IntelliJ IDEA Installation Guide

Version 1

Table of Contents

Introduction

These IntelliJ installation instructions are used by students enrolled in the Master of Science in Software Development and other Computer Science Department programs in both on-campus and online programs. The document begins with a discussion of IntelliJ, downloading and installing IntelliJ, creating a Hello World project, and exporting that project for submission. Note that as new versions of IntelliJ are released, some of the screens may look different than the screenshots in this document. Nevertheless, this guide will help get you started quickly on most any version of IntelliJ.

Although the examples in the main document are for the Microsoft Windows family, including Windows 8 and Windows 10, you may use IntelliJ on Linux or on a Mac. You will need to follow a separate guide for downloading and installing IntelliJ on those platforms, and you can skip down to the Creating a Hello World Project section after IntelliJ is installed on your platform.

If you can't determine how to proceed or something goes wrong, and web searches don't help, ask your facilitator or instructor for help. Good luck, and have fun!

IntelliJ Overview

IntelliJ is one of the most popular Java application development environments for desktop and Android applications. IntelliJ runs under Windows, Linux, and Mac OS. It supports advanced Java development from a default install, and can be further extended with a variety of plugins to suit particular environments. IntelliJ supports basic and advanced debugging, as well as a host of application creation tools. By learning to use IntelliJ, you'll have experience with an industry standard Java development environment.

Note that the proper name of this tool is "IntelliJ IDEA", but the tool is often referred to as just "IntelliJ". This document refers to it as such. IntelliJ is created and maintained by JetBrains, an organization that develops various code development software with an eye towards making the tools both powerful and intelligent.

Downloading and Installing IntelliJ

Step 1: Downloading IntelliJ

Visit Website

Go to https://www.jetbrains.com/idea/download/ to get started downloading IntelliJ. JetBrains regularly updates their website, so what you see may be different than the following.

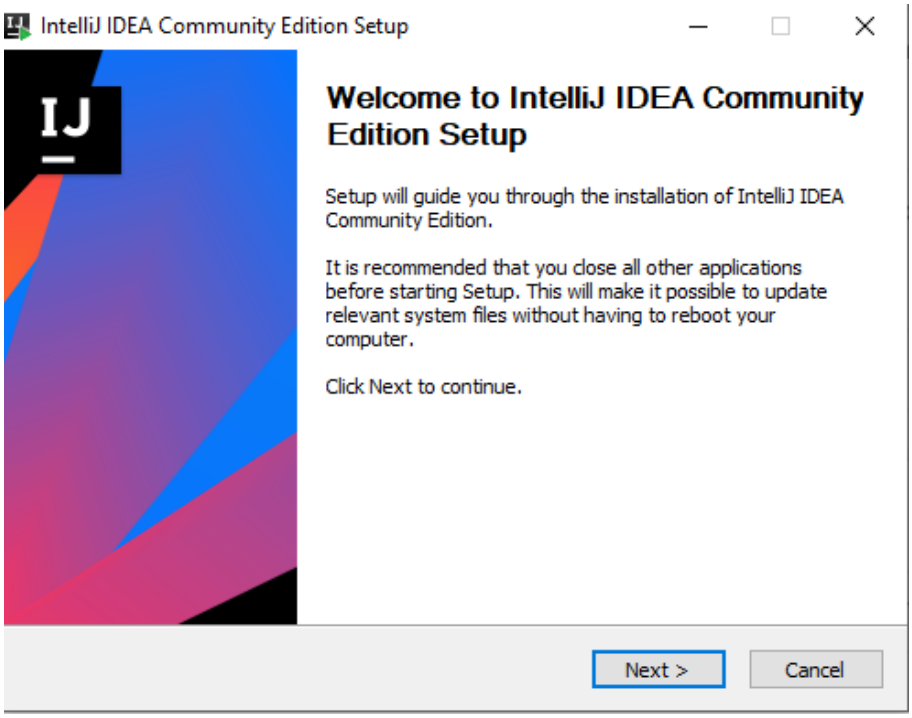
Download and Execute

Click the “Download” link under the “Community” option to get started. The Community edition is free and has more than enough for what you need to do, so there is no need to pay for the other edition.

Step 2: Installing IntelliJ

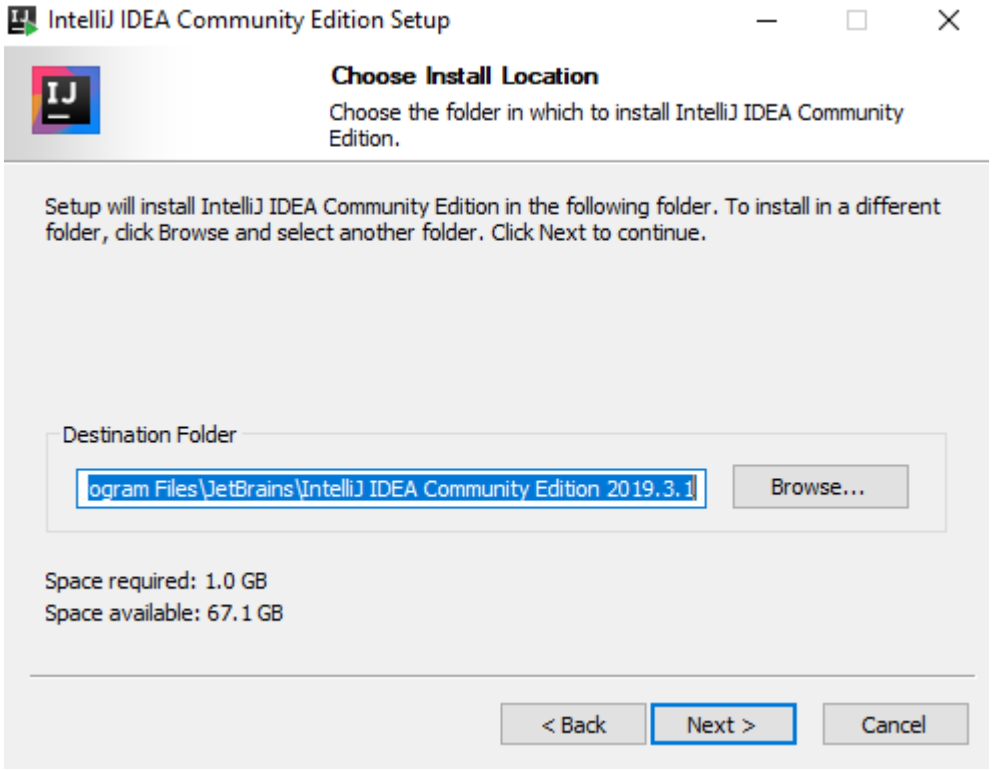
Execute Installer

Your browser will now give you the option to run the executable it downloaded. Go ahead and run it. You'll see a screen like the following.



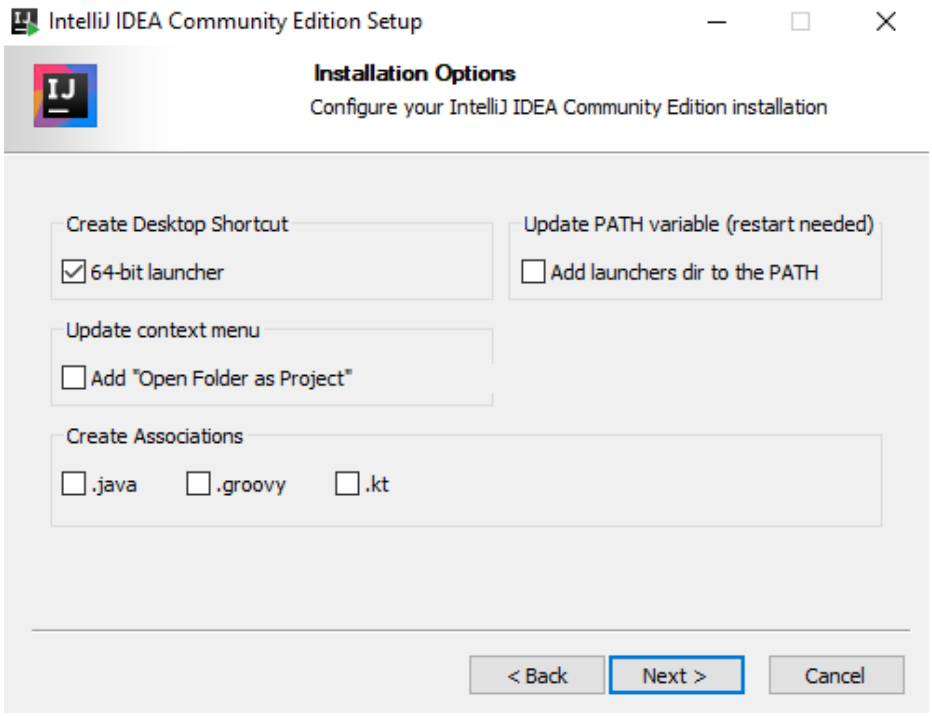
Accept Destination Folder

After clicking the Next button, the installer will show you the directory it pre-selected for installation. There is no need to change the default. The screen will look like the following.



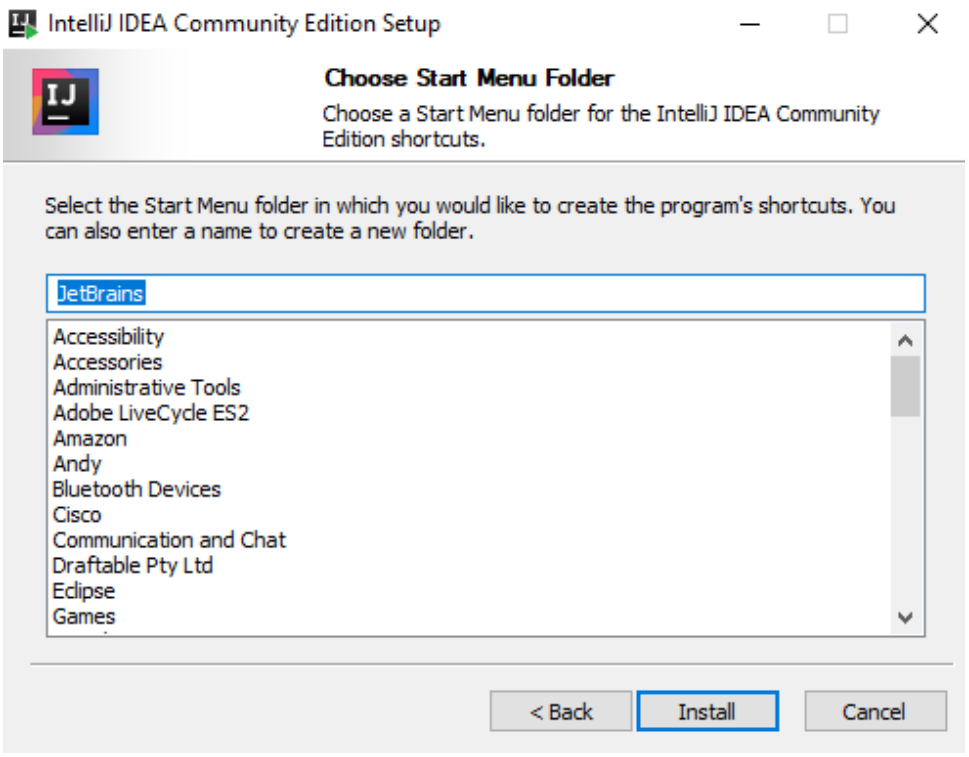
Select Other Options

After clicking the Next button, you'll be presented with other options. The only one you should select is the "64-bit Launcher" option, which will ensure that a shortcut is placed on your desktop for easy access. The screen will look like the below.



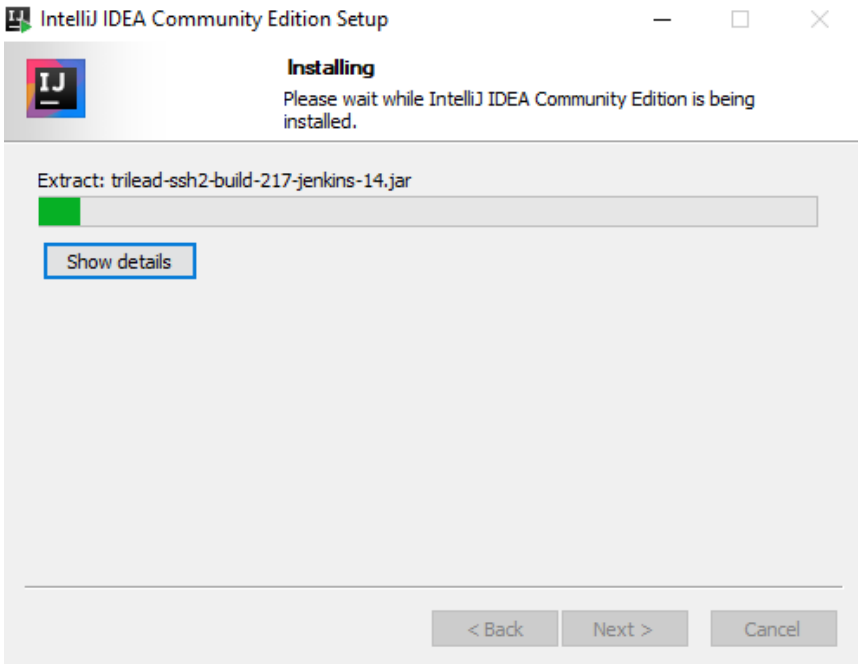
Accept Start Menu Option

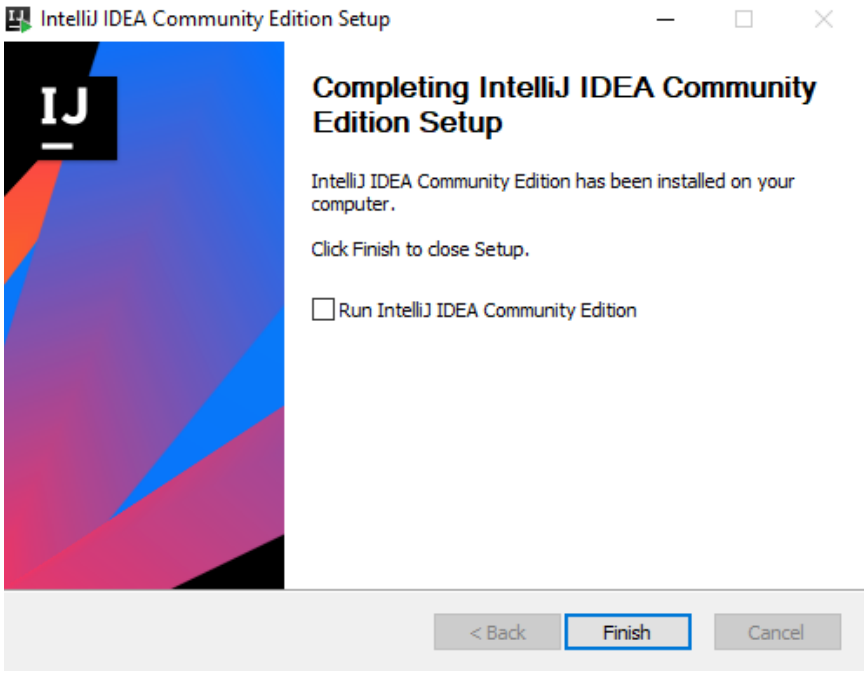
After clicking the Next button, you'll be shown the default Start Menu folder the program will be placed into. There is no need to change it so you can accept the default. The screen will look like the following.



Start the Install

Click the Install button to start the install. You'll see a progress bar for a while, followed by a screen indicating installation is completed, similar to the below.





Congratulations! IntelliJ is now installed on your machine and is ready to use.

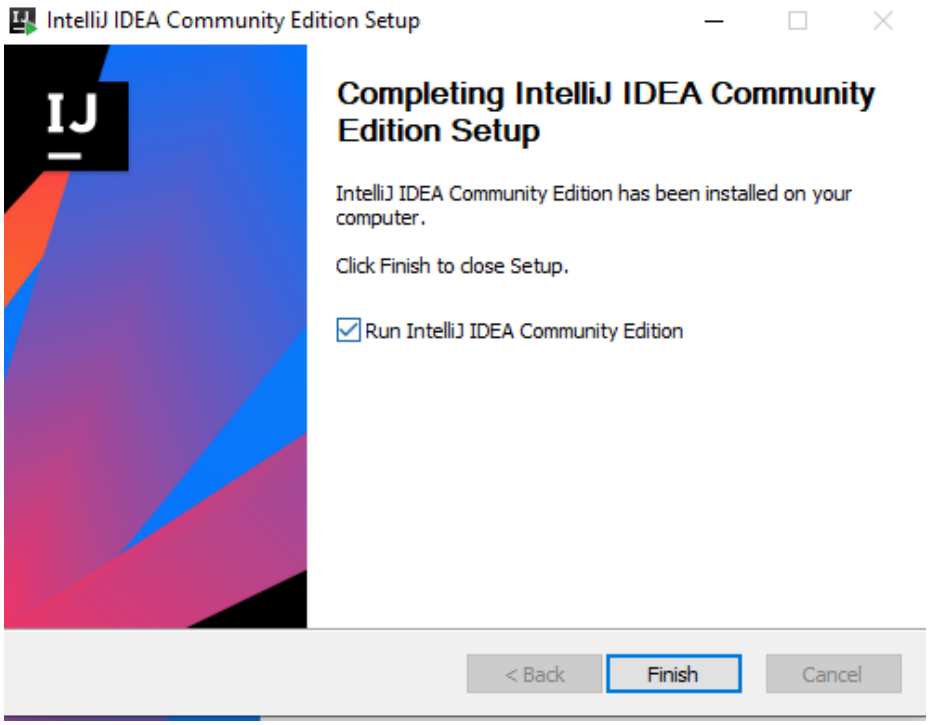
Creating a Hello World Project

To get you started, we'll walk you through creating your very first project. It will simply print "Hello World!" to the screen.

Step 1: Launching IntelliJ

Launching from Installer

After the installation is completed, you can launch IntelliJ for the first time by selecting the "Run IntelliJ IDEA Community Edition" checkbox and clicking the "Finish" button.



Launch from Windows

Going forward, you can launch IntelliJ from Windows, either from the Start Menu, or from the shortcut on the desktop.

Start Menu

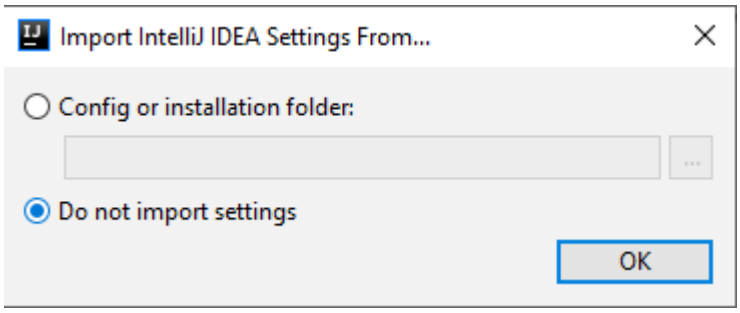
Desktop

Step 2: Customize IntelliJ

Upon first launch, several screens will appear asking you to customize your IntelliJ. We guide you through these step-by-step. You will only see the screens on the first launch.

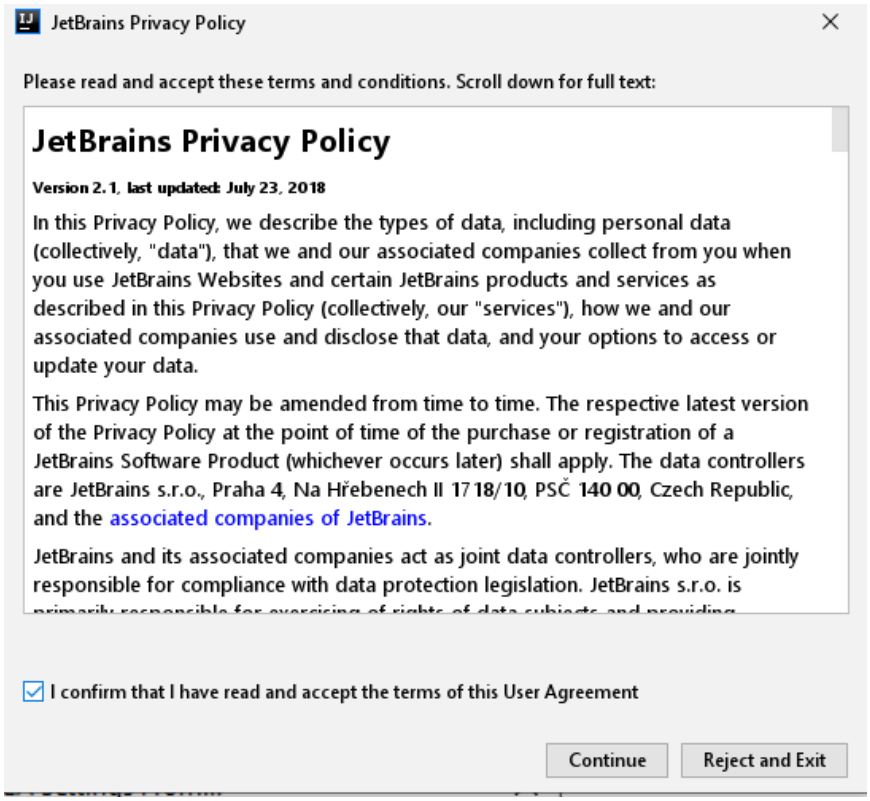
Accept Import Settings

The first screen is about whether or not you want to import anything. You have nothing to import, so just accept the default of not importing anything.



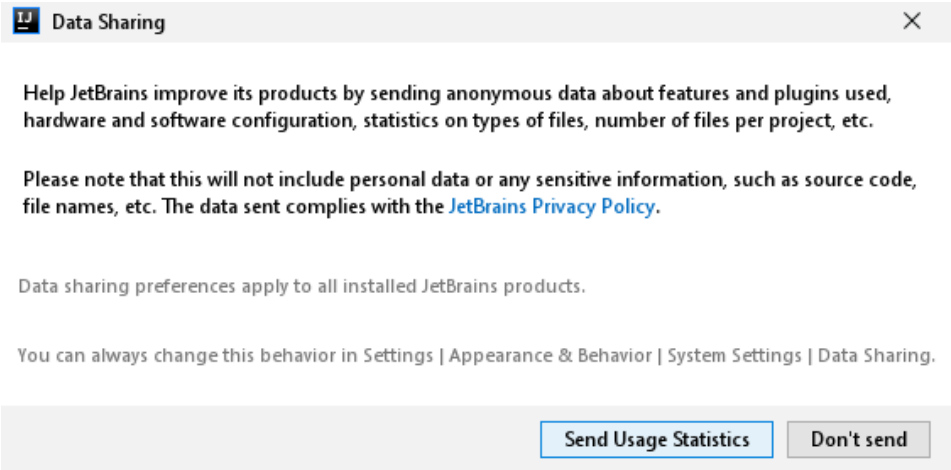
Accept Privacy Policy

Next, you'll be ask to accept the Privacy Policy. Check the "I confirm…" checkbox and hit "Continue" to continue.



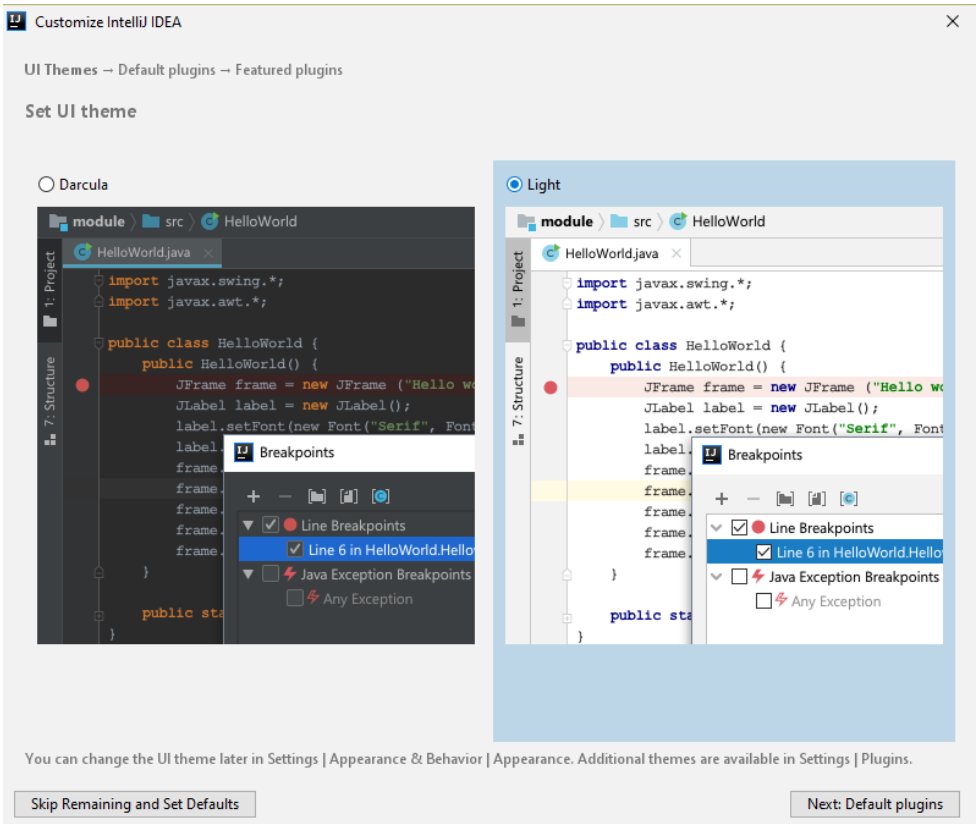
Decide About Data Sharing

Next, you'll be asked to decide whether or not you want to send anonymous usage statistics to JetBrains. This is your choice so you can choose whichever option you feel comfortable with.



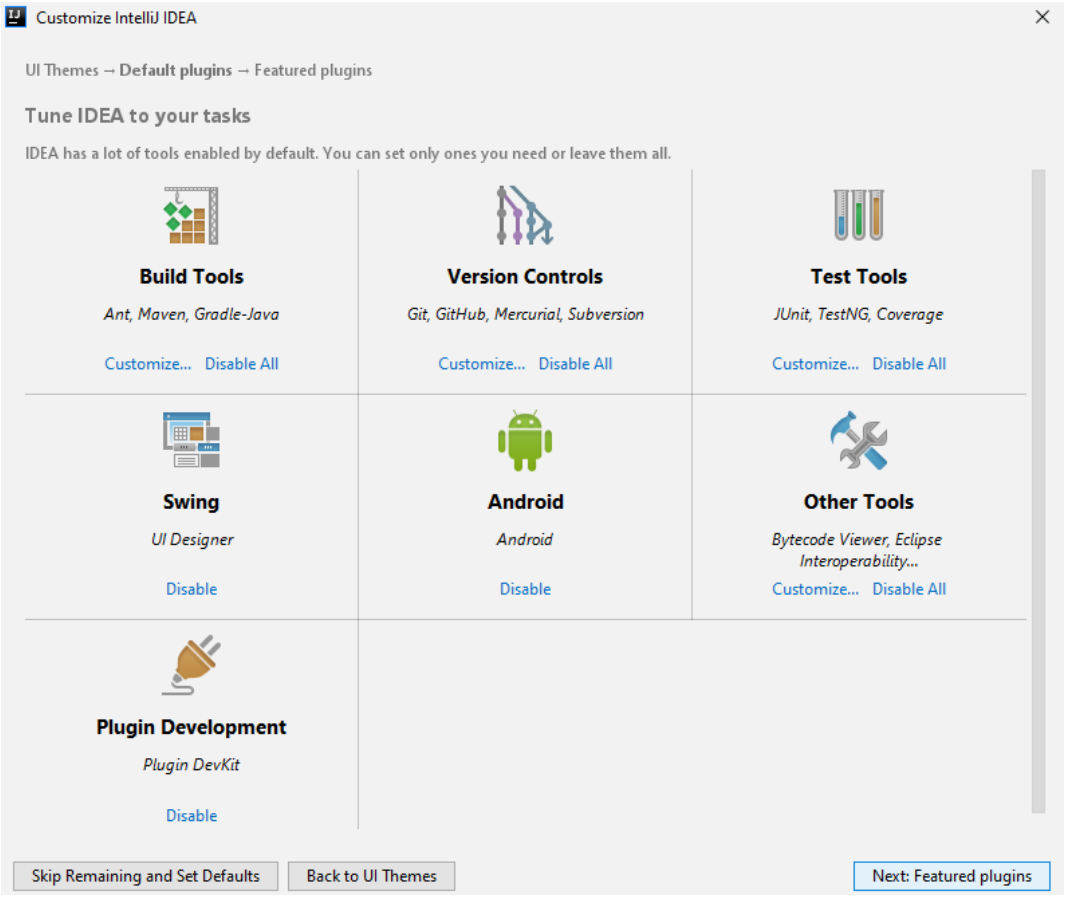
Choose UI Theme

Next, you'll be asked to choose your UI them, light or dark. Light has a white background and dark has a dark background. You can choose whichever you prefer. Below, we select the "Light" option.



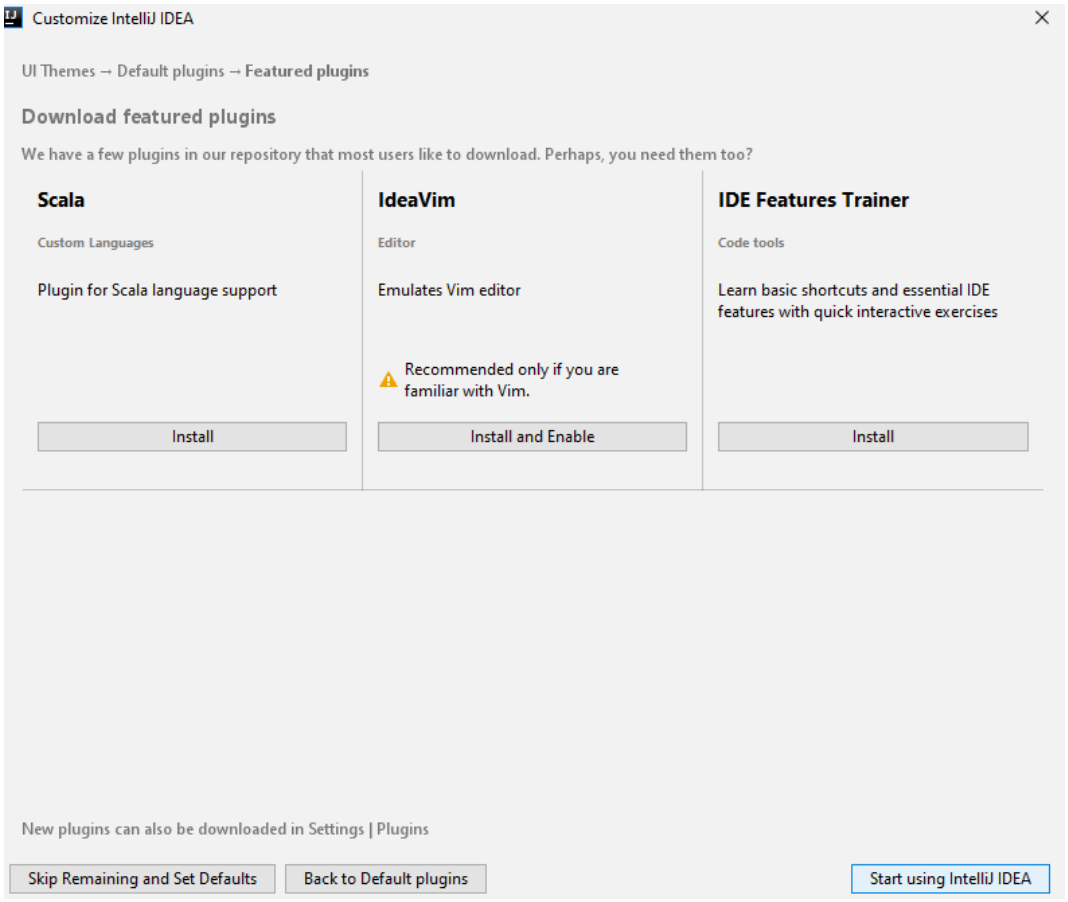
Accept Default Tools

After clicking the “Next: Default plugins” button, you’ll be presented with several other options. You don’t need to change anything. Just accept the defaults.



Accept Default Plugins

After clicking the “Next: Feature plugins” button, you’ll be asked to choose additional plugins. You don’t need them, so you can just accept the defaults.



Step 3: Creating the Project

Splash Screen

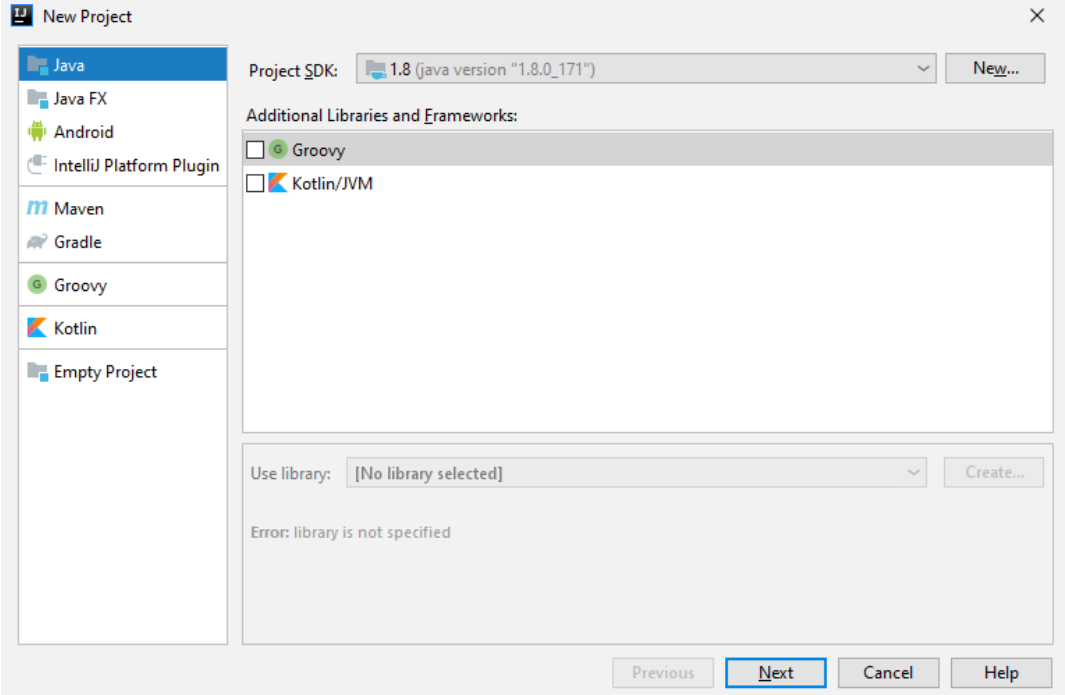
After clicking the "Start using IntelliJ IDEA" button, you'll first see a splash screen with a progress bar.

Create a Project

After this, the initial IntelliJ screen appears with a few options; only one is needed for us right now. Click the "Create New Project" option to get started creating your project.

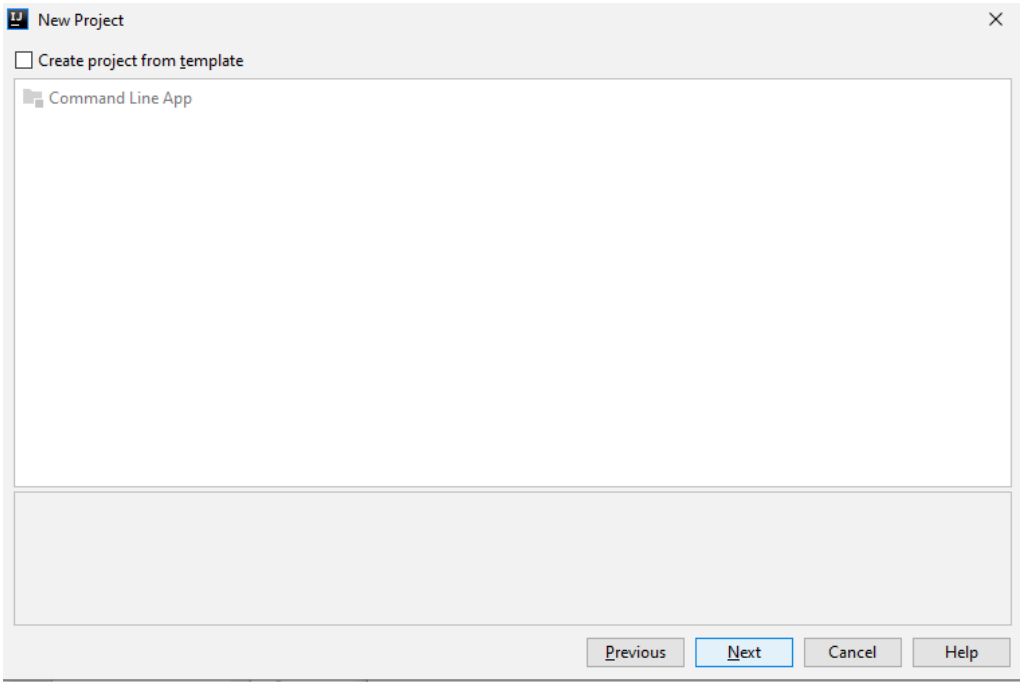
Choose Project Type

Your next step is to choose the “Java” project type. There are other types of projects you can create, but a typical project is a Java project which allows you type Java code and execute it. The Java project type is selected by default, so just click the Next button to continue.



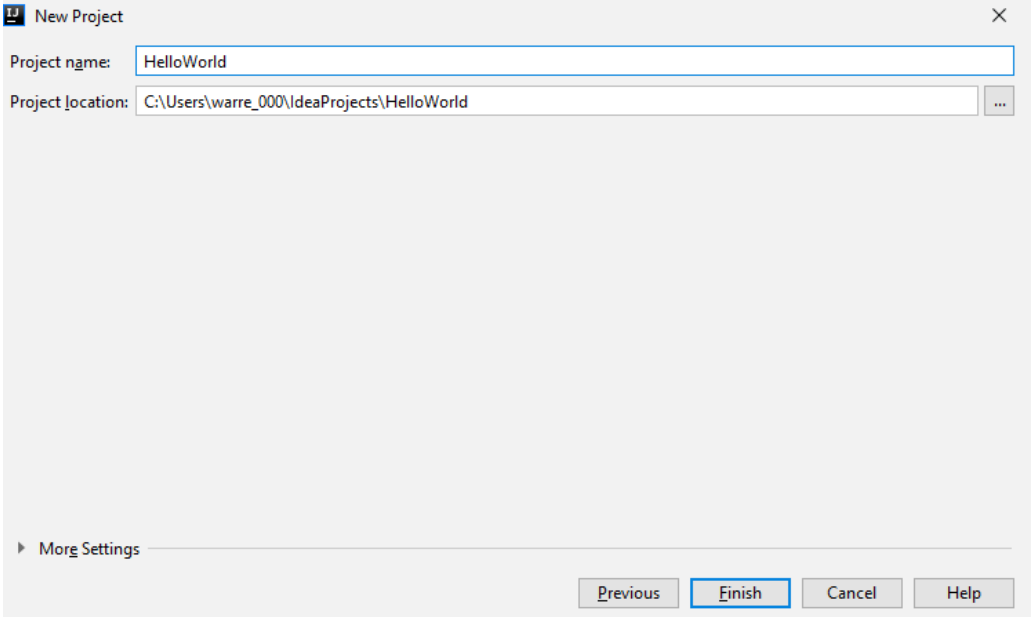
Accept Template Defaults

Next, the screen prompts you to decide whether you're creating your project from a template or not. We are creating a project from scratch (which is typical), so leave the "Create from project template" checkbox unchecked and click the Next button.



Name the Project

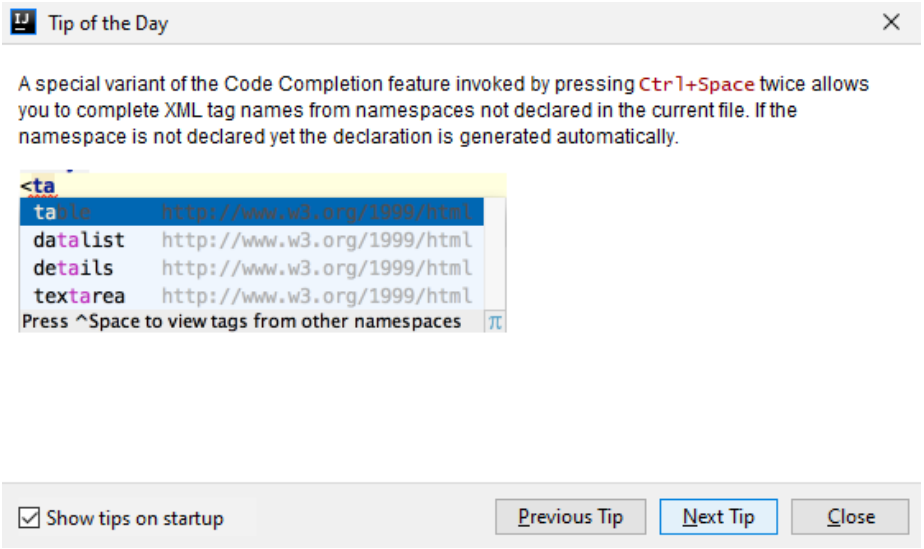
Now you need to choose the project name. Enter “HelloWorld” with the spacing and case exactly as shown. Following Java convention, you should use camel case for the name. With camel case, spaces are not included in the name, and each word is capitalized. So, “Hello World” in plain English is “HelloWorld” in camel case.



You can leave the default project location as is, because it is using your user directory. Of course, if you’d like to put your project somewhere else on your drive, you are free to do that here.

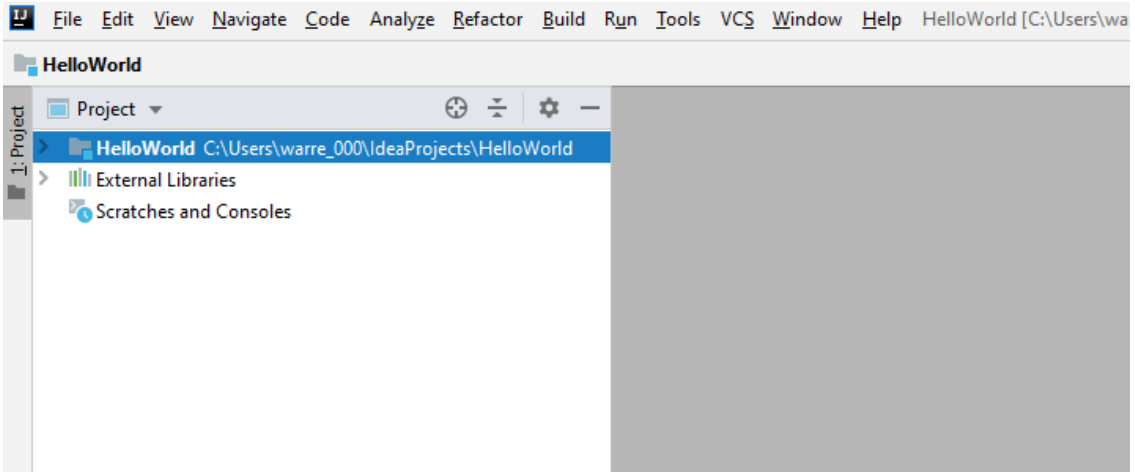
Tip of the Day

You'll next see a tip of the day which has useful information you can take advantage of. For now, dismiss the window.



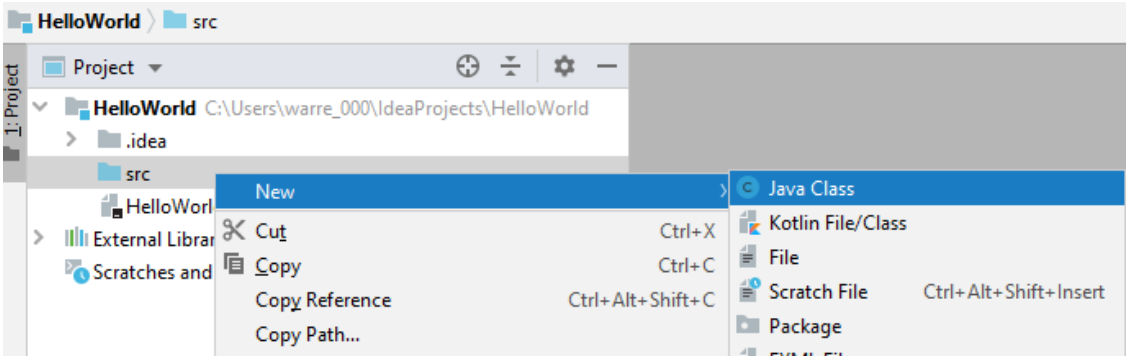
Understand the Workspace

You will then see the workspace where you create and run files in your project. The left-hand pane is the project pane where you can see the files in your project, and can use context menus to create and modify the files. The right-hand pane is the buffer where the file you are currently editing will display. The menu bar appears at the top where you can access many features, which you'll learn progressively as you become more familiar with IntelliJ.



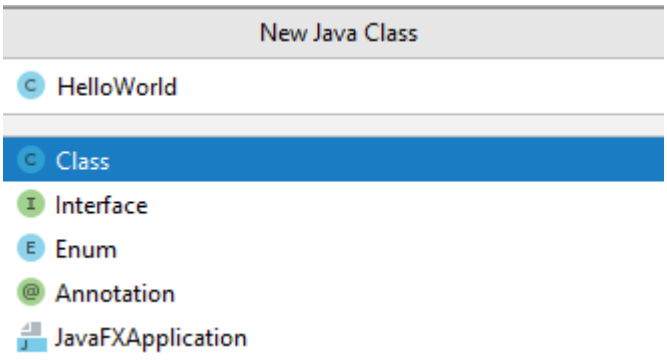
Create a Java Class

You have created a project at this point, but it has no source file in it yet, so we need to create one. Expand the carrot next to the HelloWorld project in the right-hand pane, and you'll see your "src" directory. Right click on the "src" directory and choose New/Java Class, as shown below.



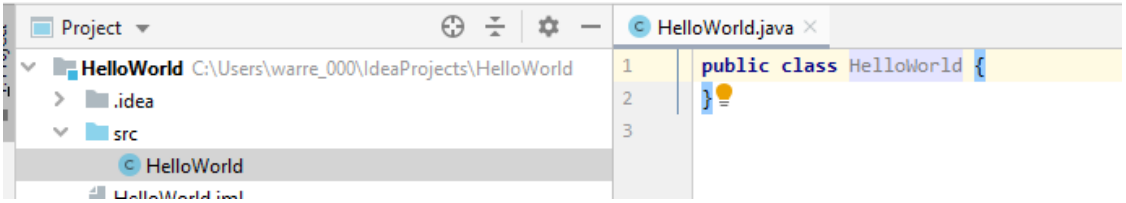
Choose the Class Type

The next screen asks you to choose the type of class – conventional class, interface, enum, and so on. You want the conventional class, so just click the "class" option.

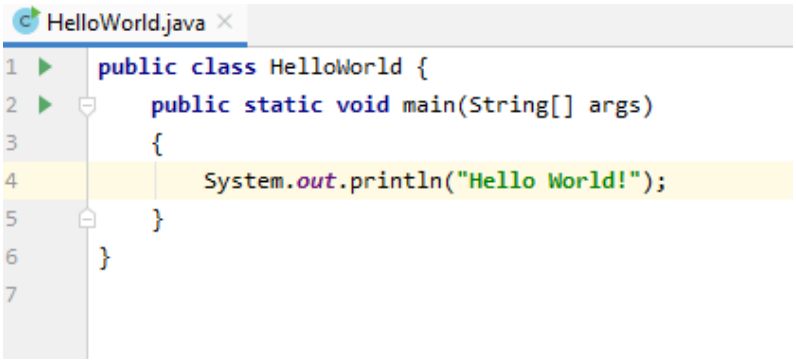


Edit the Code

The HelloWorld.java class is now created. Initially, the class has no implementation code, as follows.



However, you can now easily add the code you need in the code buffer on the right. Type the code as shown below.

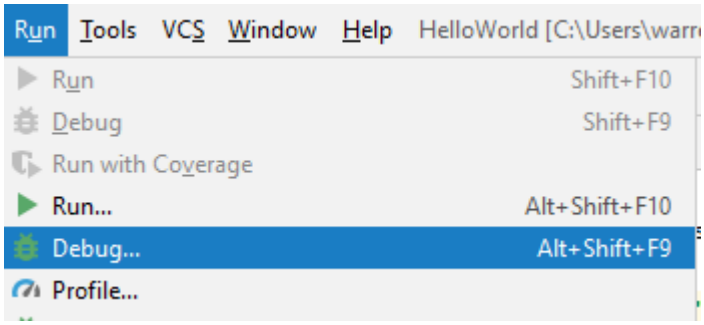


This code adds the static main method, which is the method invoked when a class is called, and adds the logic to that method that prints out “Hello World!” by using System.out.println.

Step 4: Running the Project

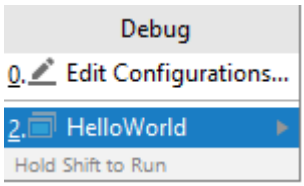
Debug the Project

Once you have your class created, you can choose to "Run" it or "Debug" it. When you are developing a class, it is recommended to debug it, because you can use breakpoints and other debugging features, and IntelliJ will help you figure out where exceptions are occurring rather than just printing them in the output window. Click Run/Debug in the menu to start.



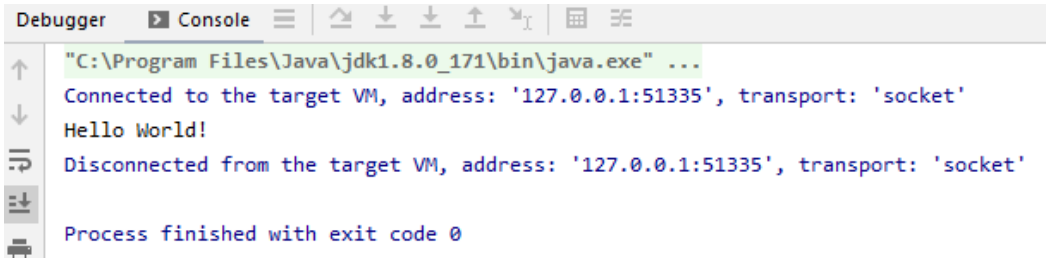
Choose HelloWorld

Next, IntelliJ asks you to decide what you are debugging. In this case, you are debugging the HelloWorld class, so choose that from the list.



View Output

Your HelloWorld class will execute, and you see this in the output at the bottom.



Congratulations! You have executed your first class and have seen its output.

Exporting a Project for Submission

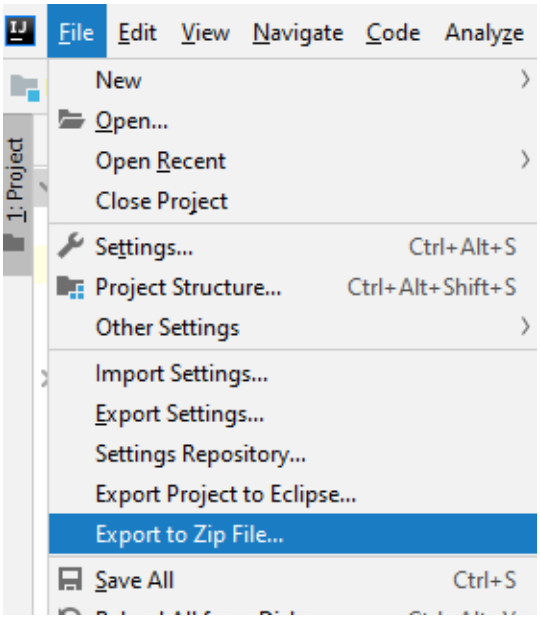
Step 1: Exporting the Project

Understand Exporting

In order to submit your project to your facilitator, you will need to export it, which means you gather its contents into a single location, useful for giving it to others. While there are many formats you can use for export, for classes at BU, you should export into a zip file, because it is a universal file format that virtually anyone can use.

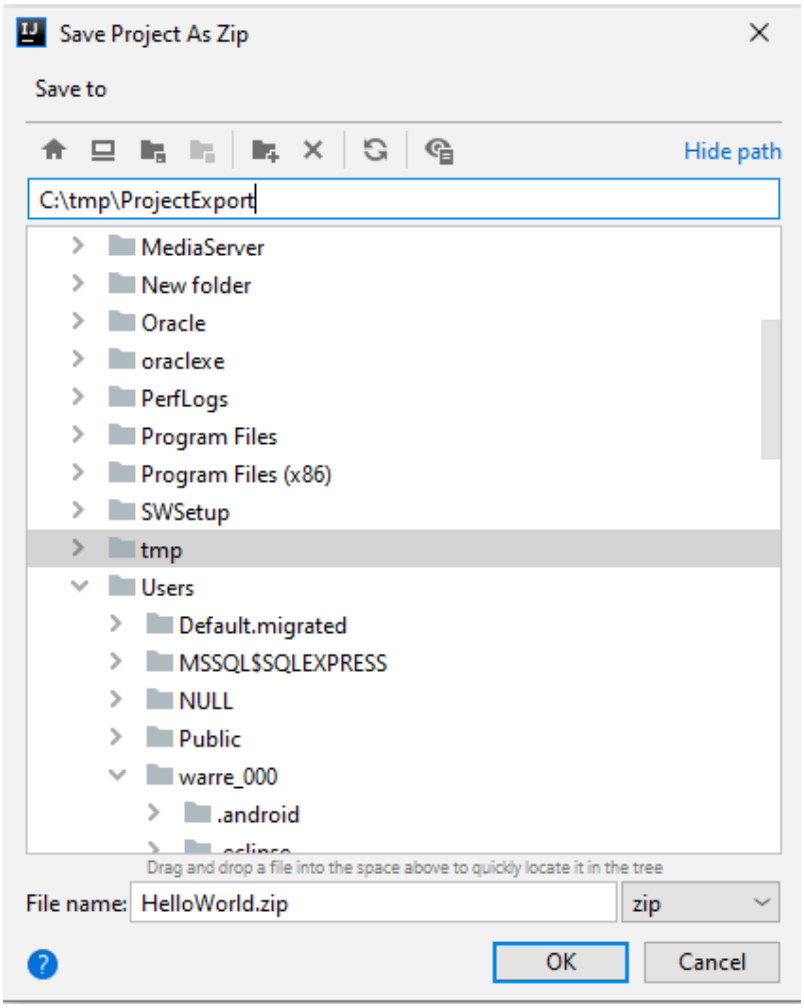
Start the Export

To start the export, click the File/Export to Zip File option.



Choose Export Options

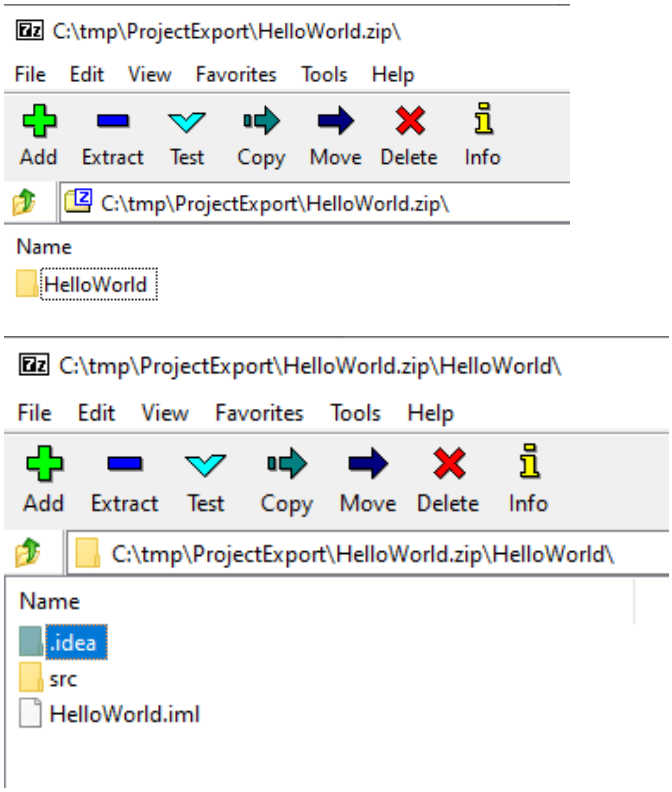
On the next screen, you decide on the directory in which the export will be created, and the name of the zip file. Choose a directory you can remember and easily access. In this example we are using “C:\tmp\ProjectExport” for the directory. For the filename, here we are using “HelloWorld.zip”; however, when submitting an assignment, it’s a good idea to include your name and the name of the assignment, so your facilitator or instructor can differentiate your submission from other submissions.



Optional Step 2: Viewing the File

View the File

After clicking the “Finish” button to create your export archive, you can optionally view the contents of your zip file. Once you are comfortable with this process, you can skip this step. In our example, the contents of the HelloWorld export look like this.



The .iml file is a configuration file for IntelliJ, which specifies the module paths, dependencies, and other settings. The .idea folder contains IntelliJ’s project specific settings files. The src directory contains the source Java files you created.

Congratulations! You can now submit your export as part of your assignment in your class.

Next Steps

You have now downloaded and installed IntelliJ, created and executed a simple HelloWorld project, and exported it for submission. This is a great first step! As you start your class, your instructor will ask you to create additional projects. Don't worry. If you can create one project, you can create many.

In the HelloWorld example shown in previous sections, we intentionally kept things very simple. As you work on larger projects, you may need to use other features of IntelliJ. You may also need to add additional libraries. Because IntelliJ is so popular, there is a plethora of help documentation, and even step-by-step tutorials, on the web for almost anything you want to accomplish. There are also good books on the subject if you want to dive deep. And of course, your facilitator and instructor are available to help you should you get stuck. Take advantage of all of this help to succeed!