

An Introduction to IntelliJ

IntelliJ IDE for Java Developers

Produced Dr. Siobhán Drohan
by: Mr. Colm Dunphy
 Mr. Diarmuid O'Connor
 Dr. Frank Walsh



Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
<http://www.wit.ie/>



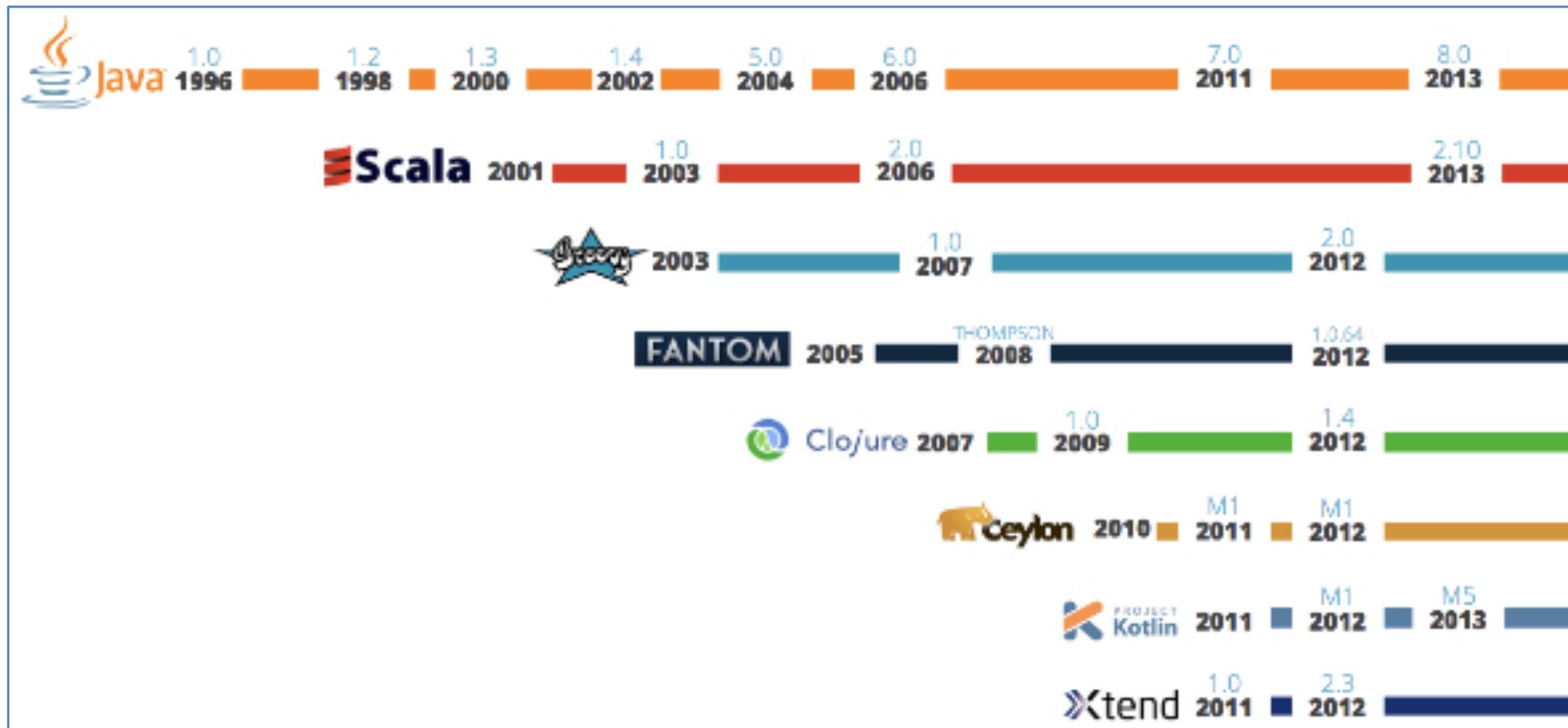
" Most people talk about Java the language, and this may sound odd coming from me, but I could hardly care less. At the core of the Java ecosystem is the JVM. "

James Gosling,

Creator of the Java Programming Language(2011, TheServerSide)

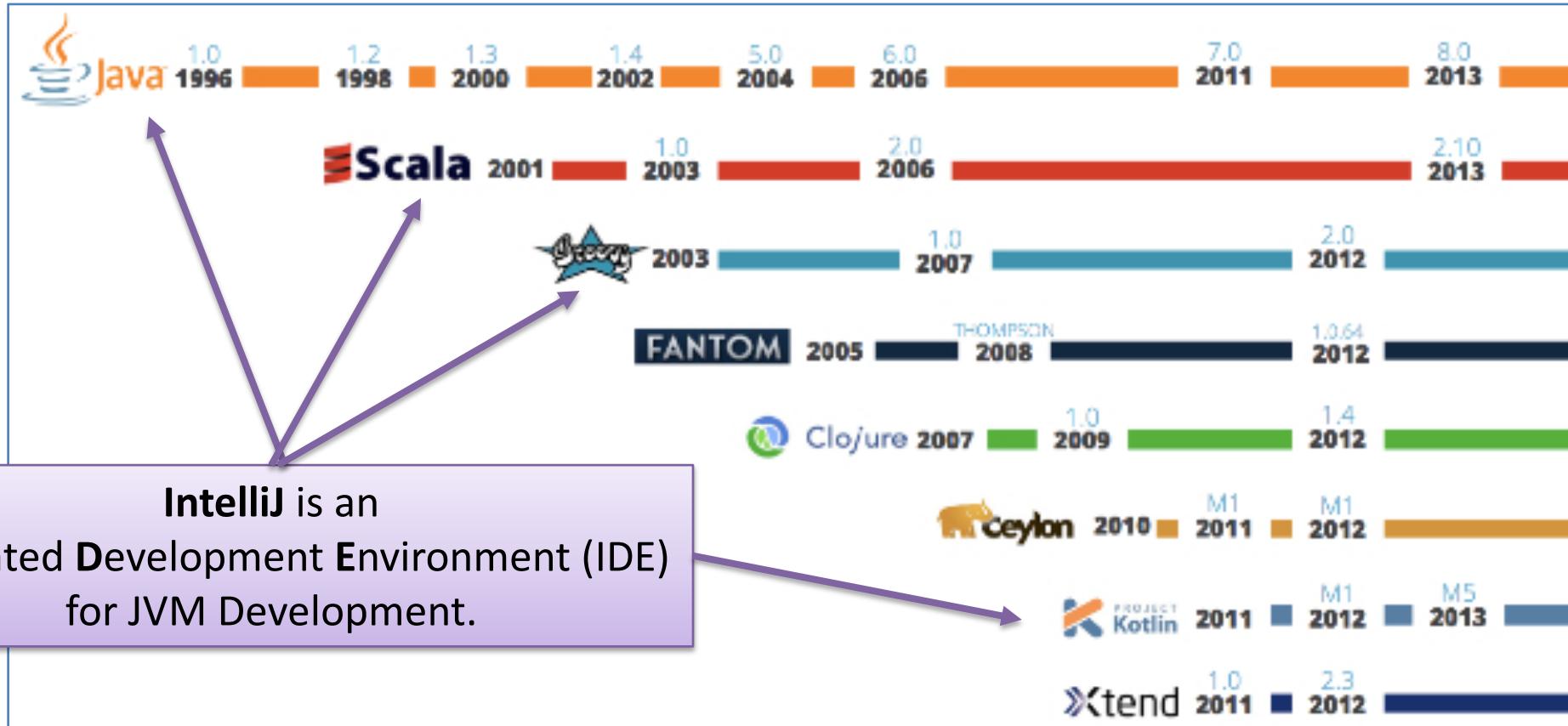
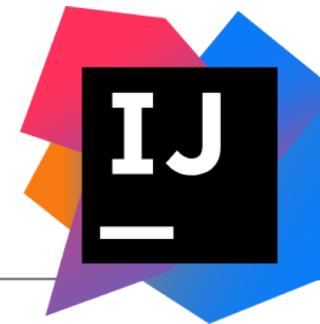


JVM Language History

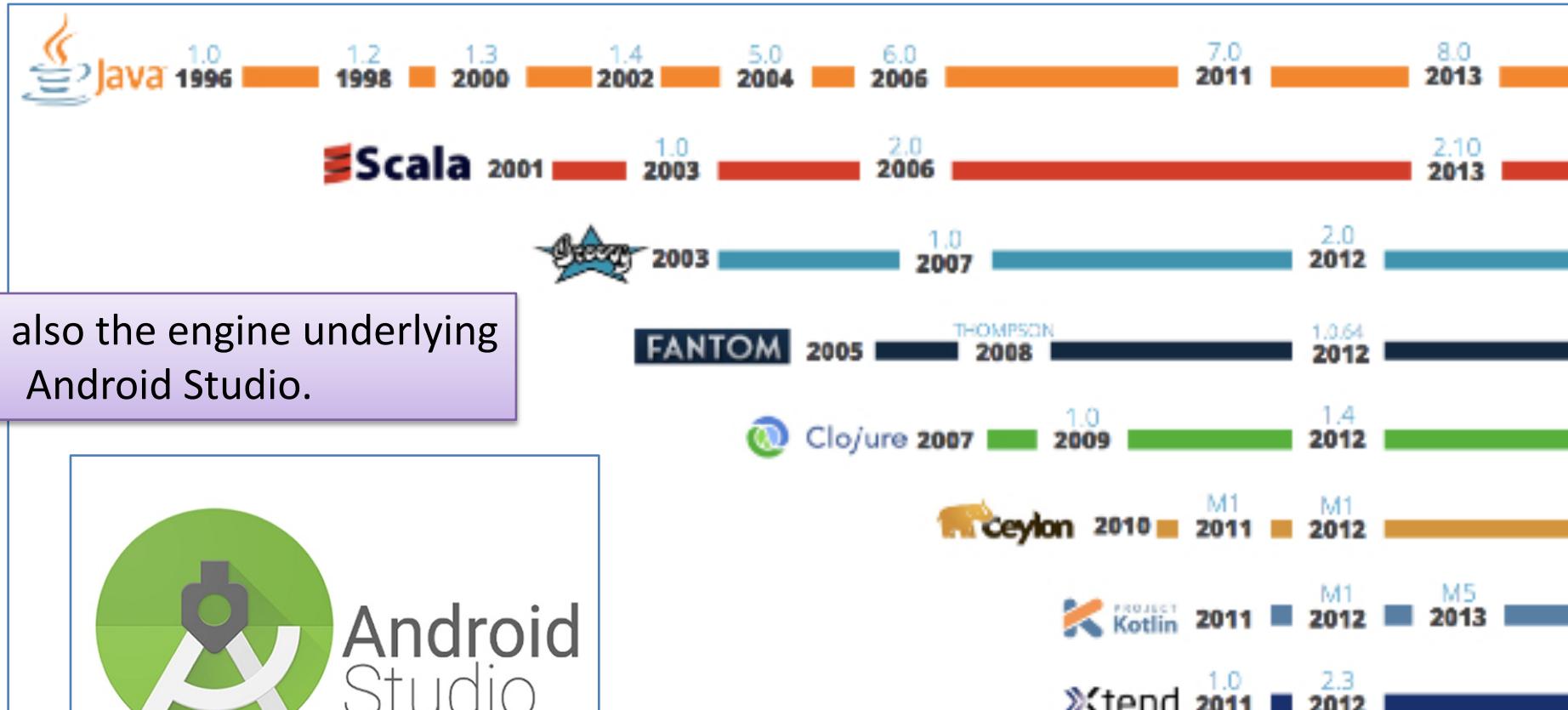
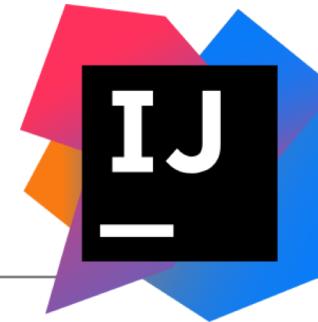


[Source](#)

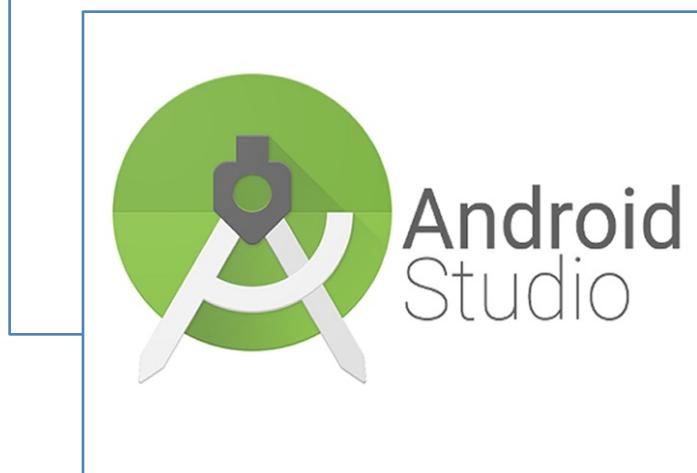
IntelliJ and the JVM



IntelliJ and the JVM

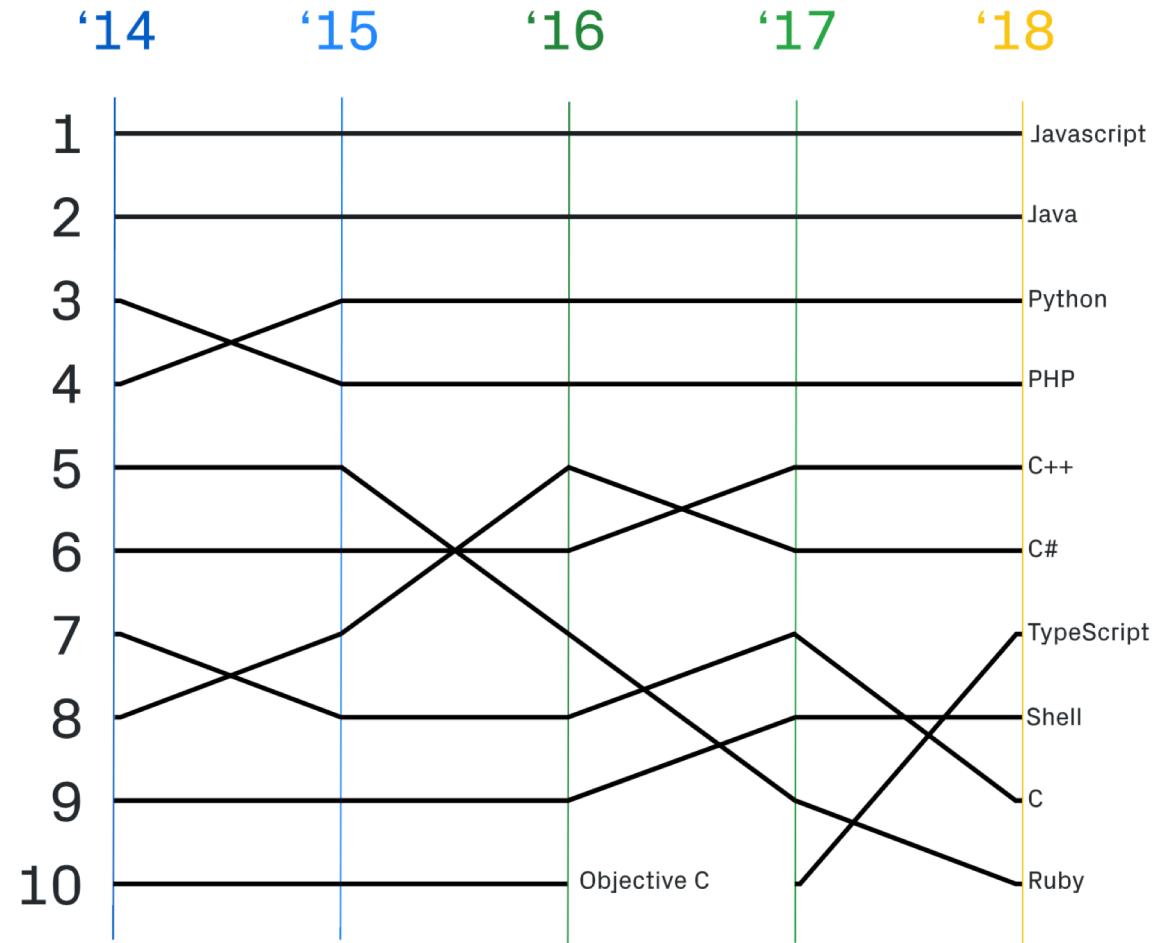


IntelliJ is also the engine underlying
Android Studio.



GitHub Most Popular Languages

- octoverse



Installing IntelliJ

BEFORE you install IntelliJ
Download and Install the **JDK** first using
IE on windows or Safari on OSX!



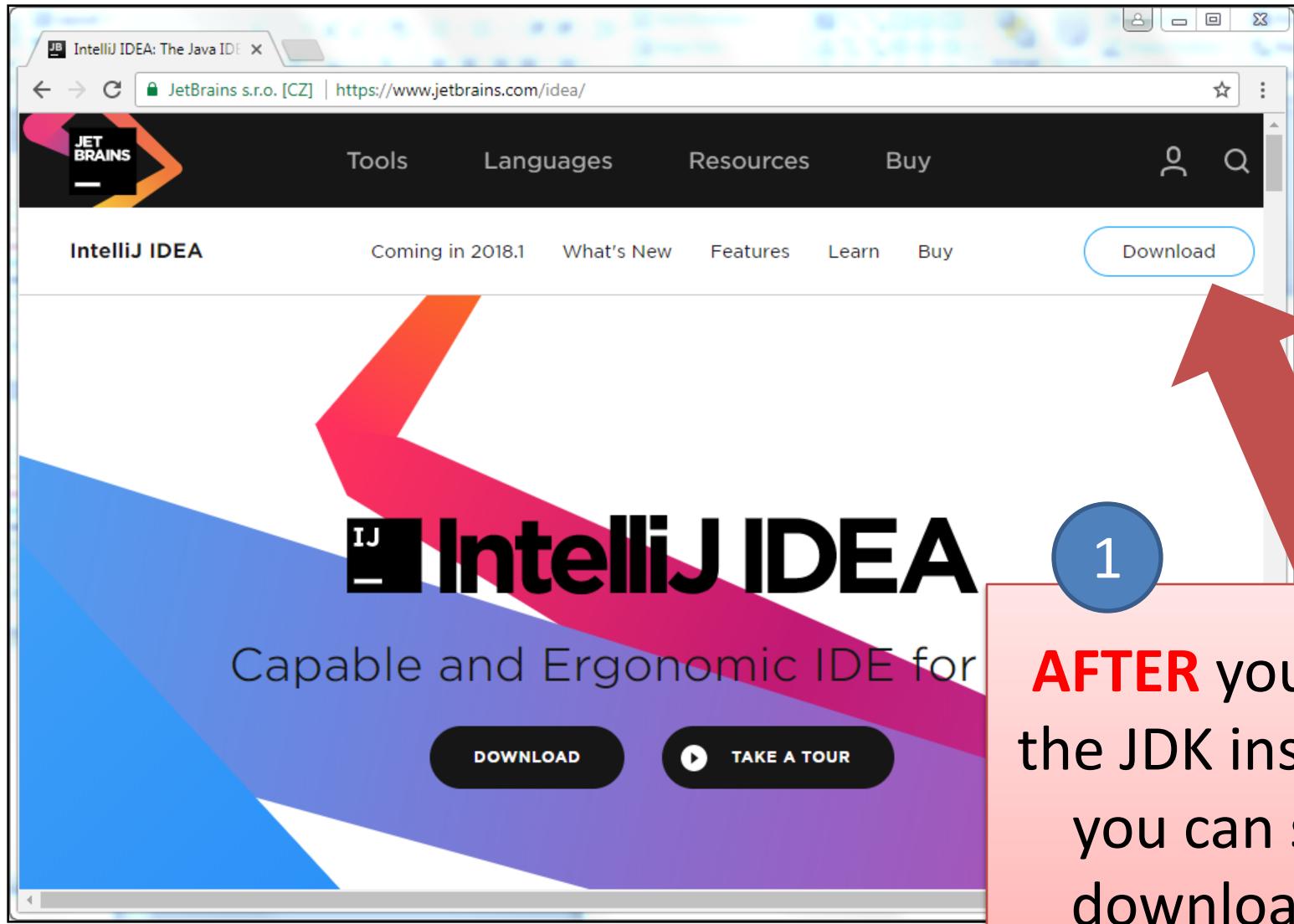
- 2 IntelliJ Editions:**
- **Community** (free) is a subset of
 - **Ultimate** (paid)



As a WIT student you can use **ULTIMATE**

The following slides show how to install Community Edition.

[Lab 07a](#) demonstrates similar steps to **install Ultimate instead**. Use Ultimate!



1

AFTER you have
the JDK installed,
you can start
downloading
IntelliJ.

JB Download IntelliJ IDEA: T

JetBrains s.r.o. [CZ] | https://www.jetbrains.com/idea/download/#section=windows

JET BRAINS Tools Languages Resources Buy

IntelliJ IDEA Coming in 2018.1 What's New Features Learn Buy Download

2 download (.exe)

The screenshot shows the IntelliJ IDEA download page on the JetBrains website. It features two main download options: 'Ultimate' and 'Community'. The 'Community' section is highlighted with a large red arrow pointing to its '.EXE' download link. A blue circle with the number '2' is positioned above the 'Community' section. The 'Ultimate' section includes details about its use for web, mobile, and enterprise development, along with its own download links. The bottom of the page displays system requirements, installation instructions, and previous versions information.

Version: 2017.3.4
Build: 173.4548.28
Released: January 30, 2018
[Release notes](#)

[System requirements](#) [Installation Instructions](#) [Previous versions](#)

Ultimate
Web, mobile and enterprise development

[DOWNLOAD](#) [.EXE](#)

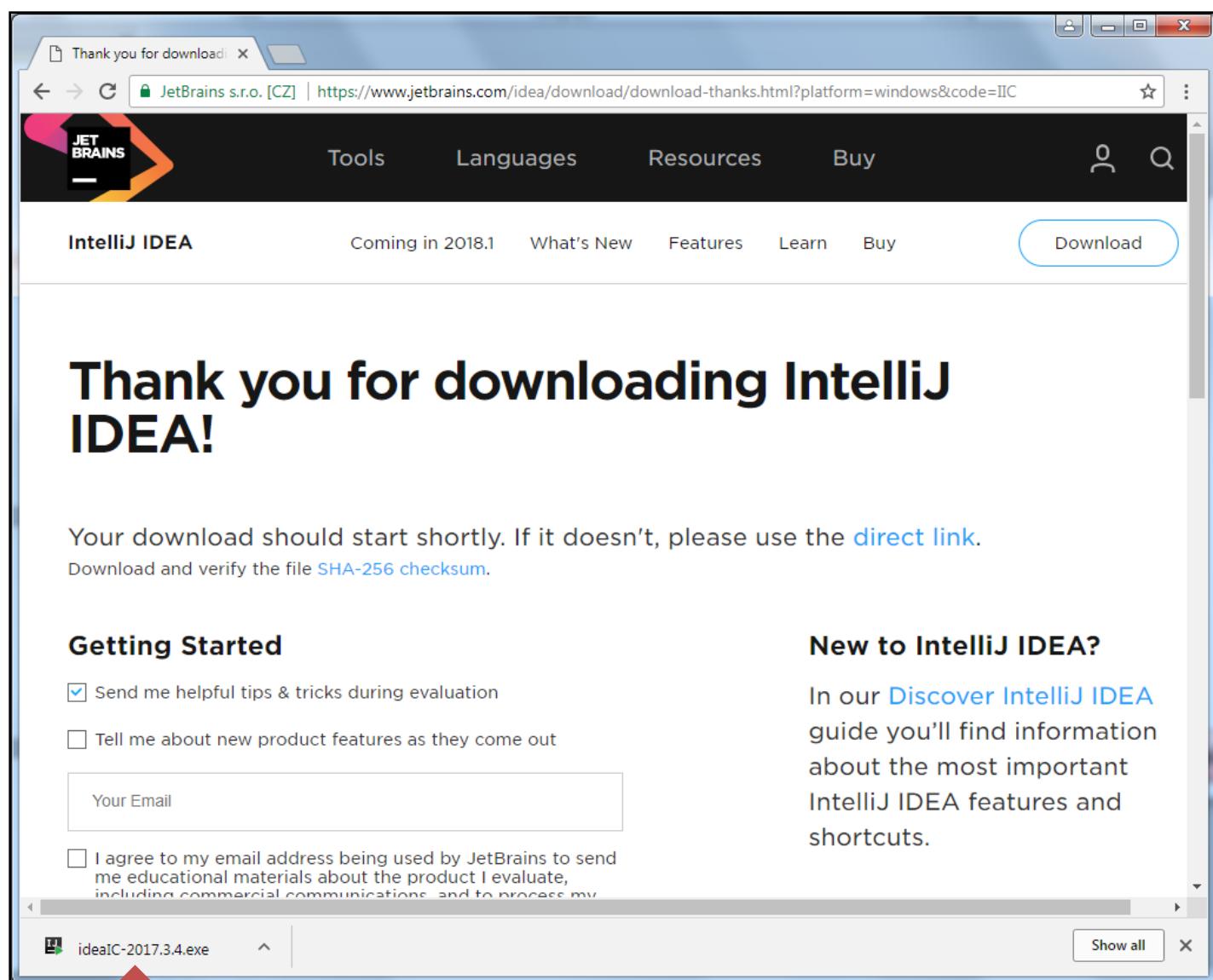
Free trial

Community
Java, Groovy, Scala and Android development

[DOWNLOAD](#) [.EXE](#)

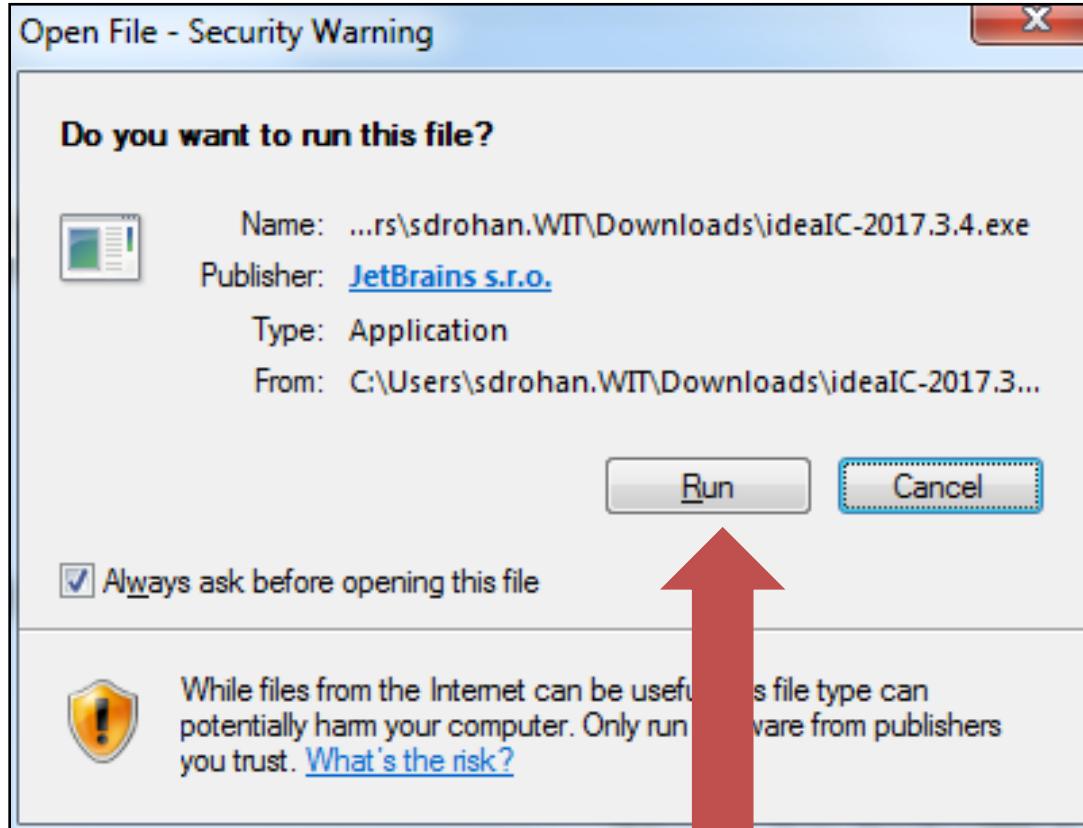
Free, open-source

License	Commercial	Open-source, Apache 2.0
Java, Kotlin, Groovy, Scala	✓	✓
Android	?	✓



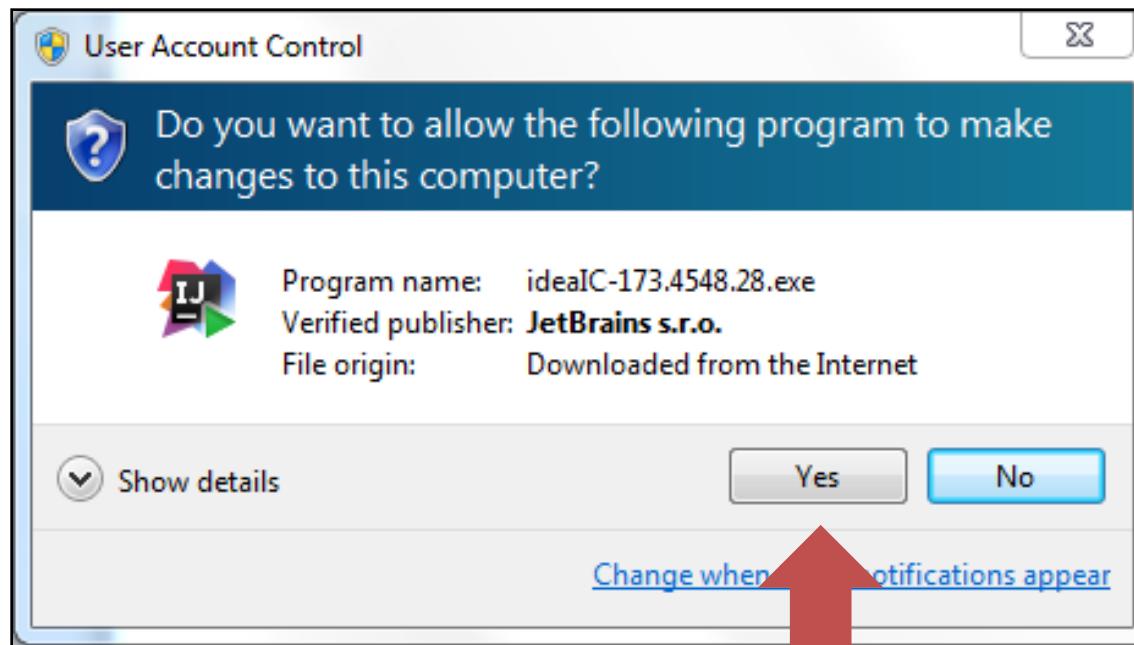
3

Click on the downloaded file to start the installation.



4

If you are presented with a security warning, click **Run**.



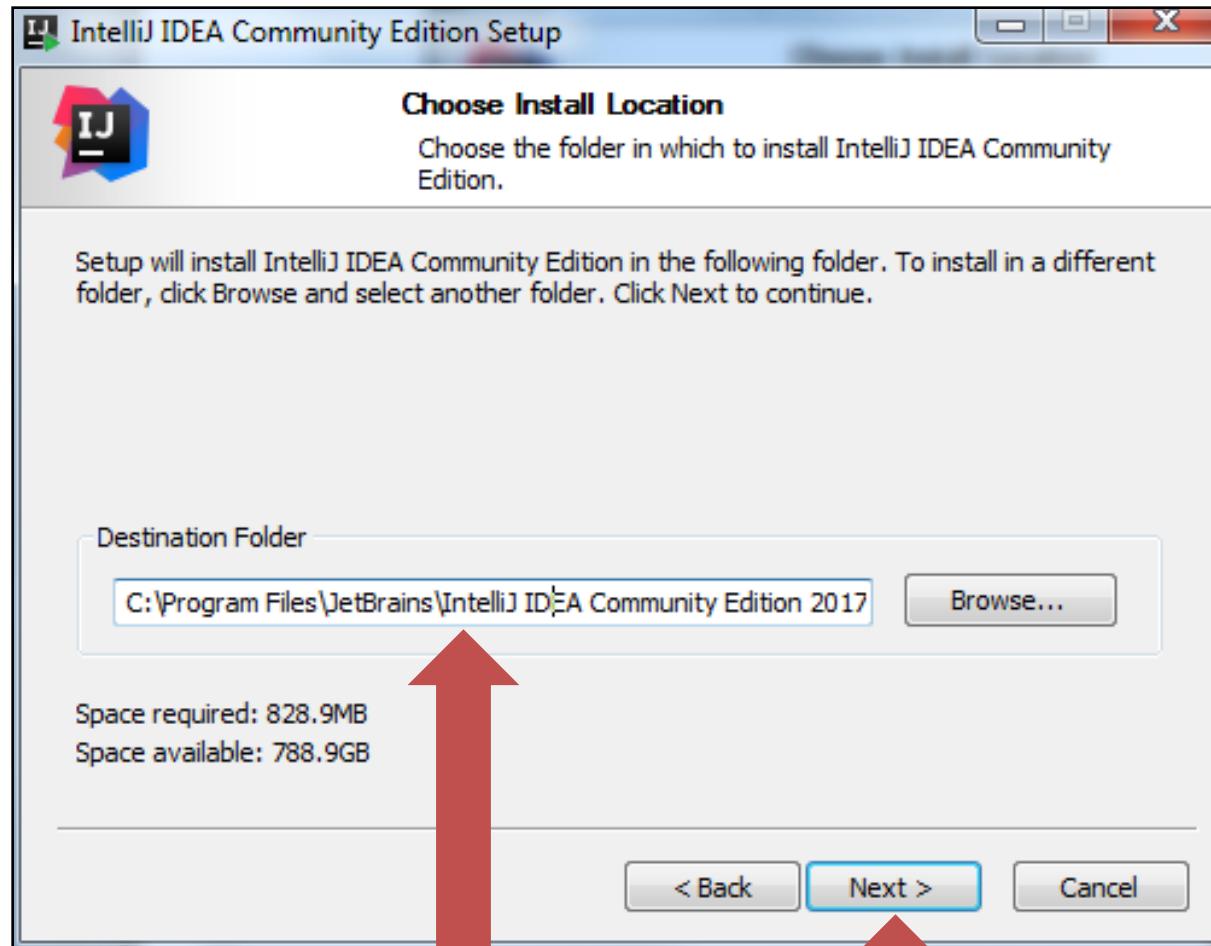
5

If you are presented with this warning, click Yes.



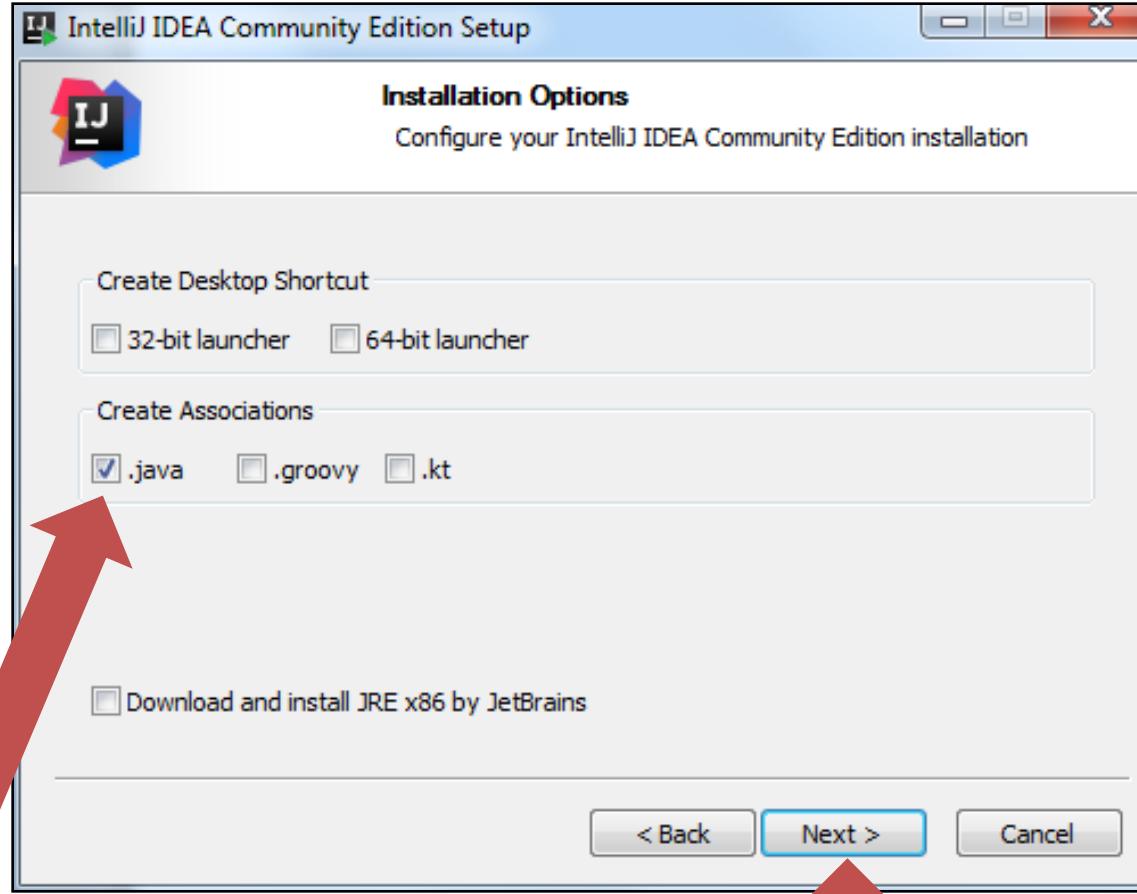
6

Click **Next** to continue...



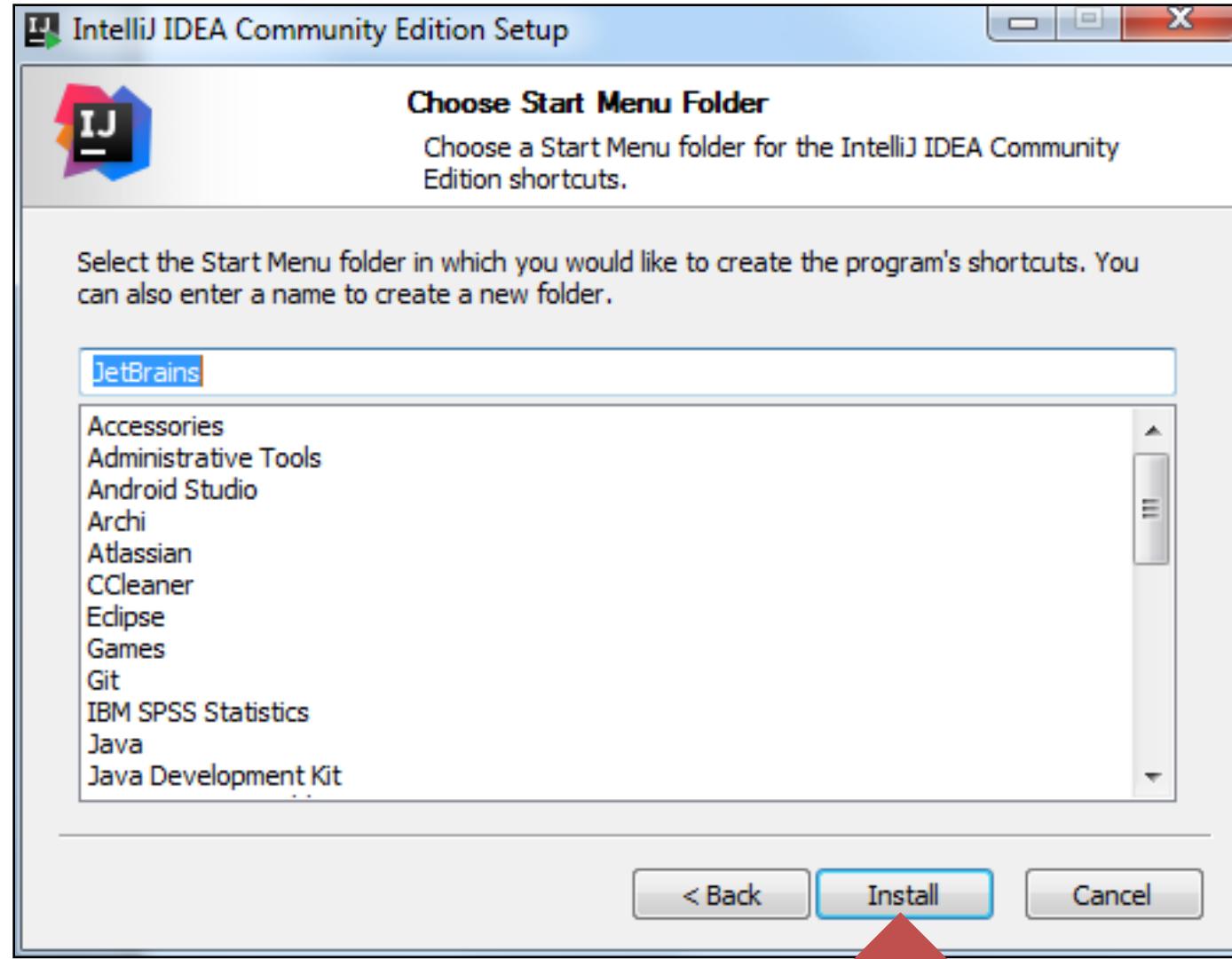
7

Accept the default destination folder
(you can change this if you prefer) and click **Next**.



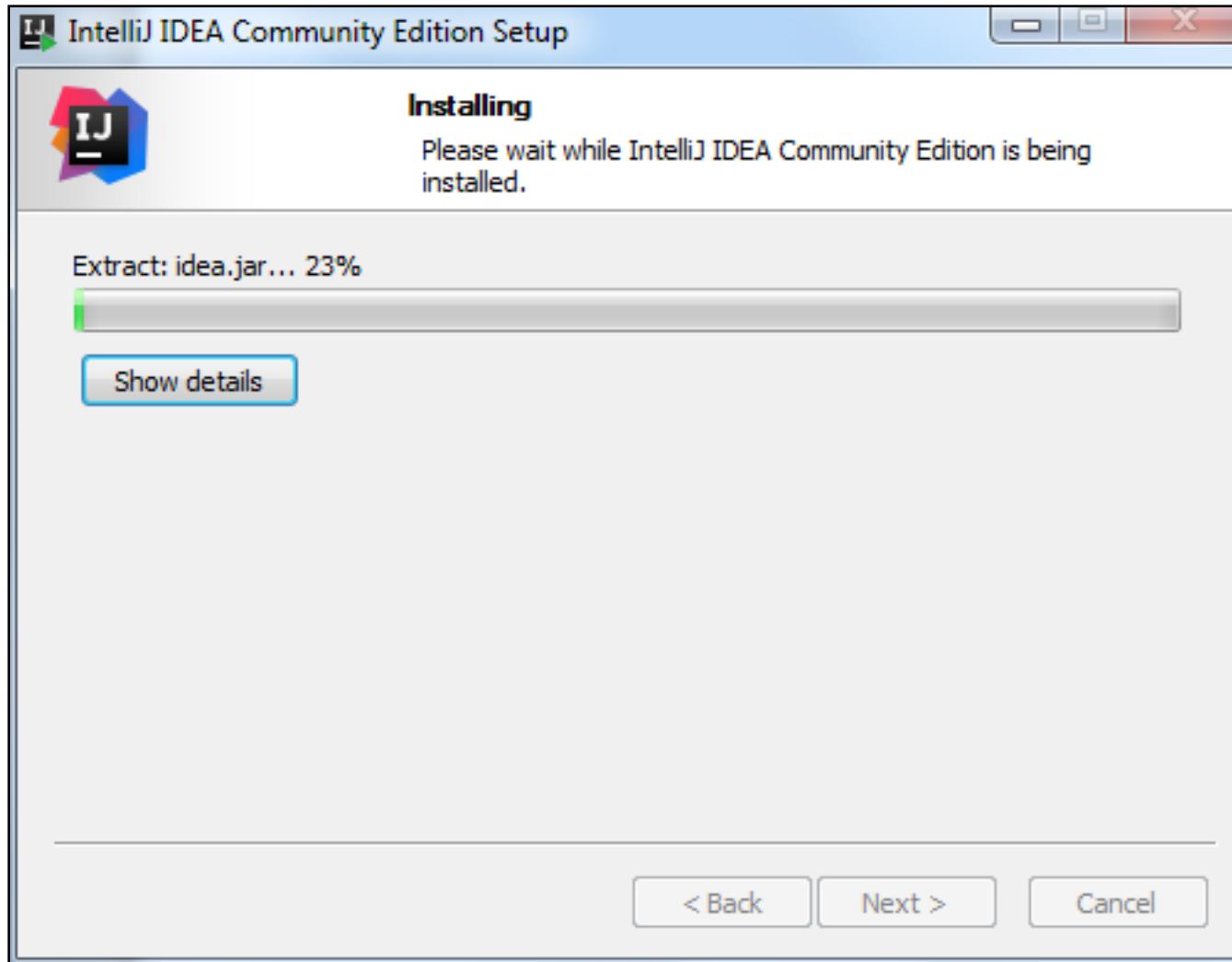
8

Tick the **.java** box to create a file association.
Click the **Next** button.



9

Click **Install** to continue...



10

The installation is in progress...

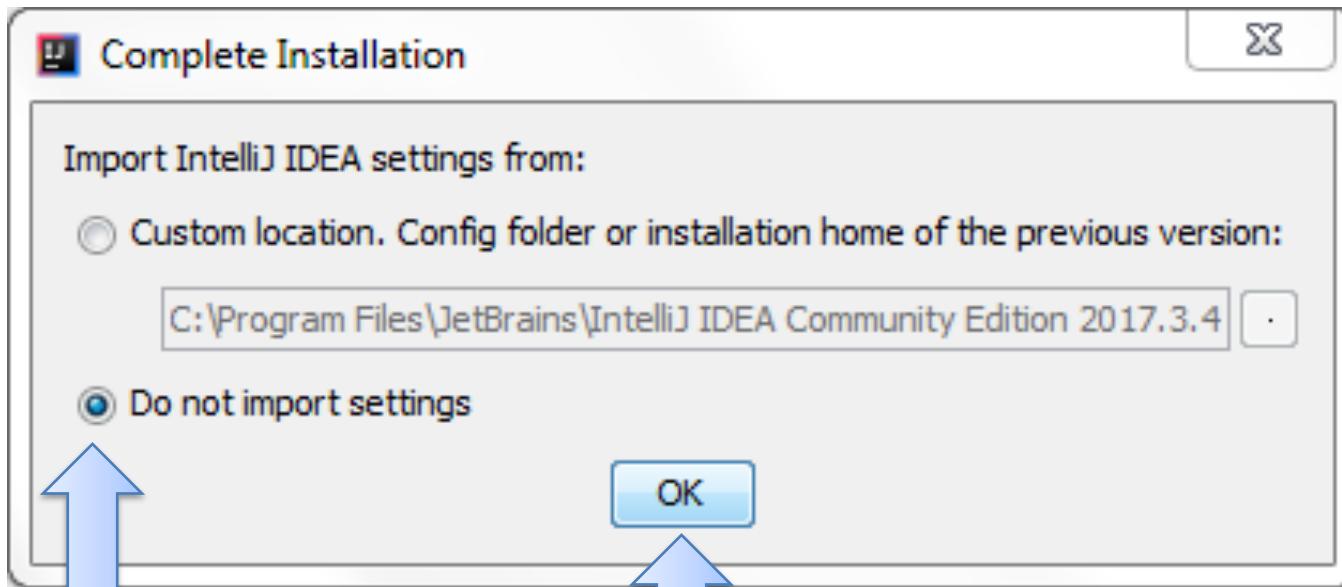


11

The installation is done...click the **Finish** button

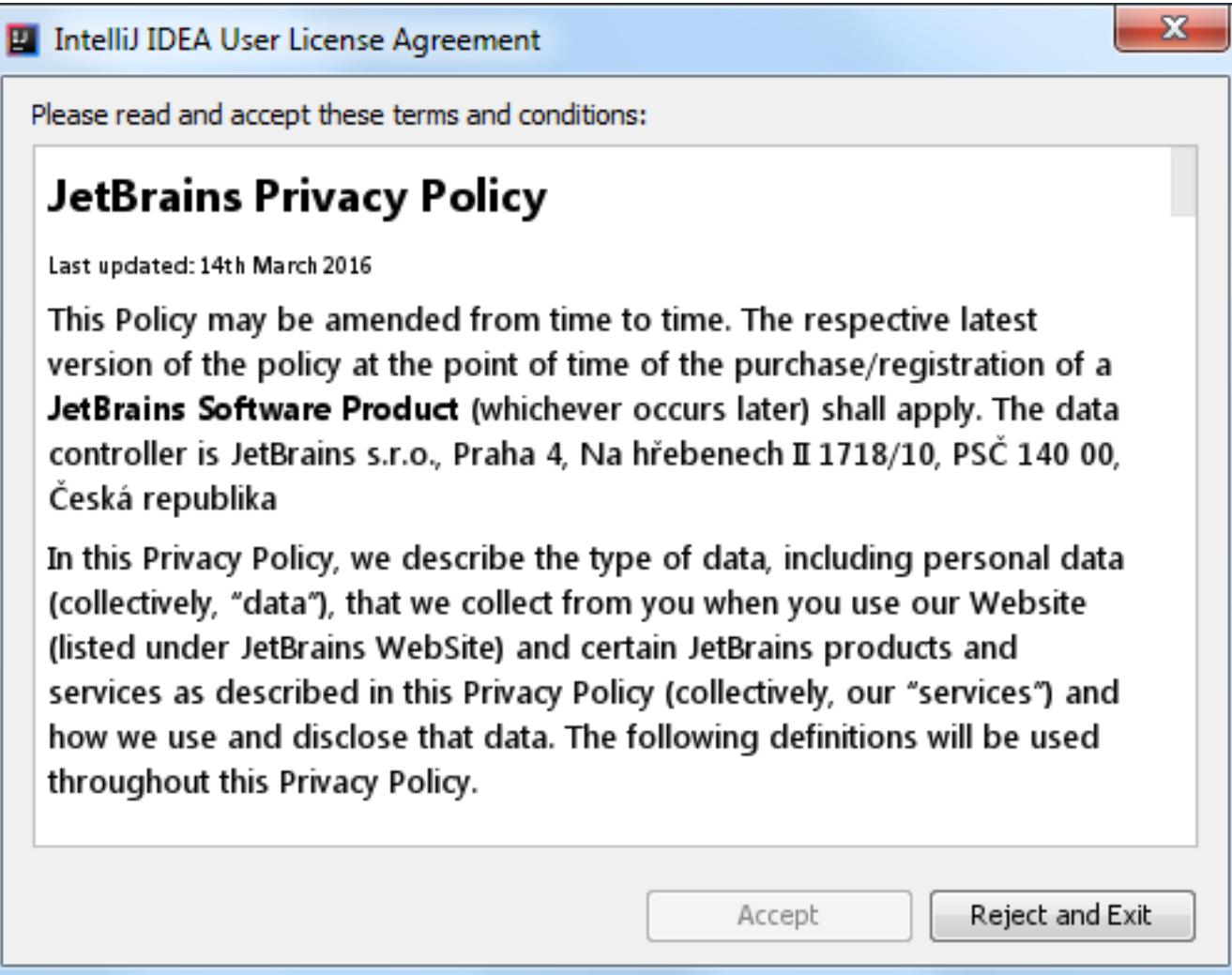
*Now that you have installed the
JDK and IntelliJ,
let's create and run our first
project.*

*We will print **HelloWorld!** to the
console.*



1

Appears when you first start IntelliJ.
Don't import the settings. Click OK.



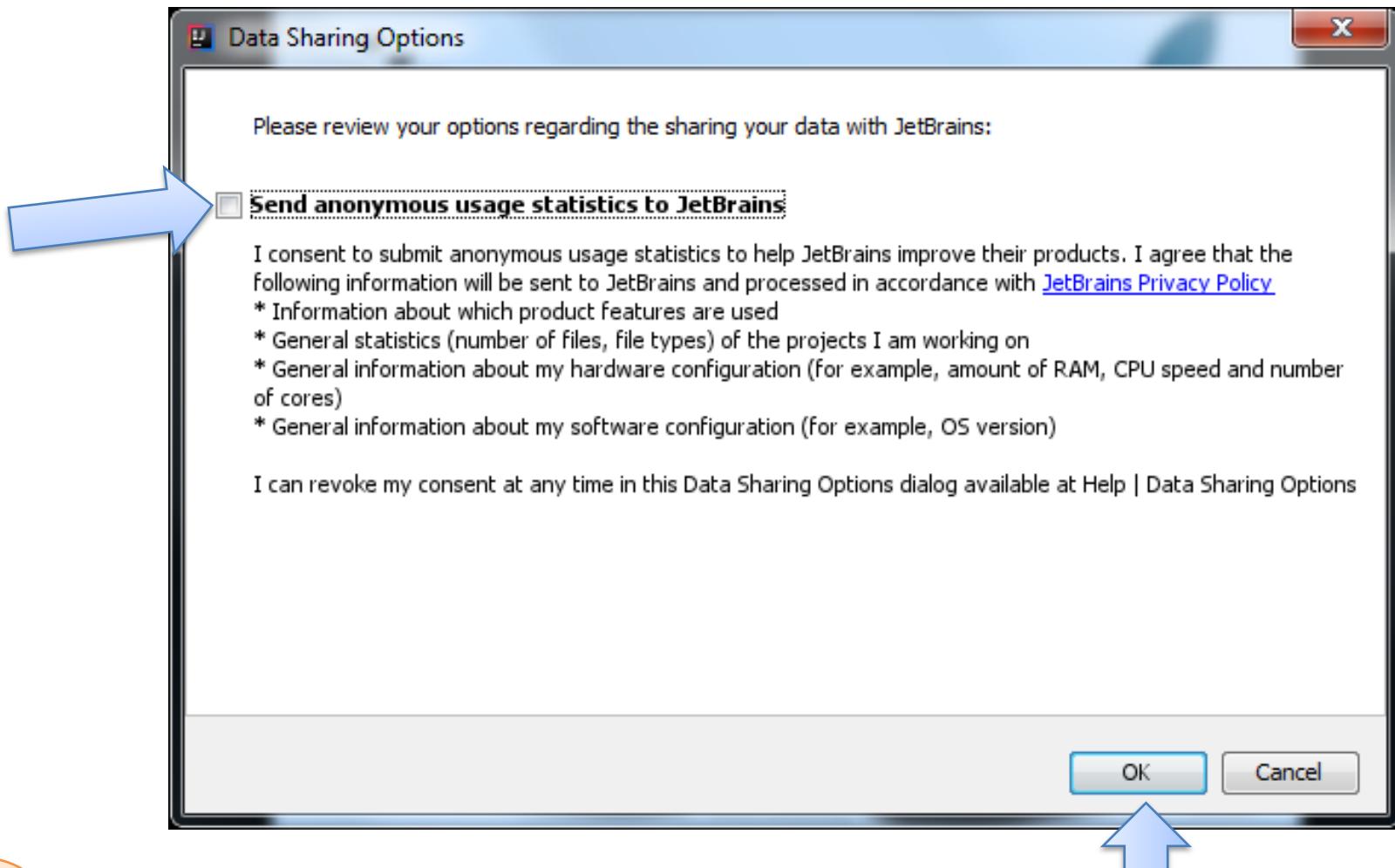
2

Scroll down to the end of the User Licence Agreement (ULA)...



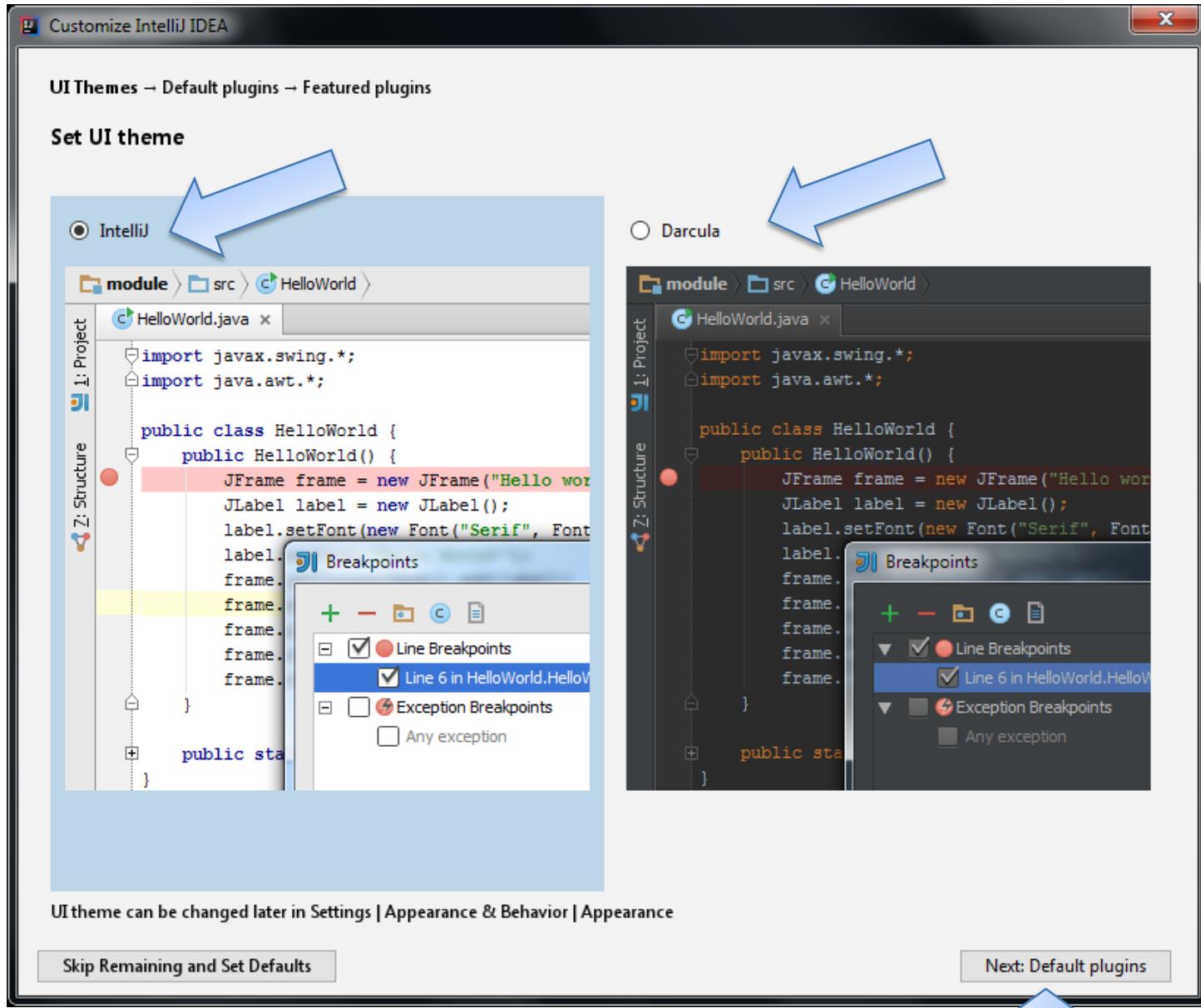
3

When the Accept button appears, click it.



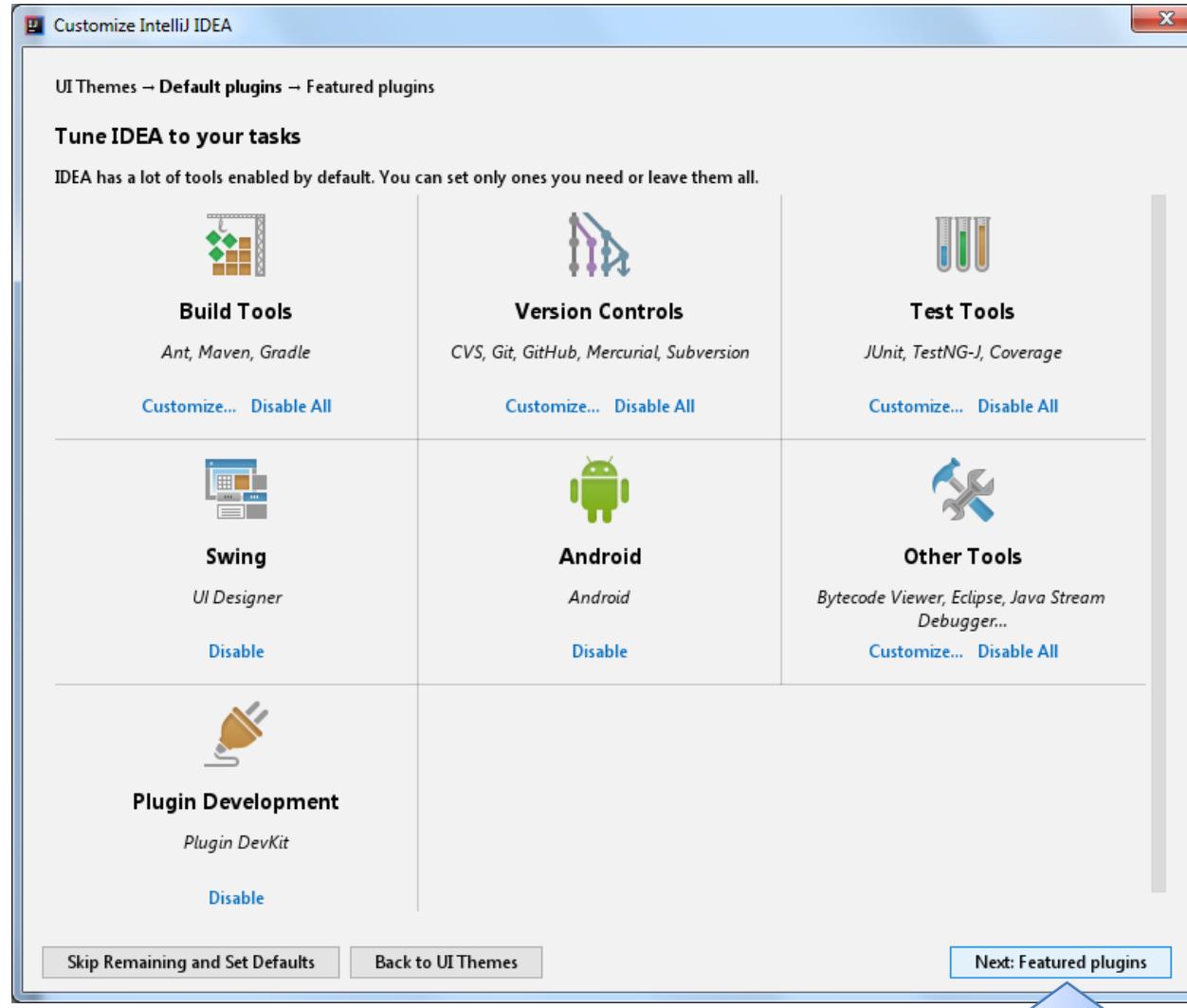
4

Decide if you want to share anonymous data with Jetbrains and click the OK button.



5

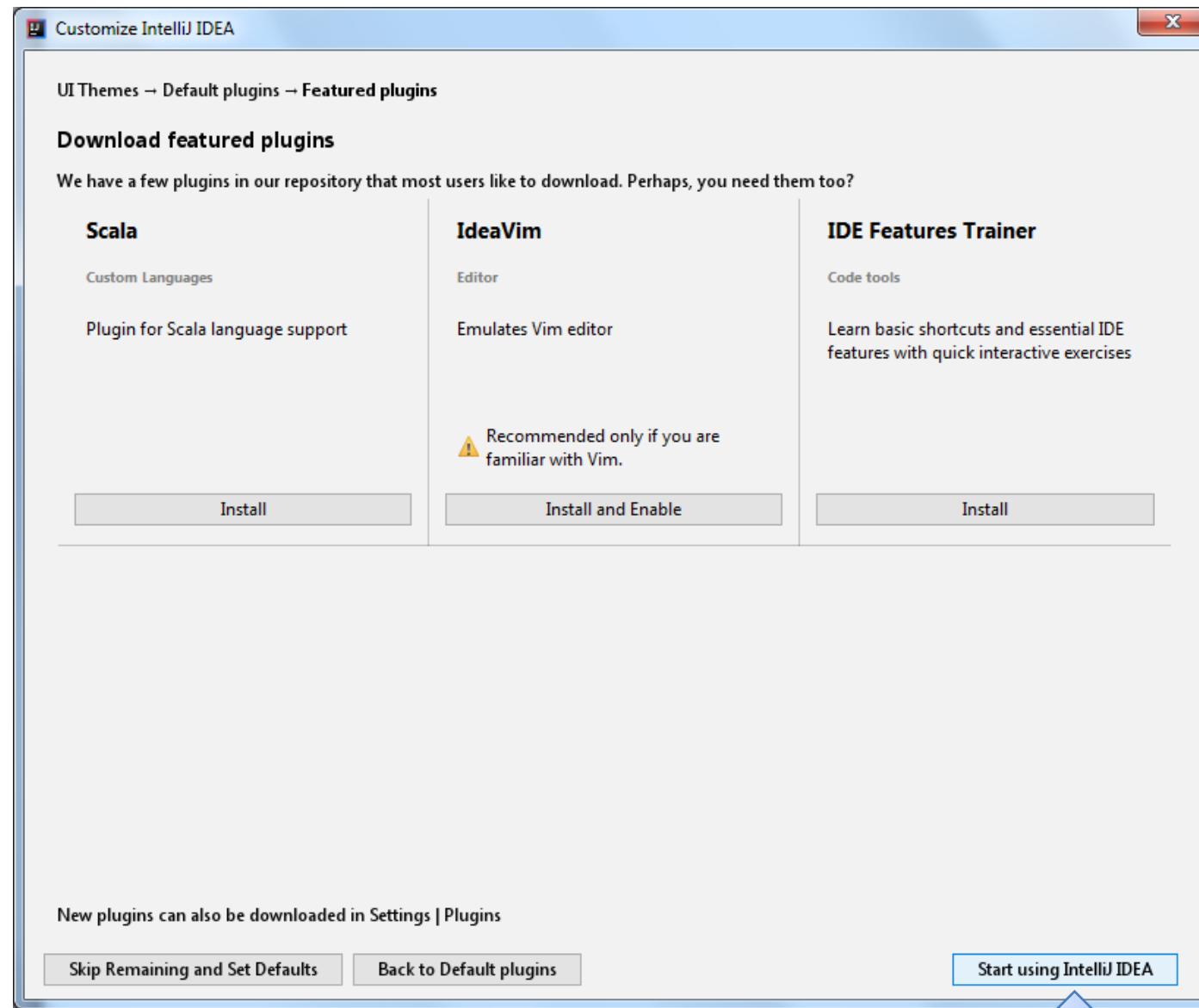
Choose your preferred theme and click Next.



6

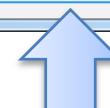
You will come across a lot of these plugins, in some form or other, during the course.

At the moment, we won't be using them, so click **Next**.



7

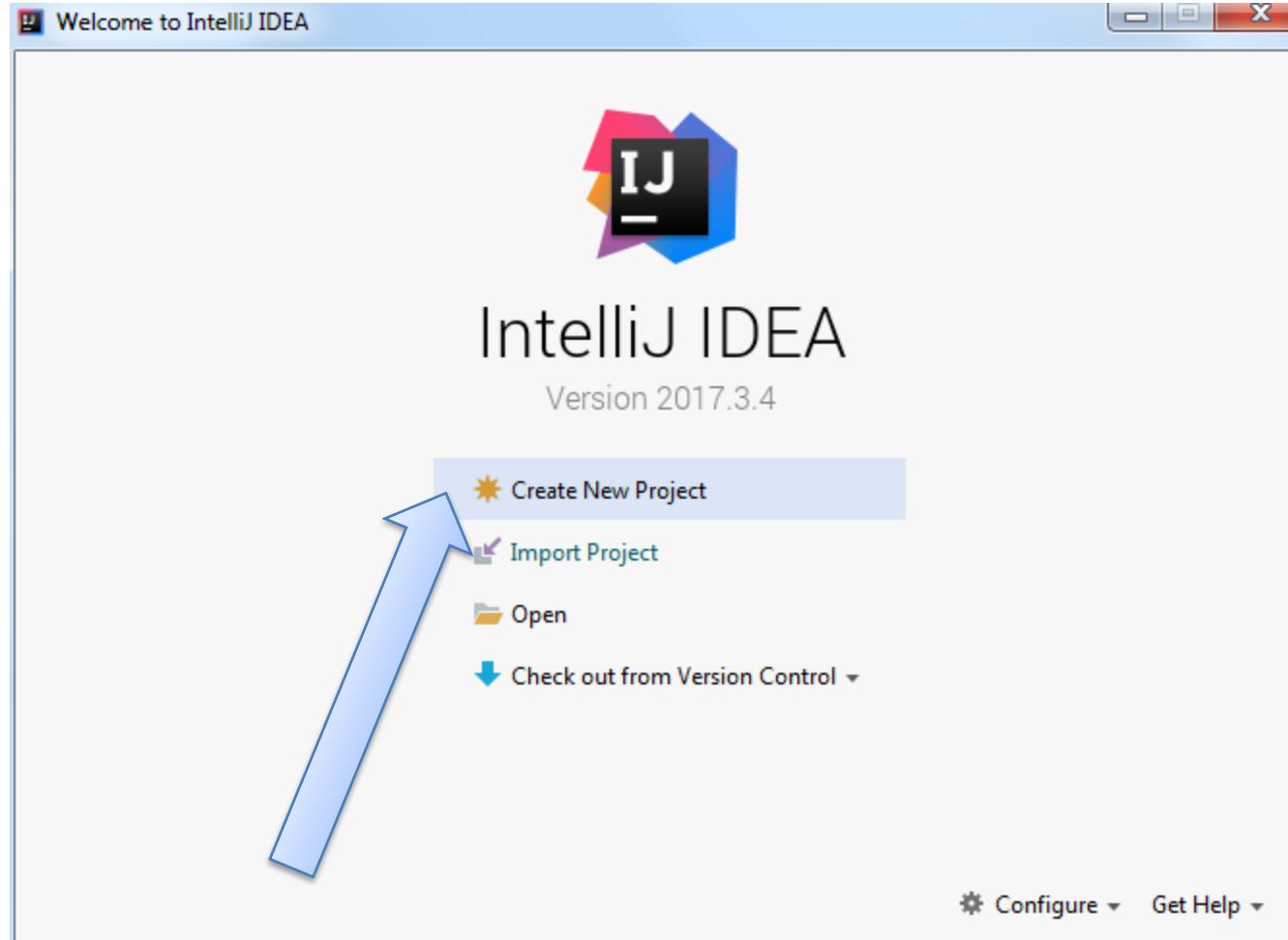
Same again here...we aren't using these plugins, so click Start...





8

You will see this splash screen and then...

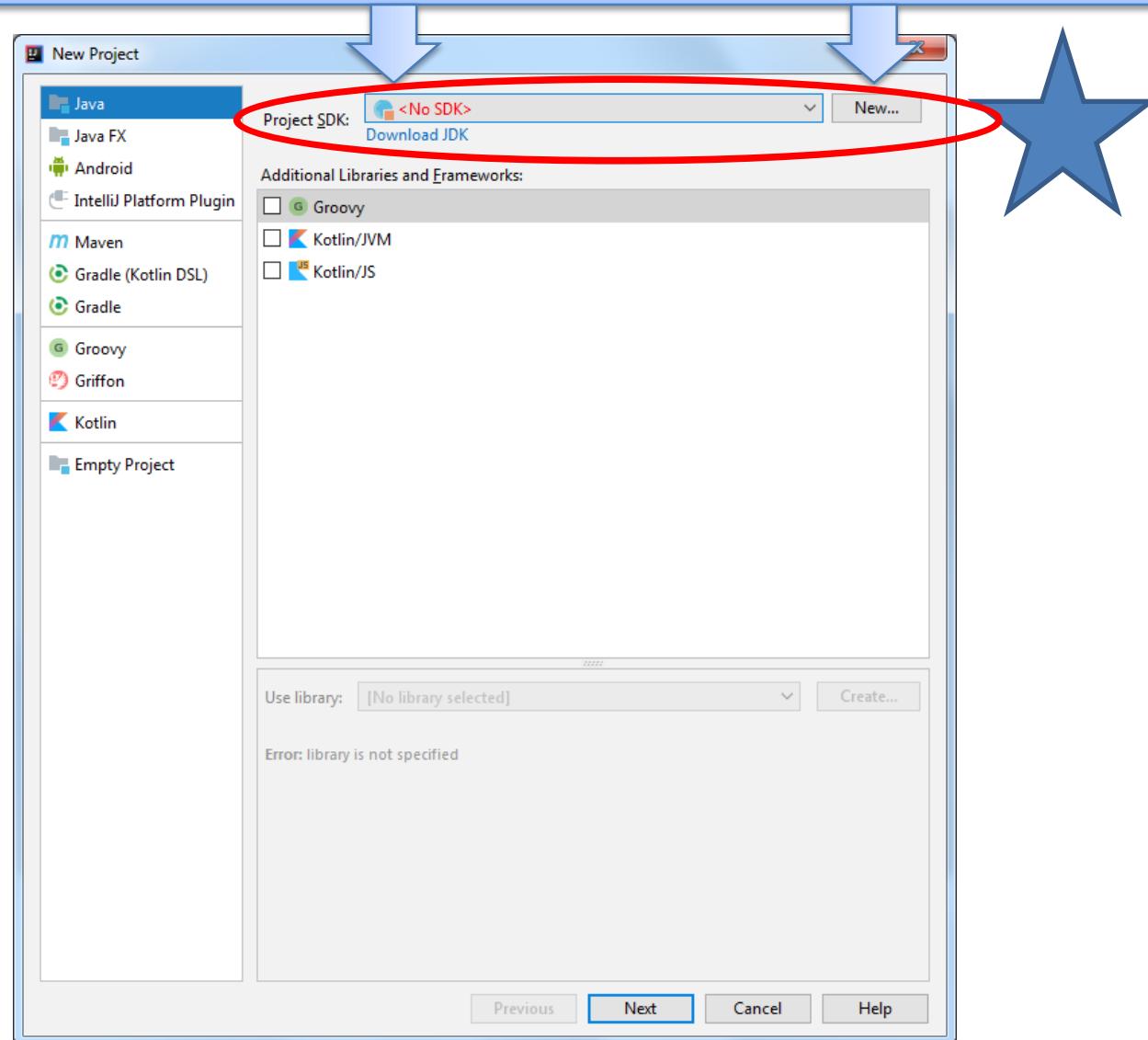


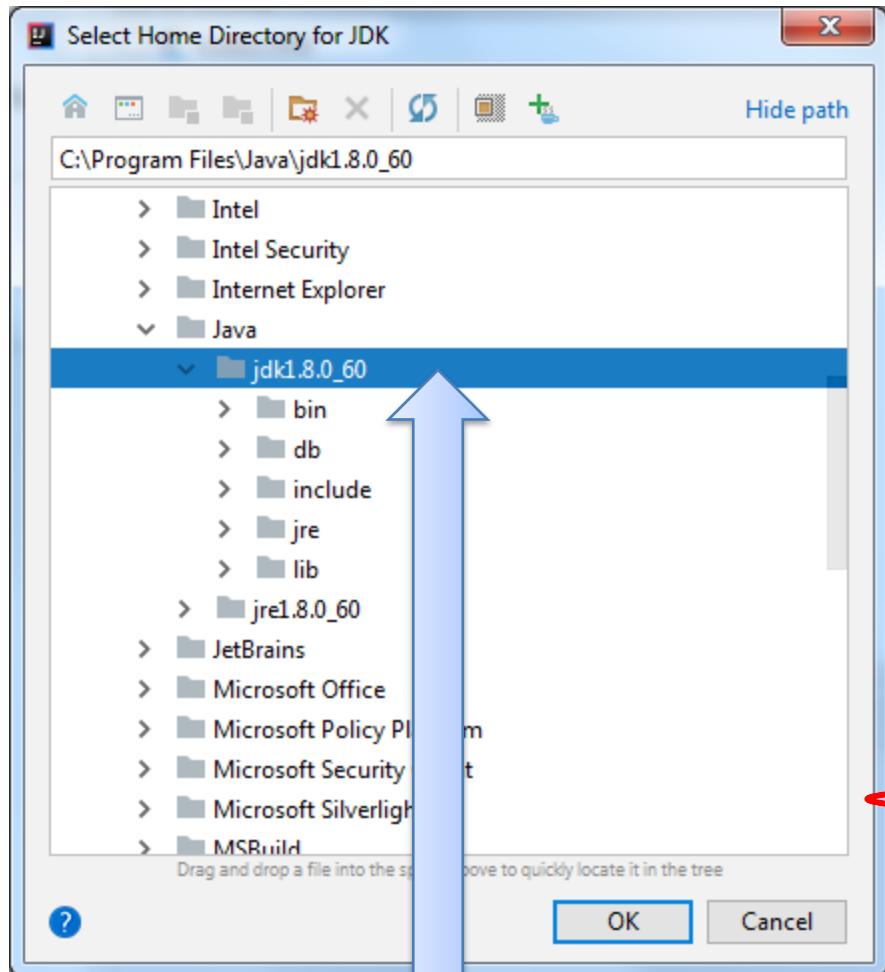
9

IntelliJ is now running...
click on the **Create New Project** link

10

You may get this...no SDK. If so, click on the “New...” button





11

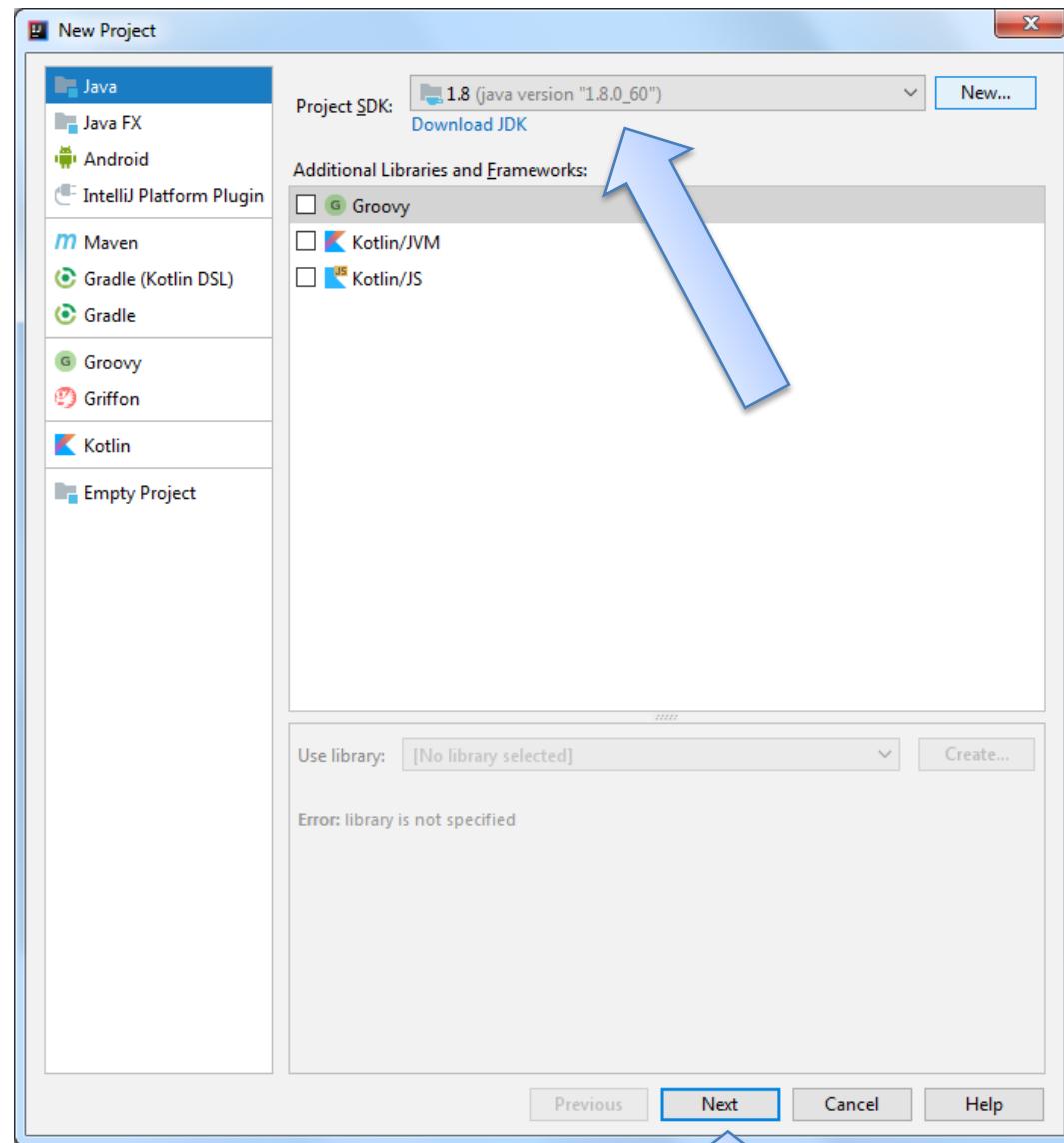
...and **locate** the version of java you previously installed. Click **Ok**

On OSX,
run a terminal window
and enter the following command

Echo \$(/usr/libexec/java_home)

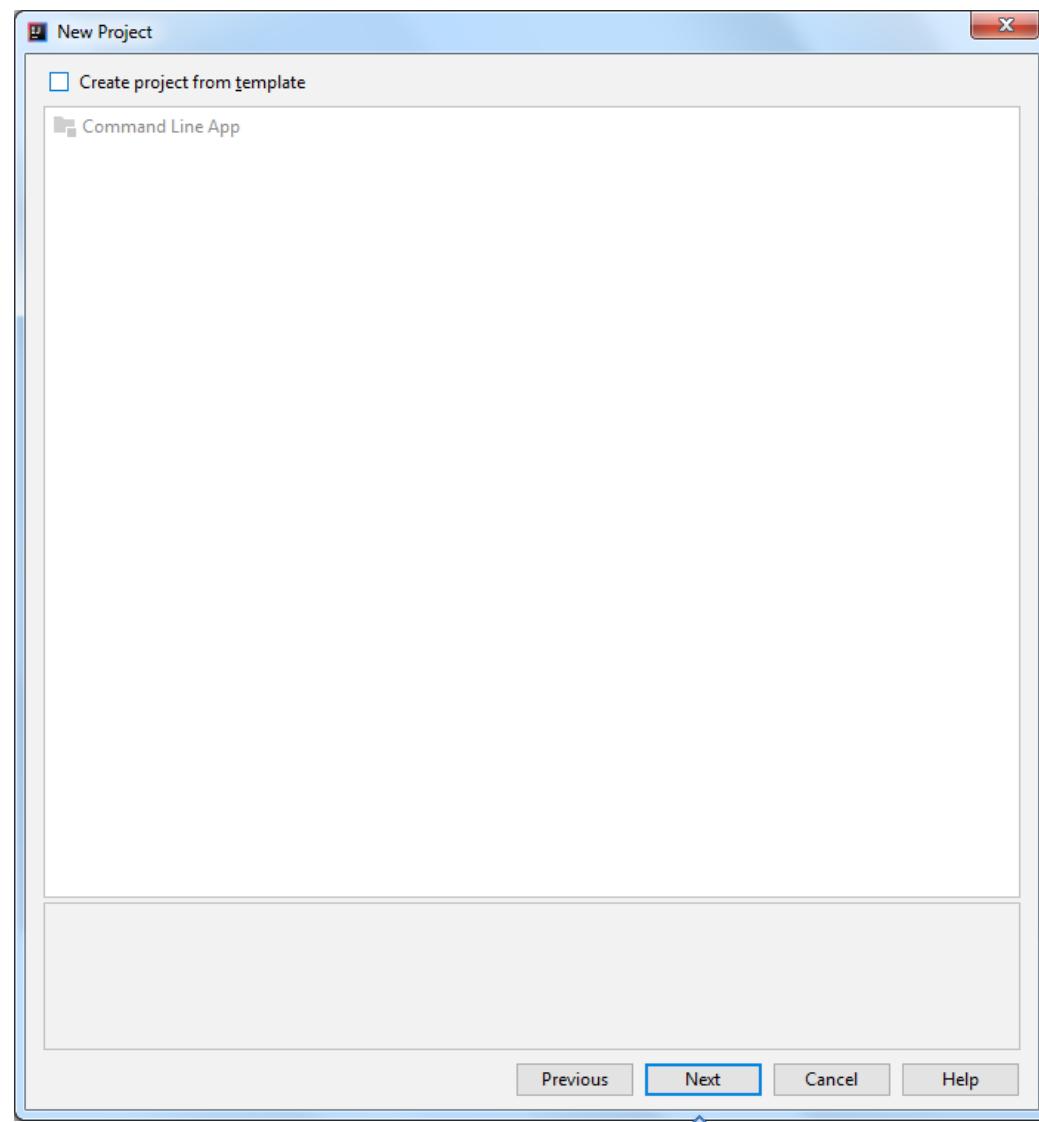
```
colmdunphy ~ bash 80x24
Last login: Sun Feb 18 15:47:12 on ttys000
[Colms-MacBook-Pro-i7-2:~ colmdunphv$ echo $(/usr/libexec/java_home)
/Library/Java/JavaVirtualMachines/jdk-9.0.4.jdk/Contents/Home
Colms-MacBook-Pro-i7-2:~ colmdunphv$
```

(IntelliJ already comes with JDK v8 labelled 1.8)



12

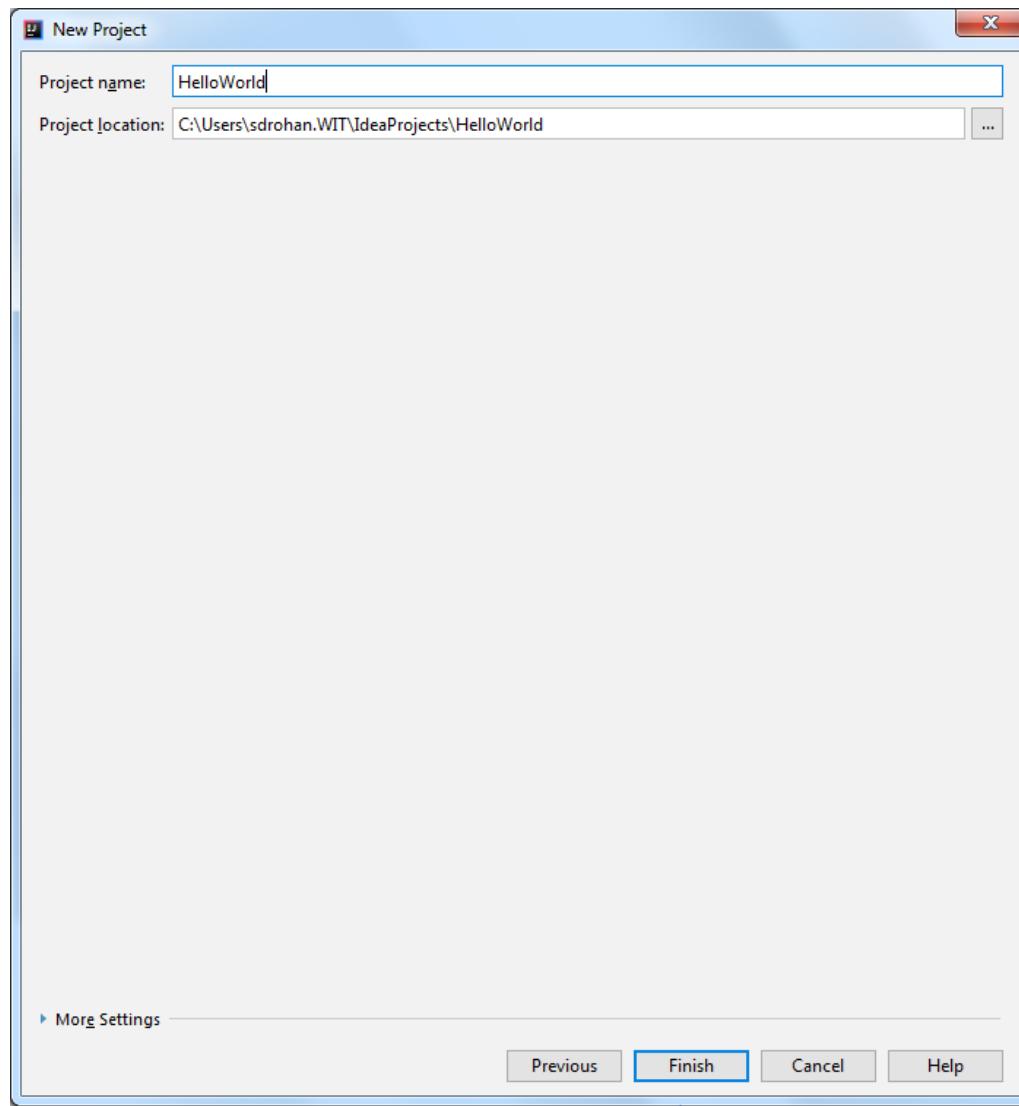
Now that Java is located, click Next.



13

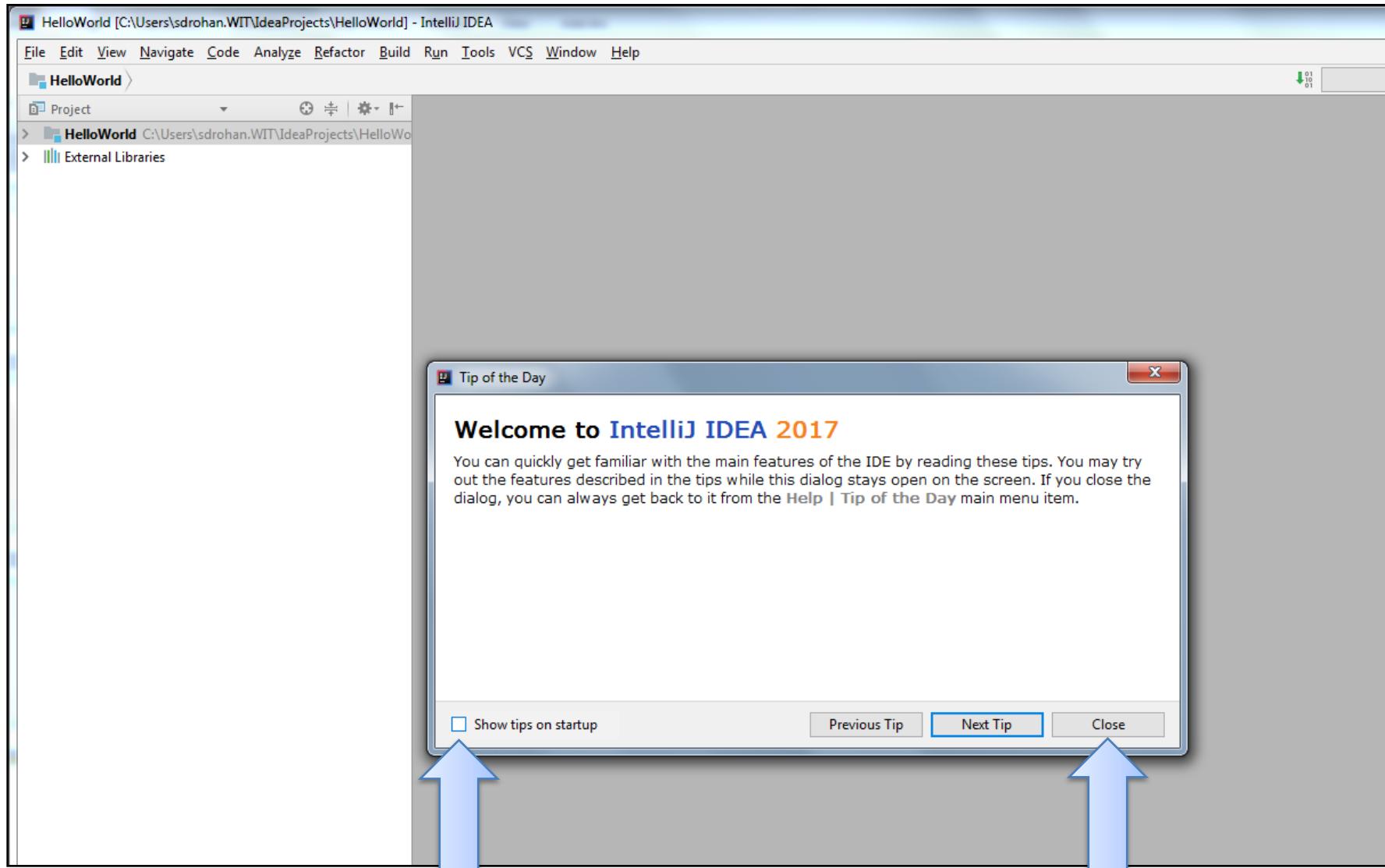


Accept the default settings by clicking Next



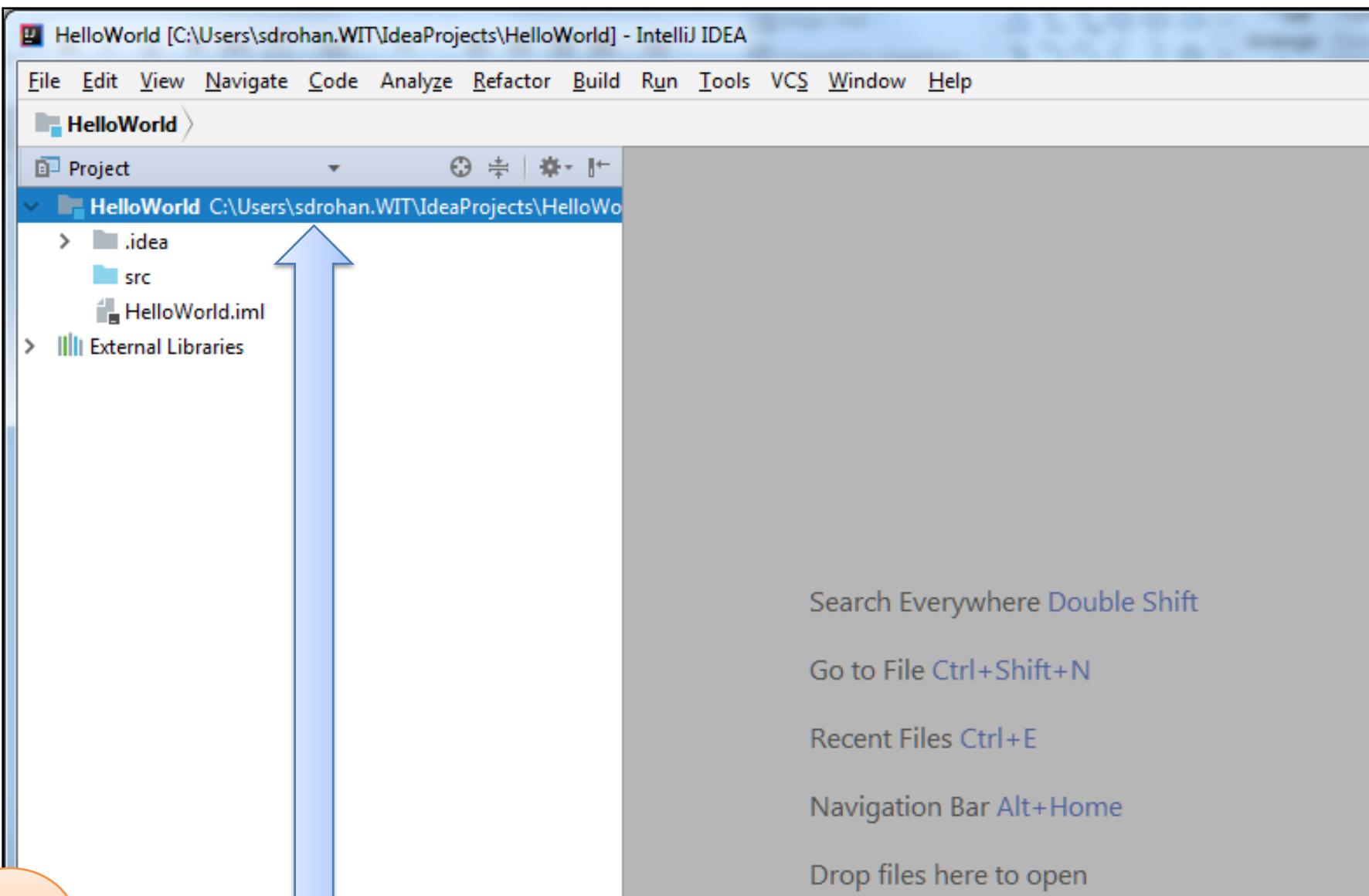
14

Enter project name (**HelloWorld**) and if you are happy with the default location, click Finish.



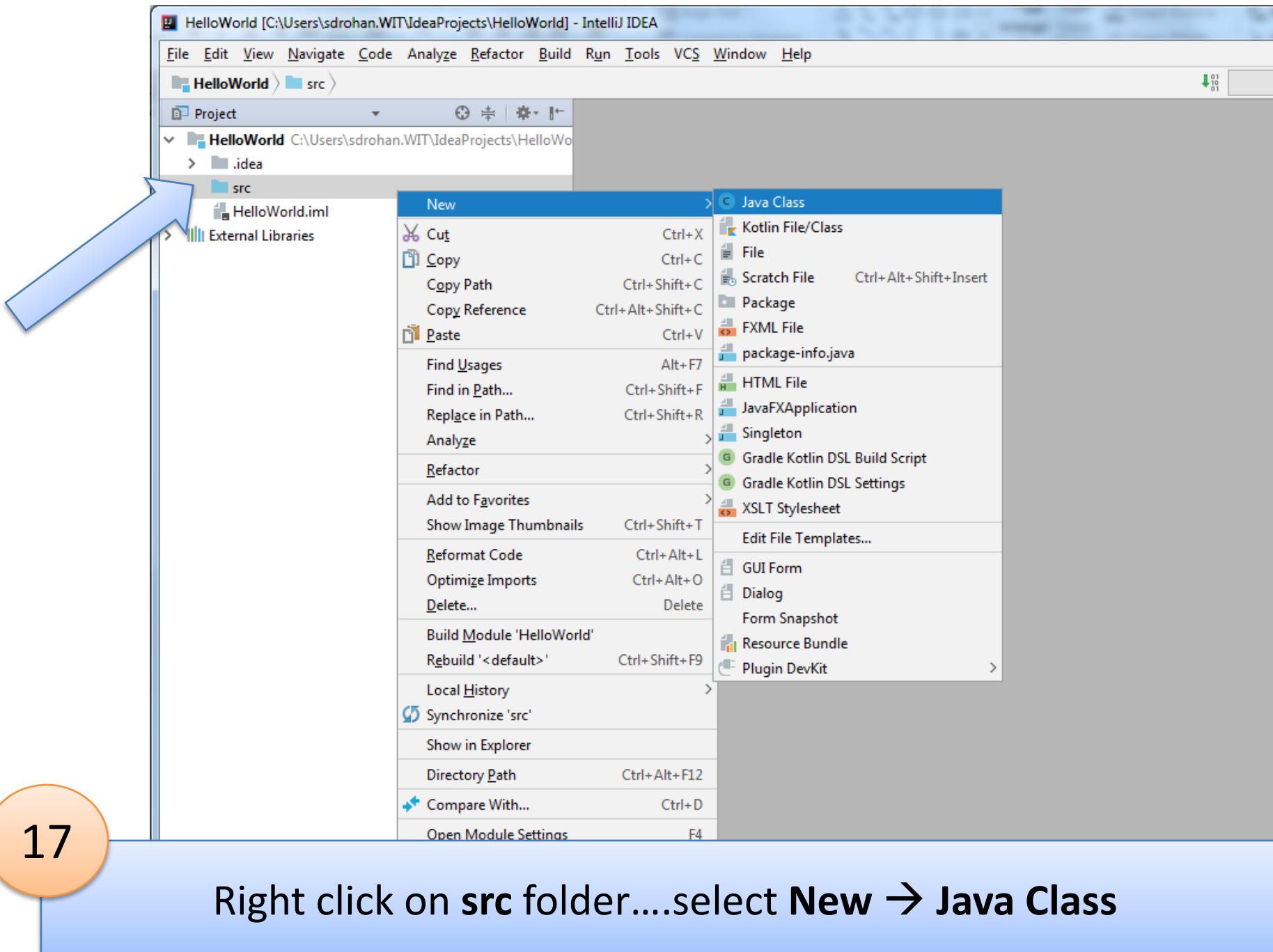
15

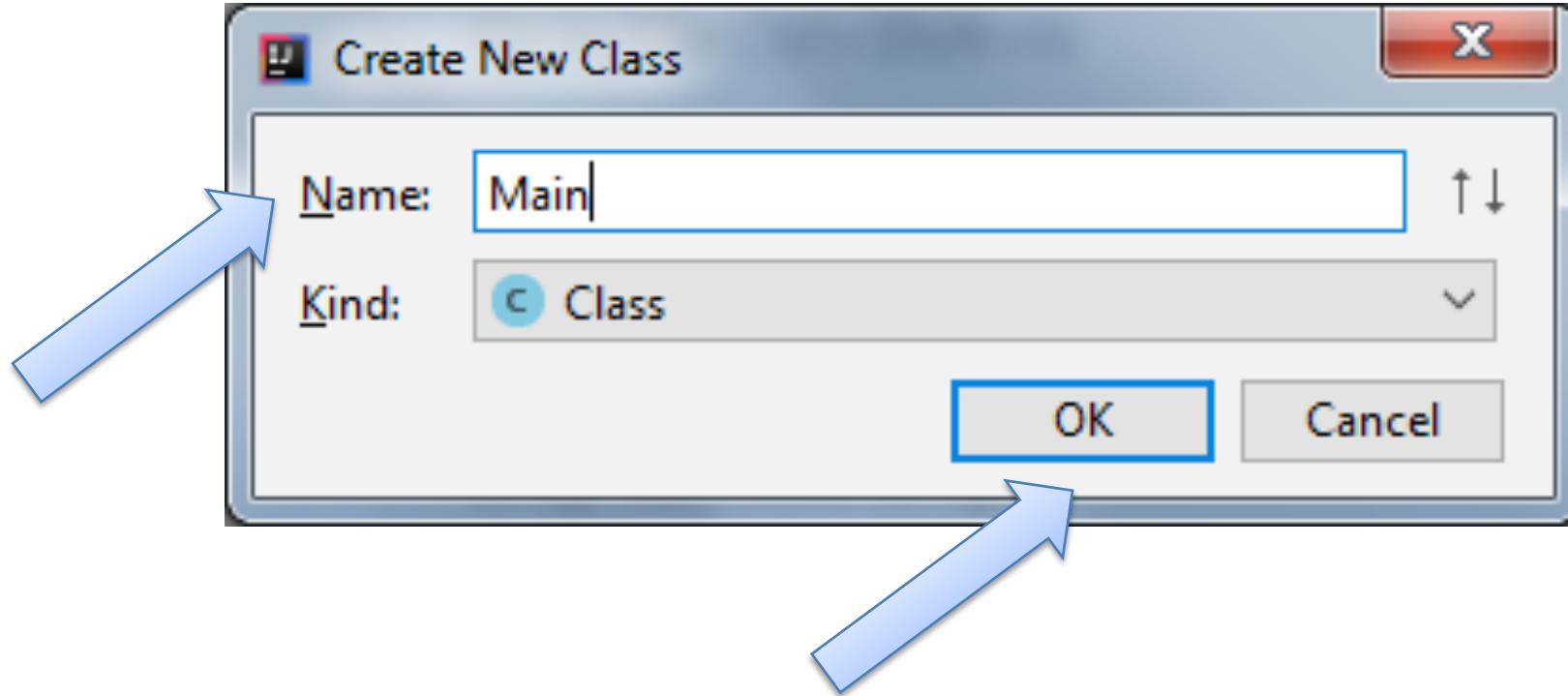
Decide whether you want to see tips each time you start IntelliJ. Then close the tip window.



16

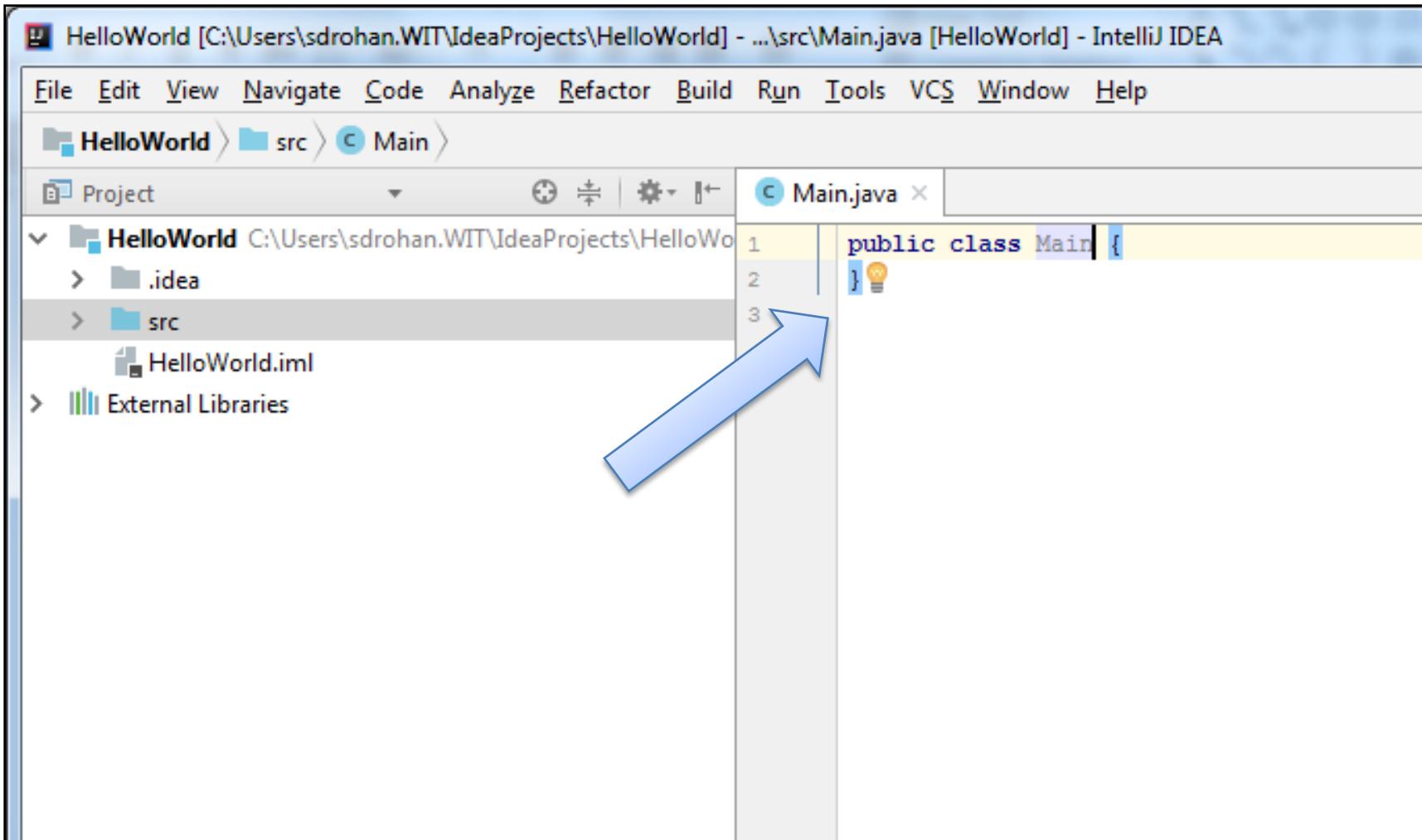
An empty HelloWorld project is created.





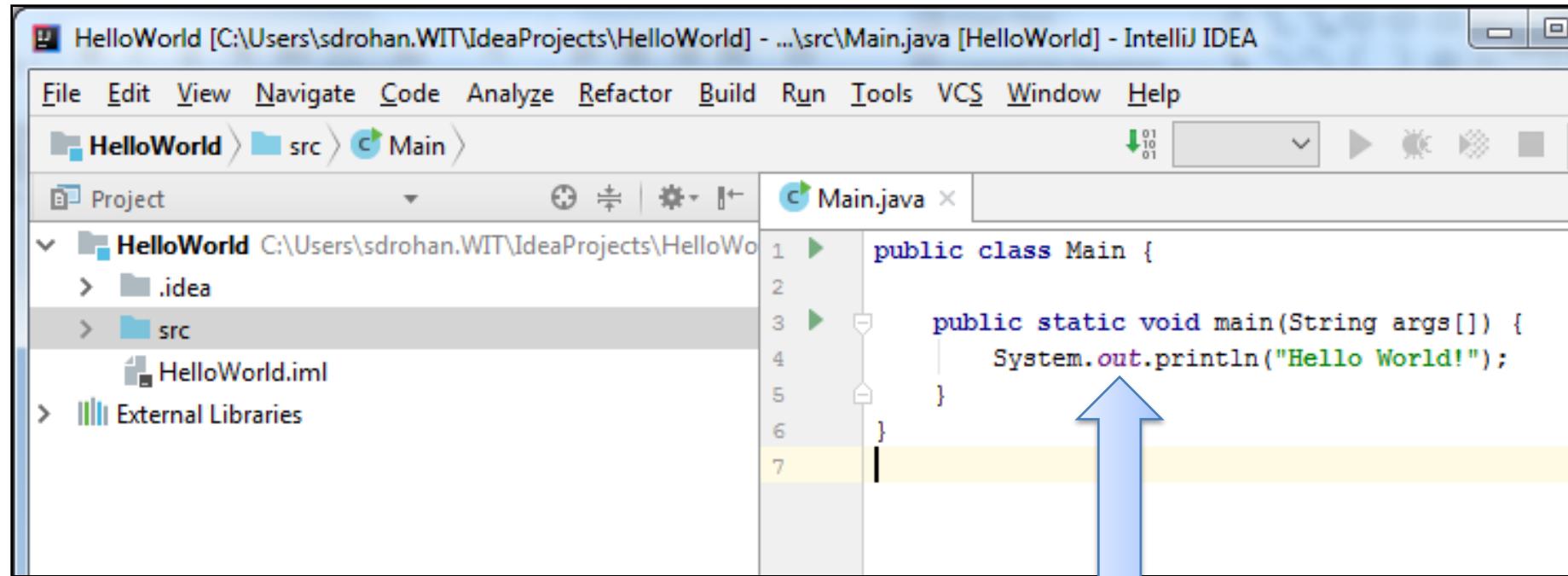
18

Enter the class name **Main** and click **OK**



19

A new **Main** class has been created.



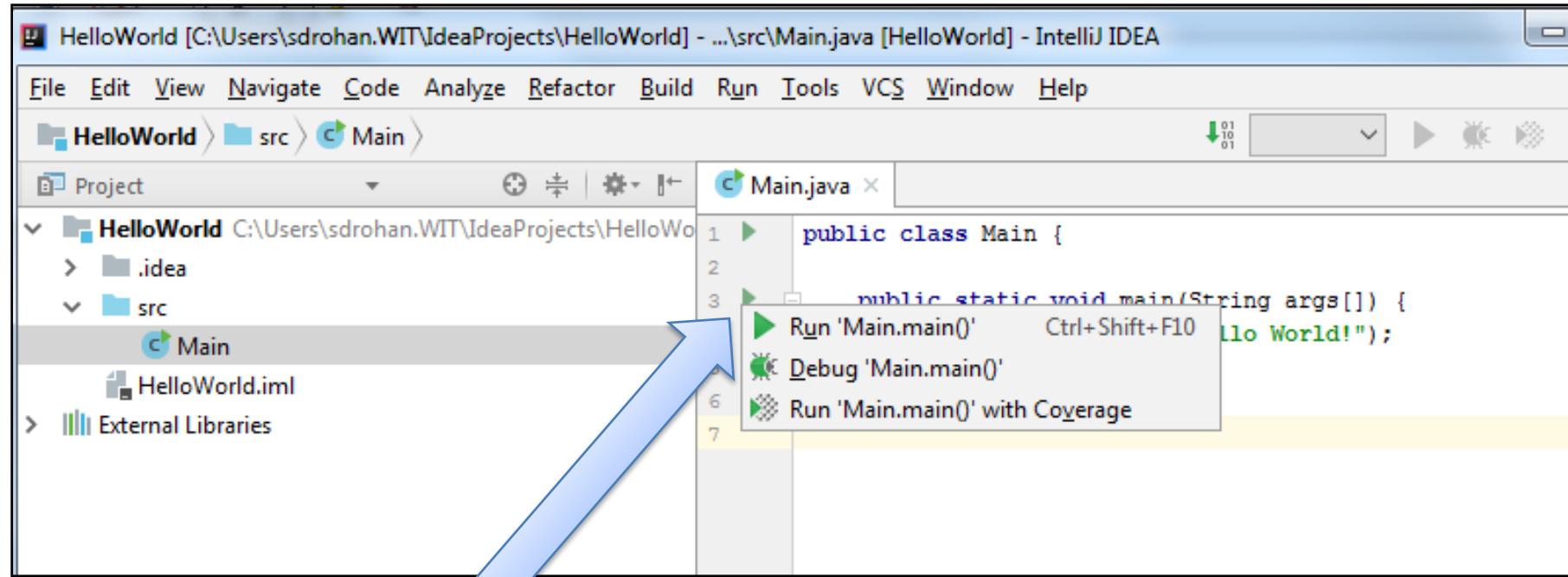
The screenshot shows the IntelliJ IDEA interface with a project named "HelloWorld". The "src" folder is selected in the Project tool window. The "Main.java" file is open in the editor, displaying the following Java code:

```
1 public class Main {  
2     public static void main(String args[]) {  
3         System.out.println("Hello World!");  
4     }  
5 }  
6  
7 }
```

A large blue arrow points upwards from the bottom of the slide towards the code in the editor.

20

Enter this code into the **Main** class.



21

Click on the green triangle beside the main method.
Then select **Run 'Main.main()'**

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

HelloWorld > src > Main

Project

HelloWorld C:\Users\sdrohan.WIT\IdeaProjects\HelloWorld

- .idea
- out
- src
- Main
- HelloWorld.iml

Main.java

```
1 public class Main {  
2     public static void main(String args[]) {  
3         System.out.println("Hello World!");  
4     }  
5 }  
6  
7
```

Run Main

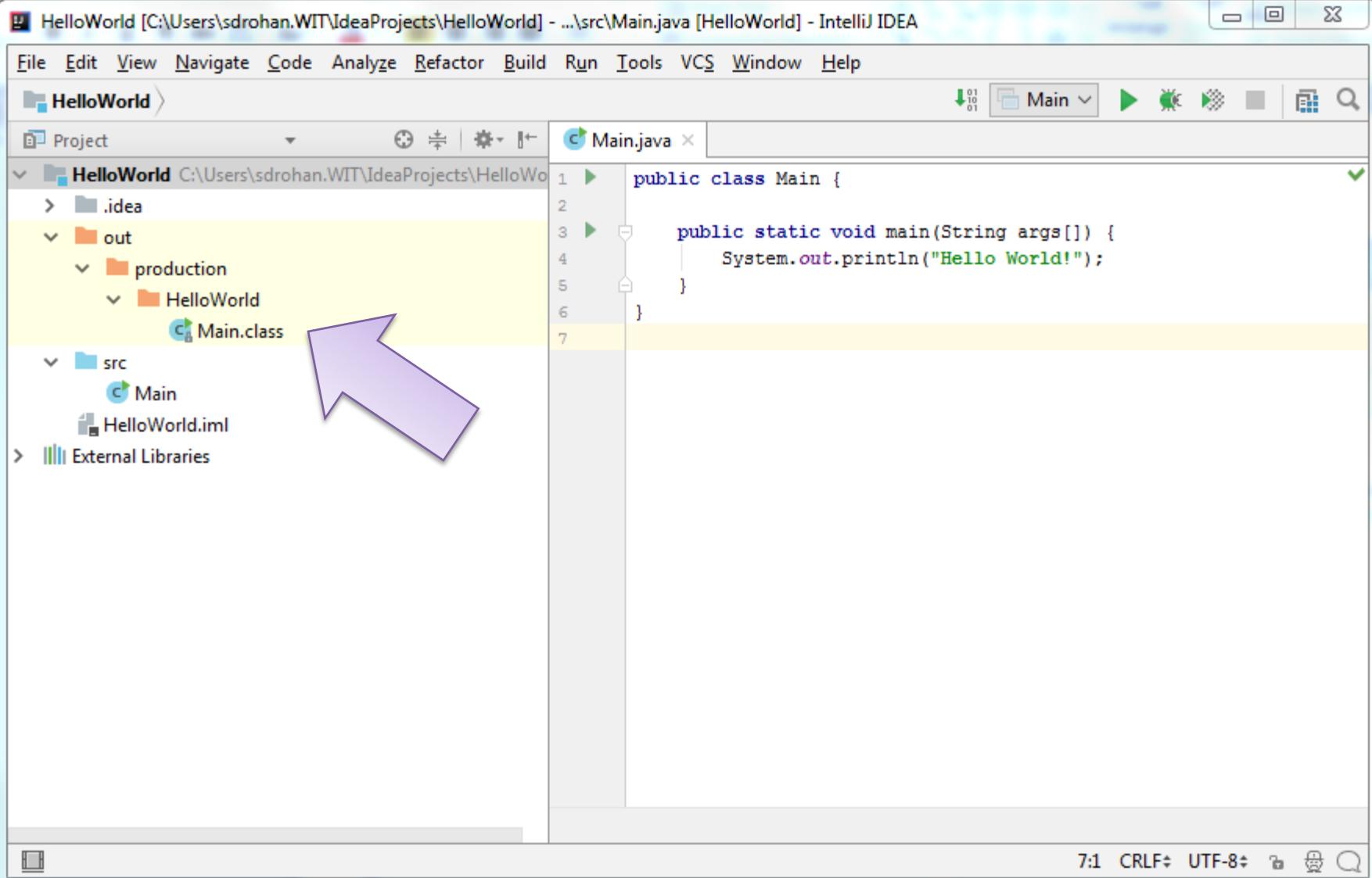
"C:\Program Files\Java\jdk1.8.0_60\bin\java" ...
Hello World!
Process finished with exit code 0

Compilation completed successfully in 6s 305ms (moments ago)

22

Hello World! is printed to console.

*Let's look at where our
Main.class
file lives...*



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

Main.class is the **bytecode** of Main.Java

Questions?

