tutorial For JDK Installing

Based on the tutorial of "2020S-Java-A" designed by teaching group in SUSTech

Modified (mainly change to markdown file) by ZHU Yueming in 2021. Jan. 11th

IDEA part modified from ZHU Yueming's tutorial by WANG Daxing in 2022. Sept.

Improve by ZHU Yueming in 2023. Aug. 28th.

- Merge jdk and idea.
- Add compile and run on Mac system
- Change JDK installation way from open JDK to installer

Objectives

1. Install JDK 17 and configure environment variable.
2. Learn compilation and execution of your first Java program in command line.
3. Learn how to use an **Integrated Development Environment (IDE)** in writing JAVA programs

Part 1. JDK Software Installation

1. JDK Introduce

JDK: The Java Development Kit (JDK) is a software development environment for developing Java applications and applets. It includes a Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development.

2. Download and Install JDK

Download

URL of download JDK:

<https://www.oracle.com/java/technologies/downloads/#jdk17-windows>

JDK Development Kit 17.0.8 downloads

JDK 17 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions](#).

JDK 17 will receive updates under these terms, until September 2024, a year after the release of the next LTS.

[Linux](#) [macOS](#) [Windows](#)

Product/file description	File size	Download
x64 Compressed Archive	172.38 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip (sha256)
x64 Installer	153.48 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe (sha256)
x64 MSI Installer	152.27 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi (sha256)

JDK Development Kit 17.0.8 downloads

JDK 17 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions](#).

JDK 17 will receive updates under these terms, until September 2024, a year after the release of the next LTS.

[Linux](#) [macOS](#) [Windows](#)

Product/file description	File size	Download
ARM64 Compressed Archive	168.12 MB	https://download.oracle.com/java/17/latest/jdk-17_macos-aarch64_bin.tar.gz (sha256)
ARM64 DMG Installer	167.55 MB	https://download.oracle.com/java/17/latest/jdk-17_macos-aarch64_bin.dmg (sha256)
x64 Compressed Archive	170.56 MB	https://download.oracle.com/java/17/latest/jdk-17_macos-x64_bin.tar.gz (sha256)
x64 DMG Installer	169.98 MB	https://download.oracle.com/java/17/latest/jdk-17_macos-x64_bin.dmg (sha256)

Note: We recommend installer type to install jdk, and you need to download the corresponding JDK according to your operating system.

- Please choose 17.0.8 version.
- If you use Mac OS, please check the form of your CPU. If your computer uses **Intel** chip, please download **x64 DMG** version. If your computer uses **Apple M1** chip, please download **Arch64 DMG** version.

Install

After your downloading process, you can open your **installation package** (.exe file or .dmg file) and then click next step to install your jdk software.

After your installation process, you can check whether the installation is successful by the following steps:

Windows User

Press **Win+R**. Type **cmd** to enter the command prompt.

Then type the command below:

```
java -version  
javac -version
```

You will see one of two things. If the JDK / Java compiler is correctly installed, you should see a version number below:

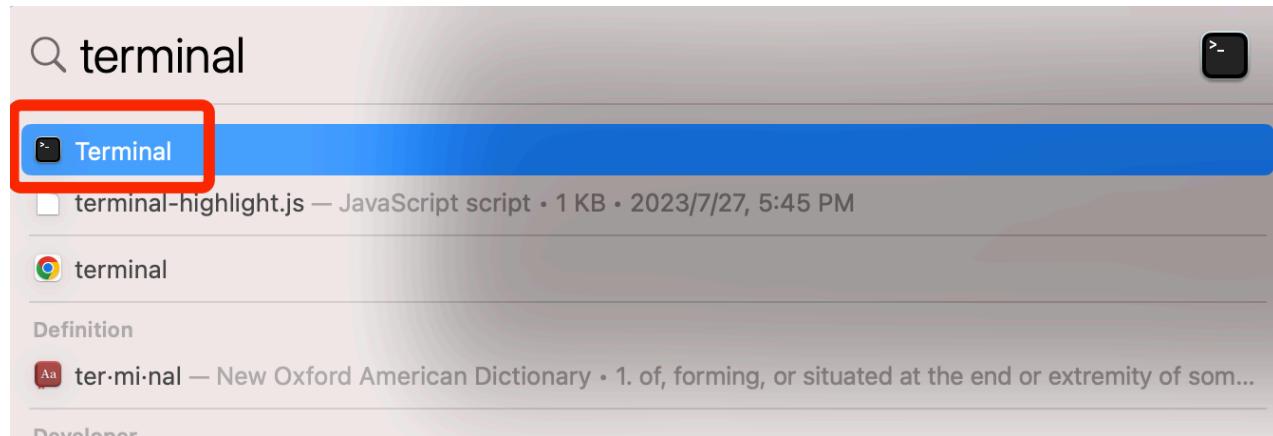
```
> java -version  
java version "17.0.4.1" 2022-08-18 LTS  
Java(TM) SE Runtime Environment (build 17.0.4.1+1-LTS-2)  
Java HotSpot(TM) 64-Bit Server VM (build 17.0.4.1+1-LTS-2, mixed mode, sharing)  
> javac -version  
javac 17.0.4.1
```

However, if the JDK isn't installed or the environment variables are not properly configured, then you'll see an error below:

```
C:\Users\18716>java -version  
'java' 不是内部或外部命令, 也不是可运行的程序  
或批处理文件。  
C:\Users\18716>javac -version  
'javac' 不是内部或外部命令, 也不是可运行的程序  
或批处理文件。
```

Mac User

Press `command+space` or `control+space`. Type `terminal` to open the terminal



Then type the command below:

```
java -version  
javac -version
```

You will see one of two things. If the JDK / Java compiler is correctly installed, you should see a version number below:

```
(base) yuemingzhu@YUEMINGs-MacBook-Air ~ % java -version
java version "17.0.4.1" 2022-08-18 LTS
Java(TM) SE Runtime Environment (build 17.0.4.1+1-LTS-2)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.4.1+1-LTS-2, mixed mode, sharing)
(base) yuemingzhu@YUEMINGs-MacBook-Air ~ % javac -version
javac 17.0.4.1
```

Compile and Run

Compile and Run a Java Program in Command line.

Process	File Type	Result	Command	Description
Compile	.java File	.class File	javac	Compile the java file through the java compiler and generate a .class file that can be recognized by JVM
Run	.class File	Running result	java	Run the .class file on the JVM to get the result

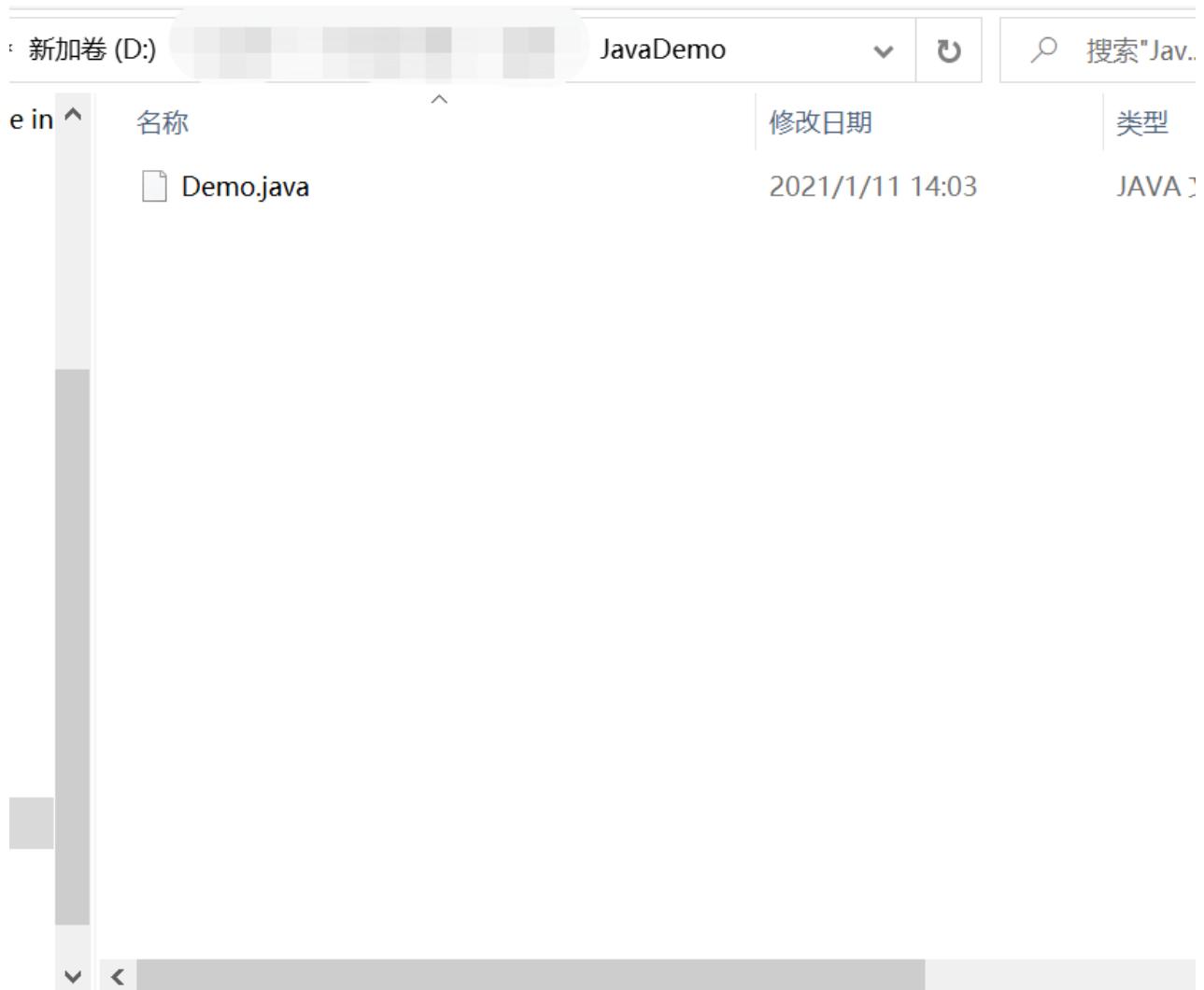
Exercise Process:

windows User:

- Open txt file and create a new text file and type the following code.

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

- Using the “save as” command in the “File” menu to save the file with the name `Demo.java`. Note that the .java file name has to be **the same as** the main Class name in your program.



- Open the cmd program; in the cmd window, use cd command to go to the directory where you save your Demo.java; use javac command to compile your ".java" file; use java command to execute the .class file (which must have a main function). The procedure is as follows

```
D:\JavaDemo>javac Demo.java  
D:\JavaDemo>java Demo  
Hello, world!
```

Also, you may notice that a new file **Demo.class** has been produced. This file works when we run the command in cmd. The `java Demo` command still works inspite you delete the **Demo.java** file. However, it is not readable because it is a binary file.

Mac User:

- Open terminal
Press `command+space` or `control+space`. Type `terminal` to open the terminal
- Create a txt file by command below

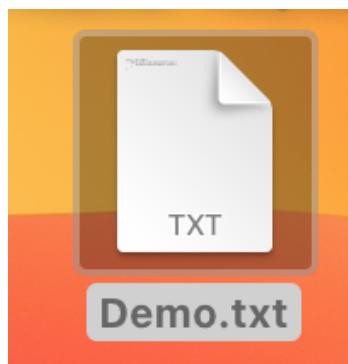
```
cd desktop  
touch Demo.txt
```

After that, a file named `Demo.txt` has been created on desktop

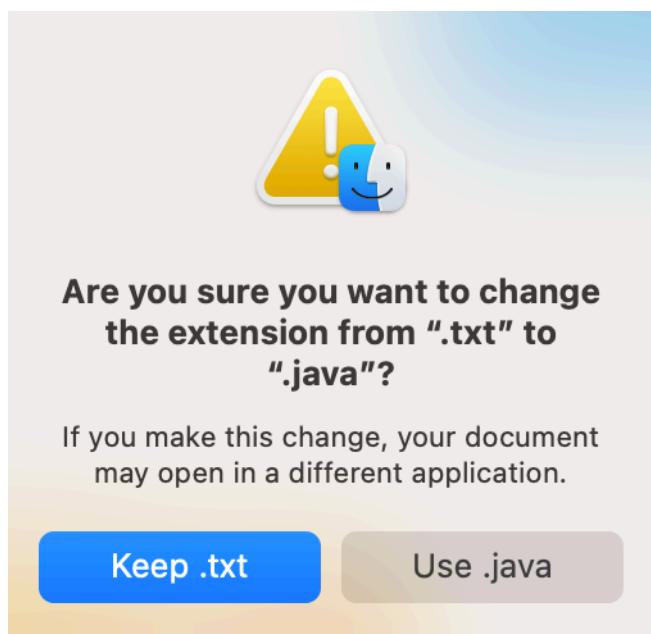
- Open the txt file and type the following code

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

- Save and close it. You can get a txt file in desktop



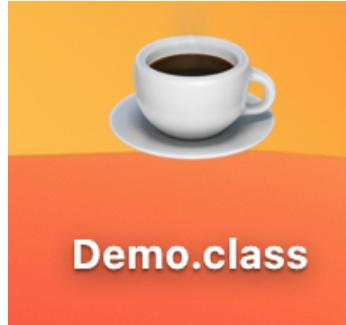
- Rewrite the txt extension name to java.



- Compile: type following command in terminal.

```
javac Demo.java
```

Then you can get a `Demo.class` file in desktop



- Run: type following command in terminal:

```
java Demo
```

Then `Hello, world!` would be displayed in your terminal.

The whole process above:

```
Last login: Mon Aug 28 17:27:10 on ttys000
(base) yuemingzhu@YUEMINGs-MacBook-Air ~ % java -version
java version "17.0.4.1" 2022-08-18 LTS
Java(TM) SE Runtime Environment (build 17.0.4.1+1-LTS-2)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.4.1+1-LTS-2, mixed mode, sharing)
(base) yuemingzhu@YUEMINGs-MacBook-Air ~ % javac -version
javac 17.0.4.1
(base) yuemingzhu@YUEMINGs-MacBook-Air ~ % cd desktop
(base) yuemingzhu@YUEMINGs-MacBook-Air desktop % touch Demo.txt
(base) yuemingzhu@YUEMINGs-MacBook-Air desktop % javac Demo.java
(base) yuemingzhu@YUEMINGs-MacBook-Air desktop % java Demo
Hello, world!
(base) yuemingzhu@YUEMINGs-MacBook-Air desktop % █
```

Other Basic CMD:

- Change to d driver:

```
d:
```

- go into sub folder `xxx`:

```
cd xxx
```

- go into root folder:

```
cd /
```

- go into super folder:

```
cd ..
```

- find all files in current folder:

```
dir
```

In this lab, you are using an editor and the compiler separately. There are software products (e.g. Eclipse, NetBeans, IDEA) that could link them up and facilitate your work. In this semester, you are recommended to use IDEA. You can download the community version at the following link:

<https://www.jetbrains.com/idea/download/>

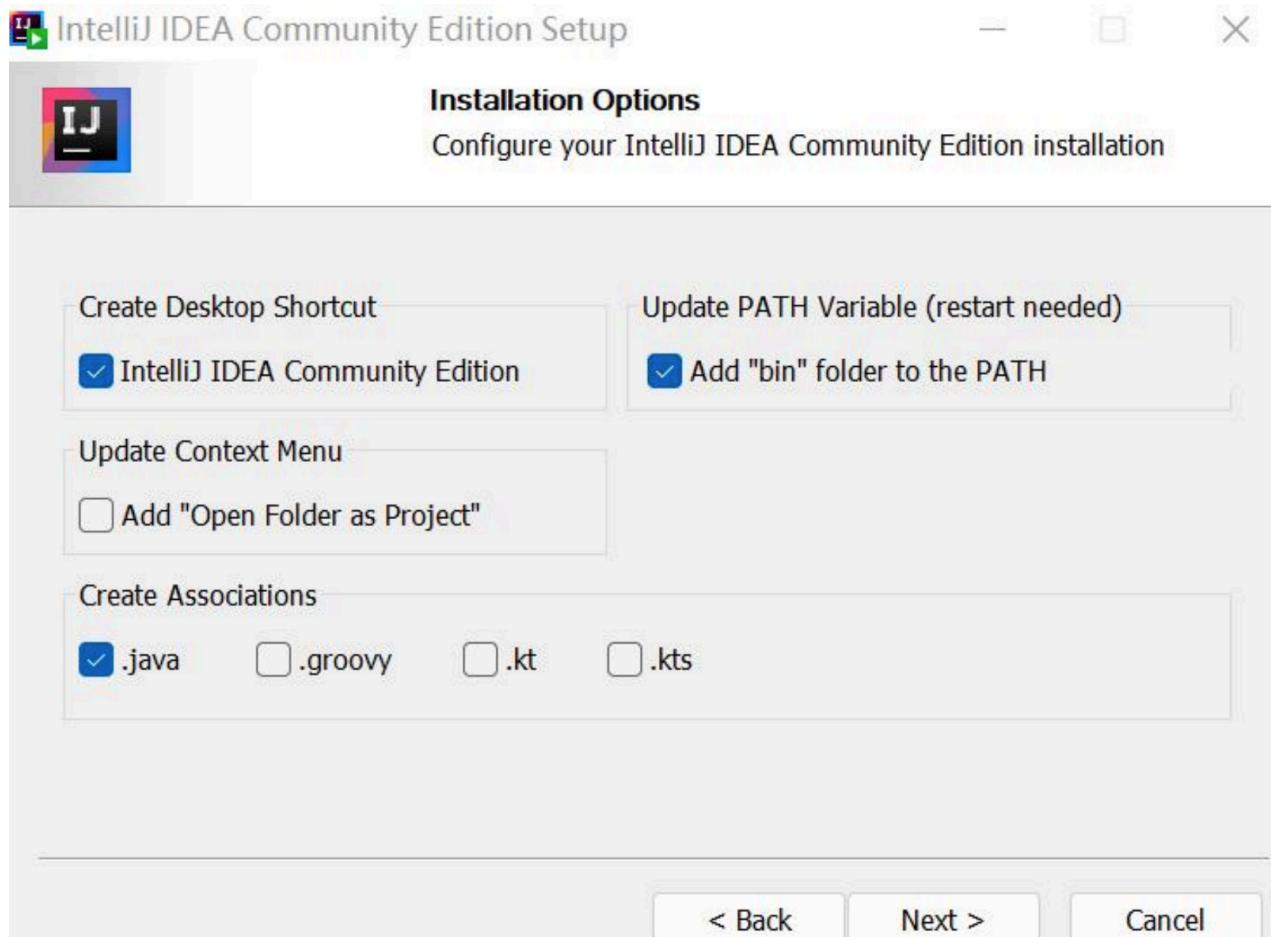
Part 2. IntelliJ IDEA

Install

In this course, we will use IDEA as our reference IDE.

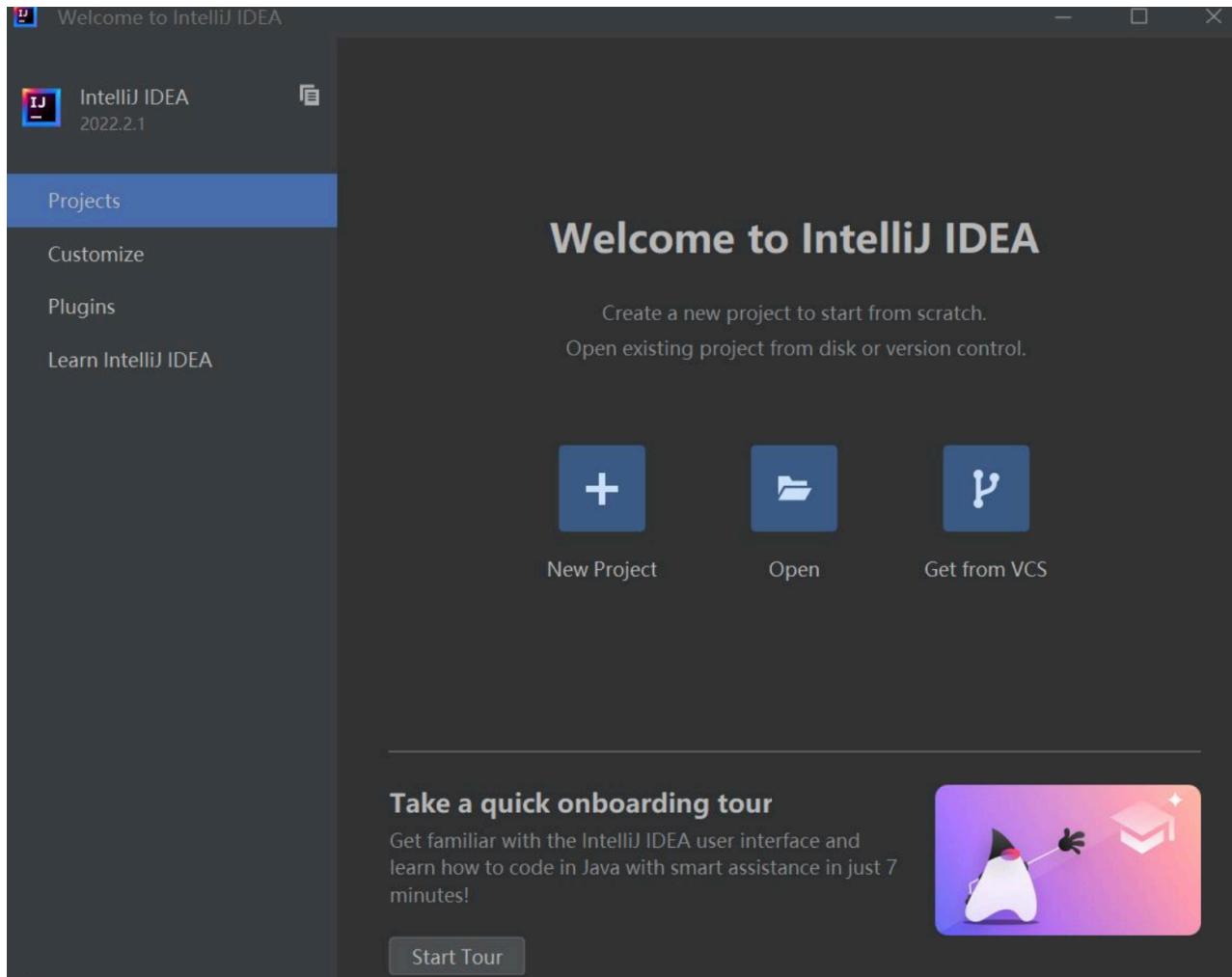
You can download IDEA (community version) at the following link: <https://www.jetbrains.com/idea/download/>

Once downloaded, run the executable. Follow the prompts to install IDEA, tick "Add launchers dir to the PATH" and ".java in Create Associations" as follows:



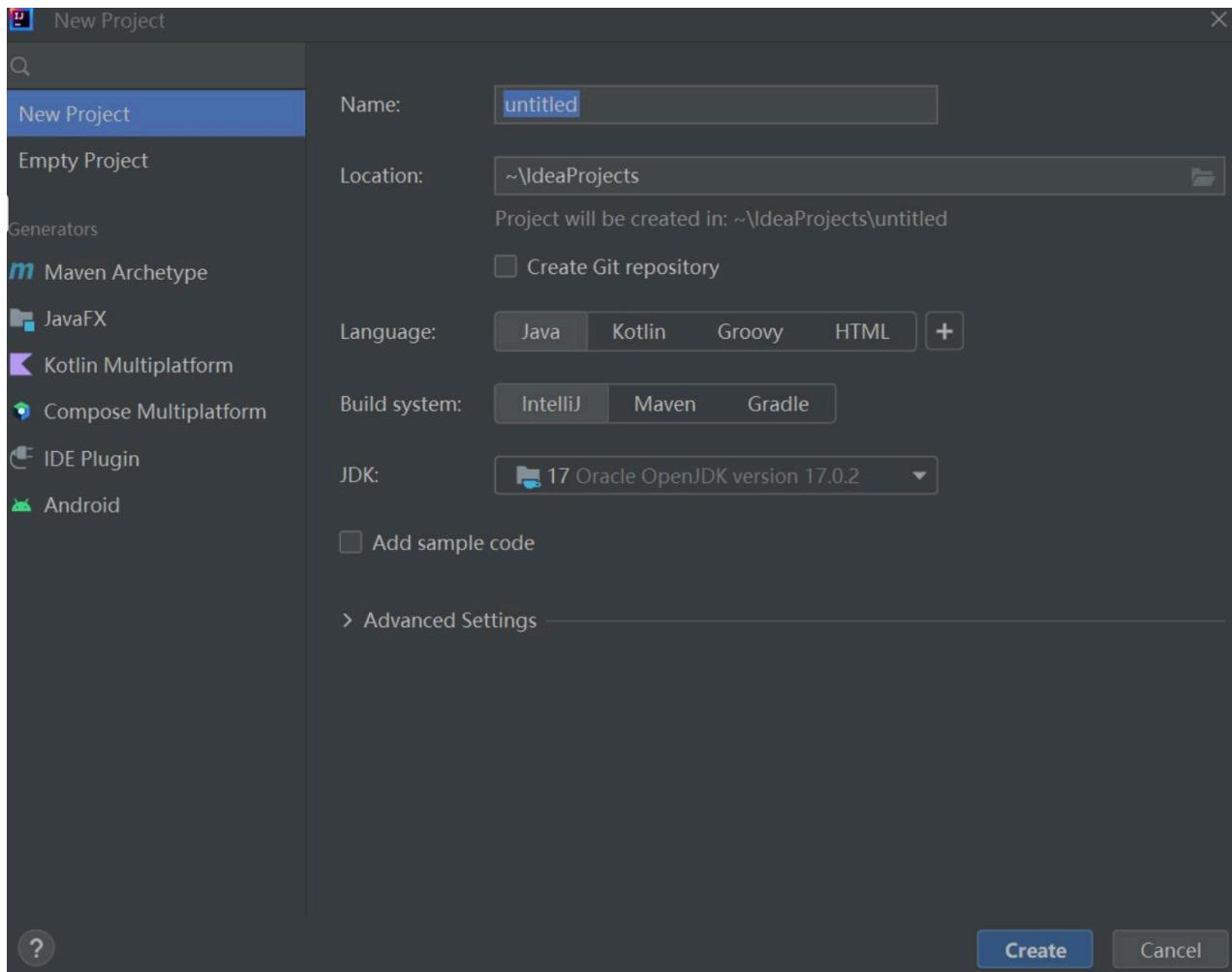
After the installation, you are suggested to restart your computer.

Find your IntelliJ IDEA Community Edition in the Start menu to start IDEA. Confirm the Privacy Policy (You are recommended to read the policy for this and any other software), and decide whether you want to send anonymous statistics data to the software developer, i.e., JetBrains. Finally, the following start window prompts:

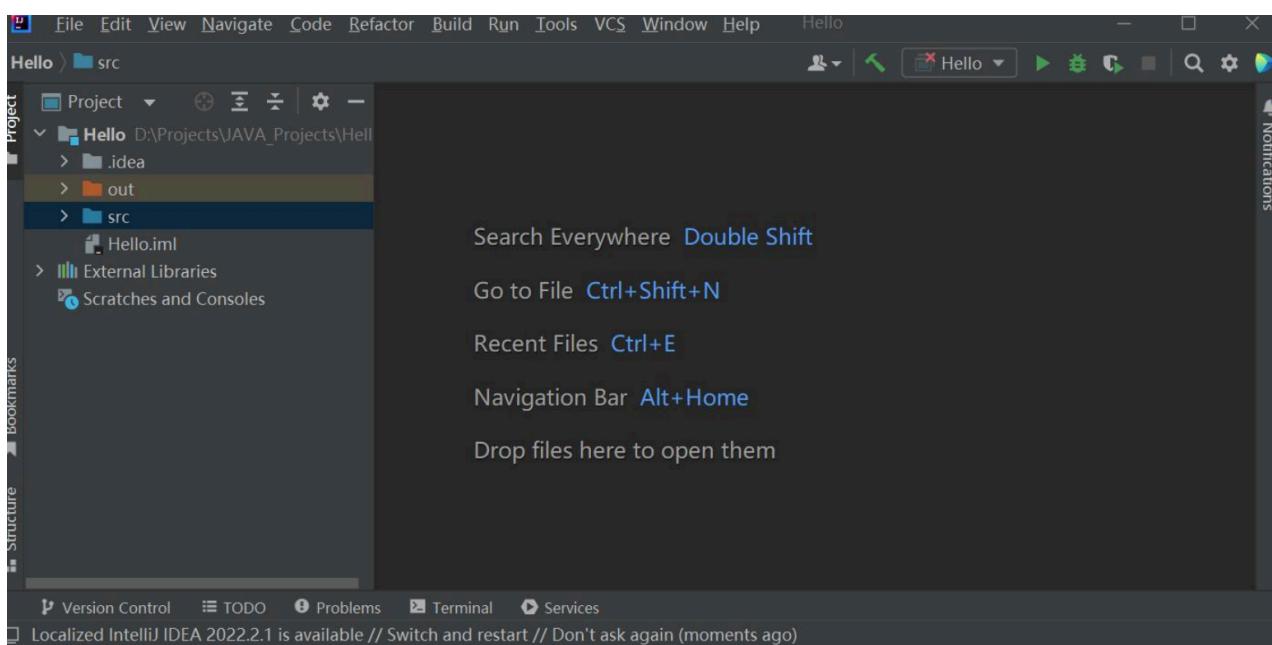


Create new Java Program

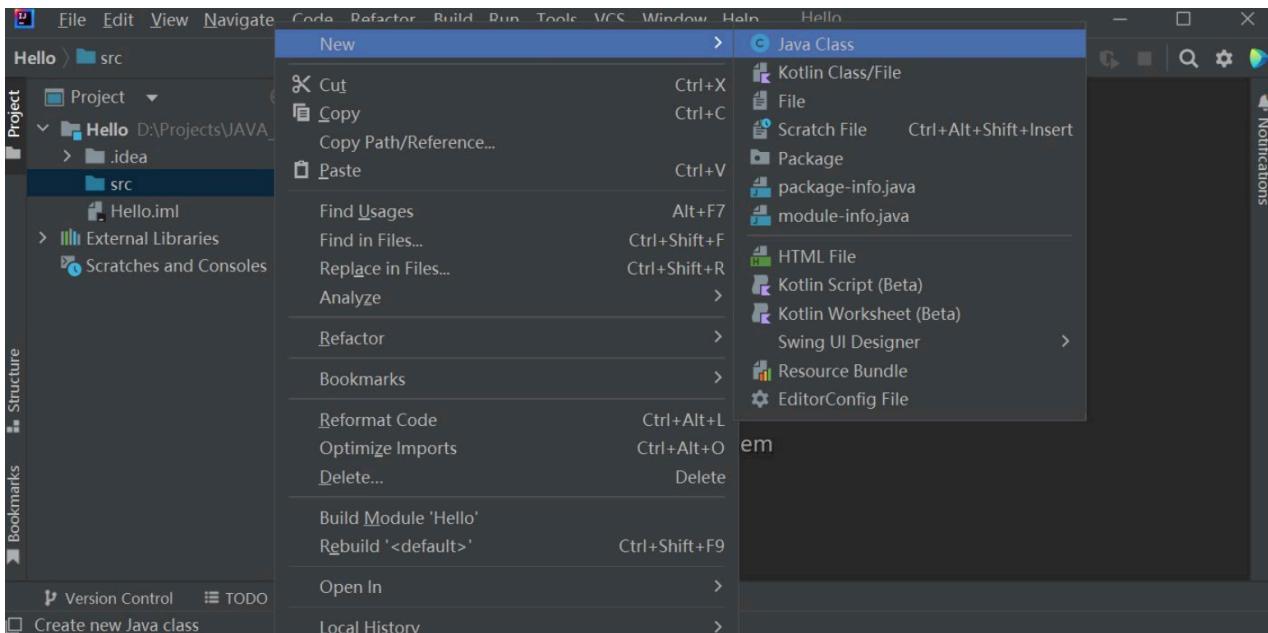
To create a new java program, click New Project. Set project name and location. This is where you want to put all your project files. Make sure that the Project SDK at the top is set. Then press "Create" with the default setting.



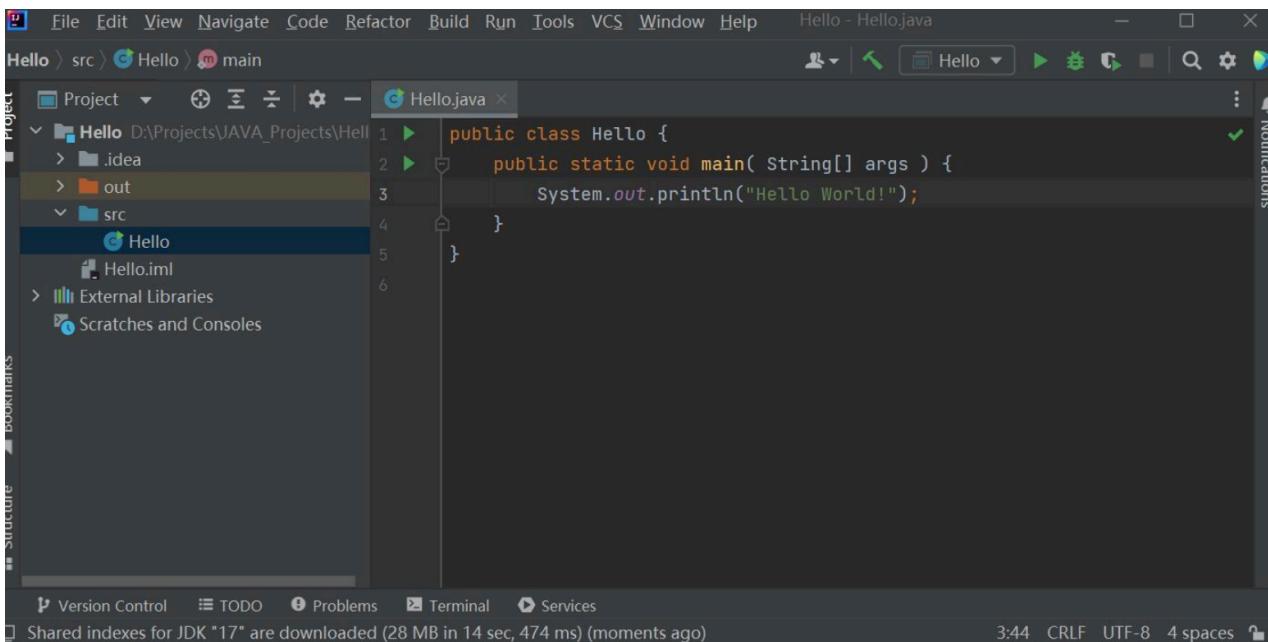
The project should appear and look like follows:



In order to start here, a java file should be created. Right click "src" (which corresponds to the "src" directory in the project directory just set), choose "new" and "Java Class", left click "Java Class" to create a java file.



After the java file is created, write some code here:



The build and run the code, simply click the green triangle:

```
1 public class Hello {  
2     public static void main( String[] args ) {  
3         System.out.println("Hello World!");  
4     }  
5 }
```

The following result should be seen:

```
D:\Software\jdk-17.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.2.1\lib\idea_rt.jar" -Dfile.encoding=UTF-8 Hello  
Hello World!  
Process finished with exit code 0
```

Futher Exercise

This part you can only type the code and run it. You do not understand the specific meaning of each statement. We would introduce in following course in detail.

2.1 Exercise 1

Write and test the following source code to find the difference of `print`, `println` and `printf`:

```
public class Sum {
    public static void main(String[] args) {
        int number1 = 4;
        int number2 = 3;
        System.out.println("number1+number2=" + (number1 + number2));
        System.out.print("number1+number2=" + (number1 + number2));
        System.out.printf("%d+%d=%d\n", number1, number2, number1 + number2);
    }
}
```

2.2 Exercise 2

Define informations of a person, and then prints out in a specific format:

```
public class Information {
    public static void main(String[] args) {
        String name = "Liming";
        int age = 18;
        float weight = 55.75f;
        char grade = 'A';

        System.out.printf("You are %s.\nYou are %d years old.\n");
        System.out.printf("You weight %.1f KG.\nThe highest grade you got is
%c\n", weight, grade);
    }
}
```

The output looks like this:

```
You are Liming.
You are 18 years old.
You weight 55.8 KG.
The highest grade you got is A
```