

Guide to Using Replit

by Sophia



WHAT'S COVERED

In this lesson, you will review the Java integrated development environment that will be used in this class. Specifically, this lesson covers:

Table of Contents

- [1. Introduction to Replit](#)
- [2. Creating a Replit User Account](#)
- [3. The Replit IDE](#)
 - [3a. Left Pane: Menu Bar and Files and Configuration Pane](#)
 - [3b. Middle Pane: Code Editor](#)
 - [3c. Right Pane: Output Sandbox](#)
 - [3d. Above the Panes: Run Button \(Do Not Use With Java\)](#)
- [4. Running a Program](#)

1. Introduction to Replit

There are different ways to write your code and run it when creating the program. For many languages, there is the option to use an **Integrated Development Environment (IDE)** to write the code. An IDE can be viewed as a text editor that has additional functionality. This allows developers to perform additional tasks that simplify the workflow of the development process. Individuals and organizations may have different preferences of which IDE they like to use.

This course uses Replit as the IDE for Java. Replit has some unique features. Replit is a tool that is completely run through a browser. This means it can be used from any computer, tablet, or mobile device to write, build, and run code.

Replit can be used to run code that has been created in the online console. There is no need to copy code or make changes to the underlying environment. Additionally, Replit can fully manage the environment that is used to build and run code. This means that the right version of Java is not important. It is also not necessary to confirm the correct **libraries**, or prewritten code.



HINT

While a Replit account is not required, having one allows for the use of all of the features. This account allows access to features that will be required for completion of work in this course. }}



TERM TO KNOW

Integrated Development Environment (IDE)

A text editor that has additional functionality to allow developers to perform some additional tasks to simplify the workflow of the development process.

2. Creating a Replit User Account



STEP BY STEP

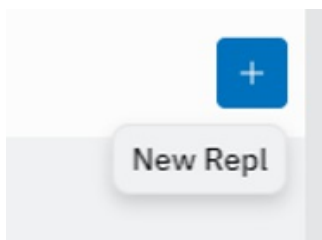
1. To get started, access the following site:

[Replit/signup](https://replit.com/signup)

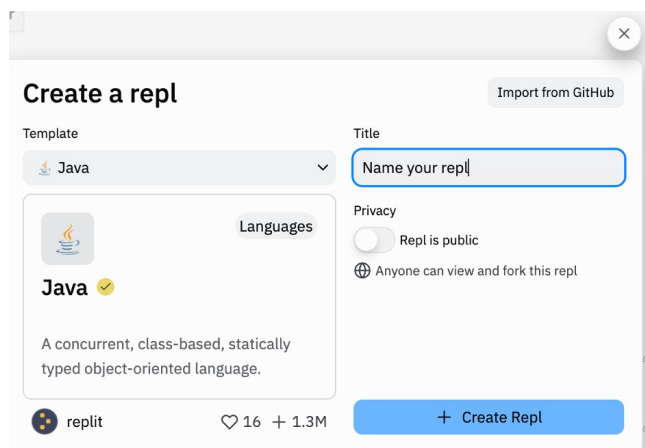
2. Follow the prompts to create a Replit user account:

The screenshot shows the 'Create a Replit account' page. At the top, it says 'Teacher? Sign up for our education product.' Below this are three input fields: 'Username', 'Email', and 'Password'. A blue 'Create account' button is positioned below the password field. Underneath the button, it asks 'Have an account? Log in'. Further down, there are three social login options: 'Continue with Google' (with the Google logo), 'Continue with Github' (with the Github logo), and 'Continue with Facebook' (with the Facebook logo). At the bottom, a small text block states: 'By continuing, you agree to Replit's Terms of Service and Privacy Policy, and to receiving emails with updates.'

3. Once the account is created, click on the “+” New Repl button in the top right. This will allow the creation of a new Java project.

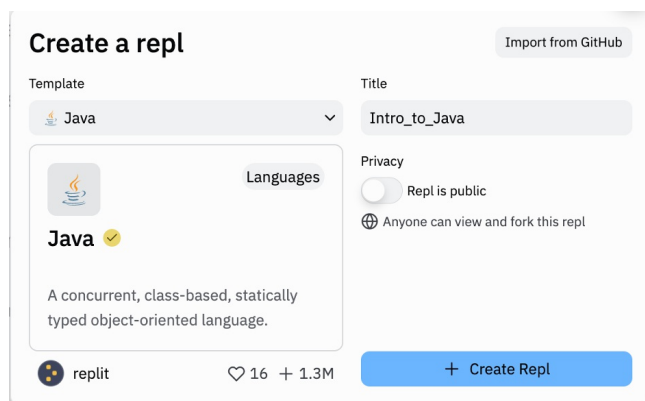


4. The Create a repl window appears. Select “Java” from the dropdown languages. If Replit has been used previously for Java coding, Java may appear at the top under Favorites.



5. By default, a random project name will be generated in the Title field. Please change that to an alternative project name.

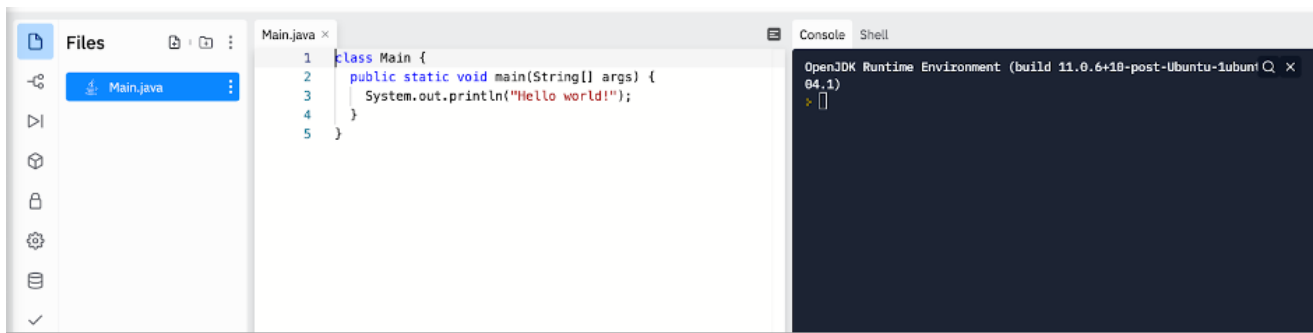
It's important to note that by default, the repl will be public. This means that this project can be accessible by anyone on the Internet. Although this can be very useful when it comes to collaboration and sharing of code, care should be given when using personally identifiable information. This includes the use of passwords in any of the projects.



6. Finally, select “+ Create Repl” to create the project.

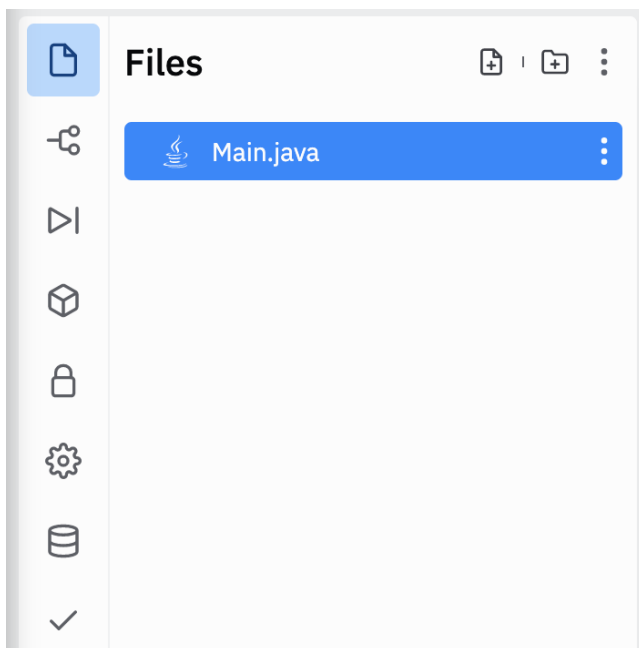
3. The Replit IDE

Once the project is created, the Replit IDE layout will appear. The IDE consists of three (3) main panes and a Run button. The main elements can be reviewed in a left to right order.








3a. Left Pane: Menu Bar and Files and Configuration Pane

The left pane actually consists of the menu bar and the files and configuration pane. Each menu bar item will change what can be seen in the files and configuration pane.



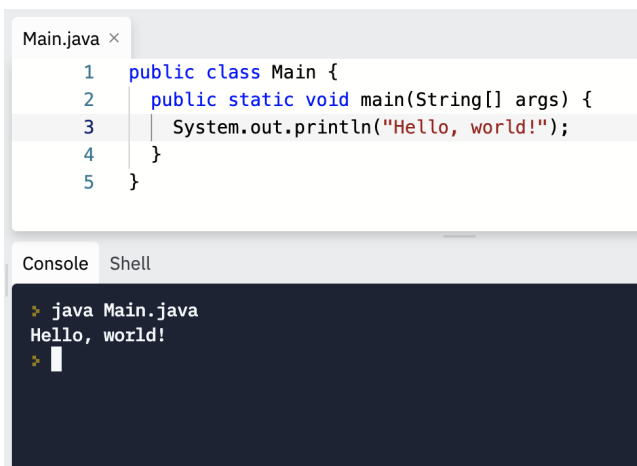
Although the menu bar has quite a few features, it is not critical to know what they all do. In most cases when using an IDE, only some features may be used within the IDE. A description of each of the menu bar features follows in the table below:

Icon	Feature	Description
	Files	This will display all of the files in the project and allow you to add new files or folders into the project. This item is the default menu bar item.
	Version Control	This isn't one that is needed right now, but if there is a need to track changes and roll back to prior versions (similar to Track Changes in MS Word), using this feature will allow it to be done.
	Debugger	This feature can be very helpful to create breakpoints in the code. It also allows users to step through the code one line at a time—but more on that in a future tutorial!
	Packages	These packages allow the import of code and functionality that hasn't been built but could be added to the program. For example, when building an image gallery and there is a need to create an image carousel that rotates, a package can be imported that already has that feature which will make it much easier to build and implement. Note that Replit packages are not the same thing as Java packages.

	Secrets (Environmental Variables)	Even if Replit is set up to be accessible by the public, variables can still be stored in a manner that's hidden away from everyone else but that allows you to make use of them in the Replit.
	Settings	This feature can be very useful as it allows customization of the layout, color scheme, font size, and other features.
	Database	This feature allows quick database key commands.
	Unit Tests	This feature allows you to run tests.
 HINT		

Throughout the course, you will be using the Setting default options. However, if you would like to change those, you can do so.

Below is a setting that is particularly useful when coding on a mobile device. It allows for changing the side-by-side option in the layout to be stacked. This changes the layout so that the code editor and output sandbox are on top of each other instead of side-by-side (see image below).



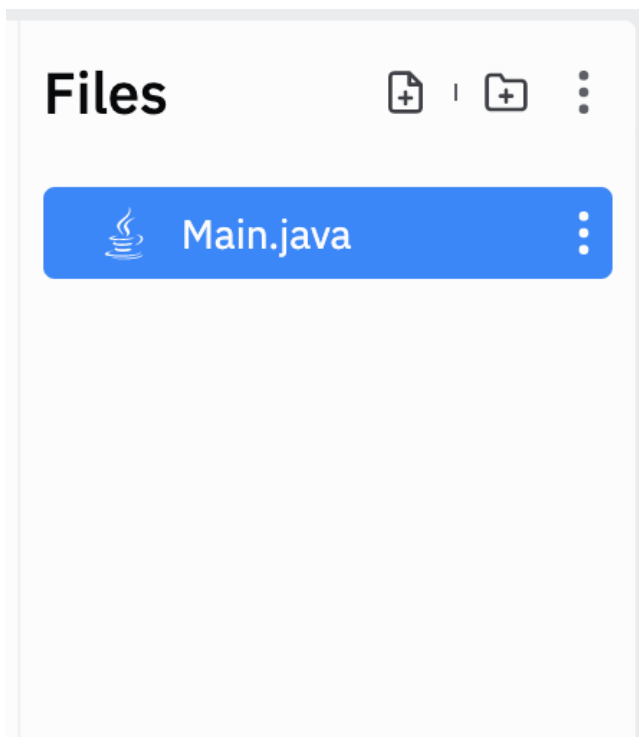
```

Main.java x
1 public class Main {
2     public static void main(String[] args) {
3         System.out.println("Hello, world!");
4     }
5 }

Console Shell
❖ java Main.java
Hello, world!
❖

```

To the right of the menu bar is the files and configuration pane. By default, this pane displays the Files menu bar information, which displays the files that make up the project. Depending on what menu bar feature is selected, this pane will update accordingly.



3b. Middle Pane: Code Editor

The middle pane is the code editor. This is the pane where you will spend the greatest amount of time. Java code is written and edited in this pane. These are unique features found within the code editor. In the configuration pane, the files have a main.java file listed. By default, when a Java project is created, Replit has created that main.java file. The code that is entered in the code editor is saved to that main.java file.



3c. Right Pane: Output Sandbox

The last pane on the right is the output sandbox. This is the section where code is run and the output is presented.



3d. Above the Panes: Run Button (Do Not Use With Java)

Above the main panes, there are additional functionalities. However, one of the more important functions is the run button. It is located in the middle of the pane at the top. Using the Run button often exposes issues

with the sandbox configuration.



When running Java applications, users should not use the Run button. The Java command should be used inside of the console window.



Libraries/Library

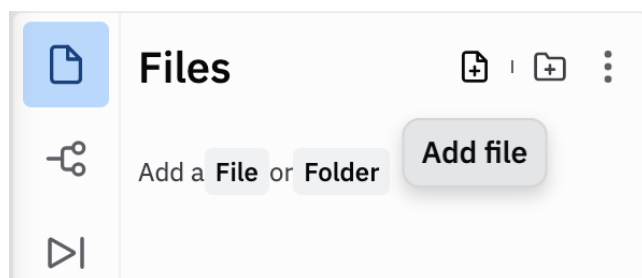
Prewritten code collections that you can use when developing a program.

4. Running a Program

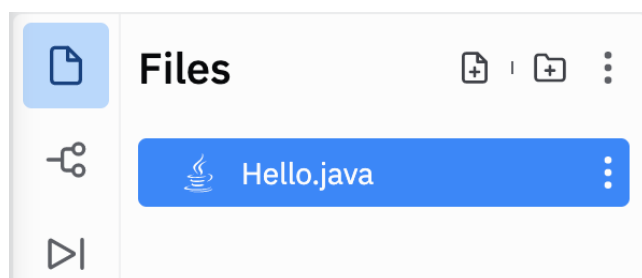
Now that you are set up with an account and have had a tour of the Replit application, let's set up your first program. To get started, enter some code and test it and see what happens.



Directions: In the Replit Files panel, click the Add File icon.



Name the file Hello.java (capitalization and spelling are important).



In the code editor, type in the following code:

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

The following is a screenshot of the Replit IDE code editor screen with “Hello World”.

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

REFLECT

Notice that on the left of the lines of code there are line numbers being displayed. These are not part of the code. They indicate the line numbers. This will be useful in the future.

Let's try another simple program.

TRY IT

Directions: In the console on the right side of the window, type the following command to run the program:

```
java Hello.java
```

REFLECT

What happened? The program was compiled and executed in Replit. In doing so, since our code didn't have any errors, it displayed the result of the print command on screen. Congratulations! You have written your first Java program, built it, and run it. That is how easy it is to run it!

HINT

Be aware that unintentional line-wrapping may occur in the code samples throughout the course depending on your display. A screenshot of Replit is typically provided following code samples for clarity.

SUMMARY

In this lesson, you set up a **Replit account**. You were introduced to the **Replit IDE** (Integrated Development Environment), including the **left pane**, which consists of the **menu bar** and the **files and configuration pane**; the **middle pane**, which is the **code editor**; the **right pane**, which is the **output sandbox**; and the additional functionalities **above the panes**, such as the **run button** (which should not be used when running Java applications). You explored how Replit will provide a working platform to try, write, test, and debug code while working in this course. At the end of this tutorial, you wrote and **ran your very first Java program!**

Source: This content and supplemental material has been adapted from Java, Java, Java: Object-Oriented Problem Solving. Source cs.trincoll.edu/~ram/jjj/jjj-os-20170625.pdf

It has also been adapted from "Python for Everybody" By Dr. Charles R. Severance. Source py4e.com/html3/

**Integrated Development Environment (IDE)**

A text editor that has additional functionality to allow developers to perform some additional tasks to simplify the workflow of the development process

Libraries/Library

Prewritten code collections that you can use when developing a program