

# GridWorld Installation Guide

## 1. Unpack the ZIP file.

Download and unpack the `GridWorldCode.zip` file. A folder named `GridWorldCode` will be created. Move it into a convenient location, such as `c:\GridWorldCode` (on Windows) or inside your home folder (on Linux/Unix/OS X).

## 2. Check your Java installation.

The GridWorld case study uses Java 5, so you must have this or a more recent version of Java installed before you work with the case study. Java can be downloaded for free from Sun Microsystems.

## 3. Install the JAR file.

The GridWorld code that forms the core of this case study is packaged in a JAR file. This can be installed on your system in the same way as any other package contained in a JAR file. For example, if you use code distributed with your textbook that is stored in a JAR file, you would install and access the `gridworld.jar` file in the same way.

If you are unfamiliar with installing a JAR file, there are additional instructions for several systems below. Another good source for instructions on how to install a JAR file can be in the online help on your own development environment. For example, with JCreator you can simply click on the Help menu and from the index select “classpath” to find instructions. You might also check the index for “JAR.”

Once the JAR file has been installed for your system, you will be able to compile and run all the examples.

## Command-line Compiler

When you compile or run programs, add *both* the current folder *and* `gridworld.jar` to the class path. In Windows, use a semicolon to separate the current folder (denoted by a single period) from the path to the JAR file.

```
javac -classpath .;\path\to\gridworld.jar *.java
java -classpath .;\path\to\gridworld.jar MainClass
```

Replace `\path\to\` with the path of the folder that contains `gridworld.jar`, such as `c:\GridWorldCode\`. Replace `MainClass` with the name of the class that you want to run. Here is an example.

```
javac -classpath .;c:\GridWorldCode\gridworld.jar *.java
java -classpath .;c:\GridWorldCode\gridworld.jar BugRunner
```

If any of the folder names contain spaces, enclose the folder path in quotation marks, as shown below.

```
javac -classpath .;"c:\Program Files\GridWorldCode\gridworld.jar"
*.java
```

In Linux/Unix/Mac OS X, use a colon as a separator instead.

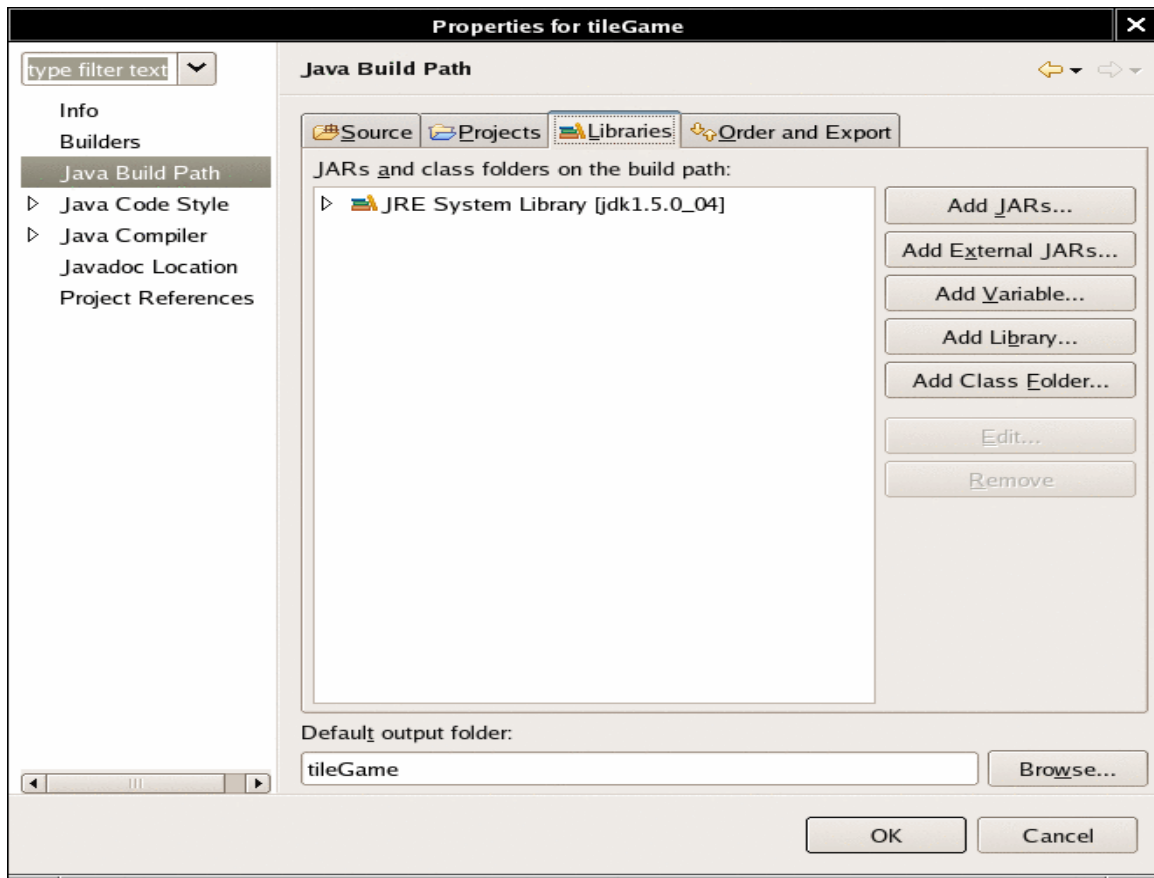
```
javac -classpath ./path/to/gridworld.jar *.java
```

Replace `/path/to` with the path such as `~/GridWorldCode`.

## Eclipse 3.1

You need to do the following for every project.

1. Start a project as usual.
2. Select the project in the “Package explorer” window.
3. Select **Project -> Properties** from the menu.
4. Click on **Java Build Path** and select the **Libraries** tab.

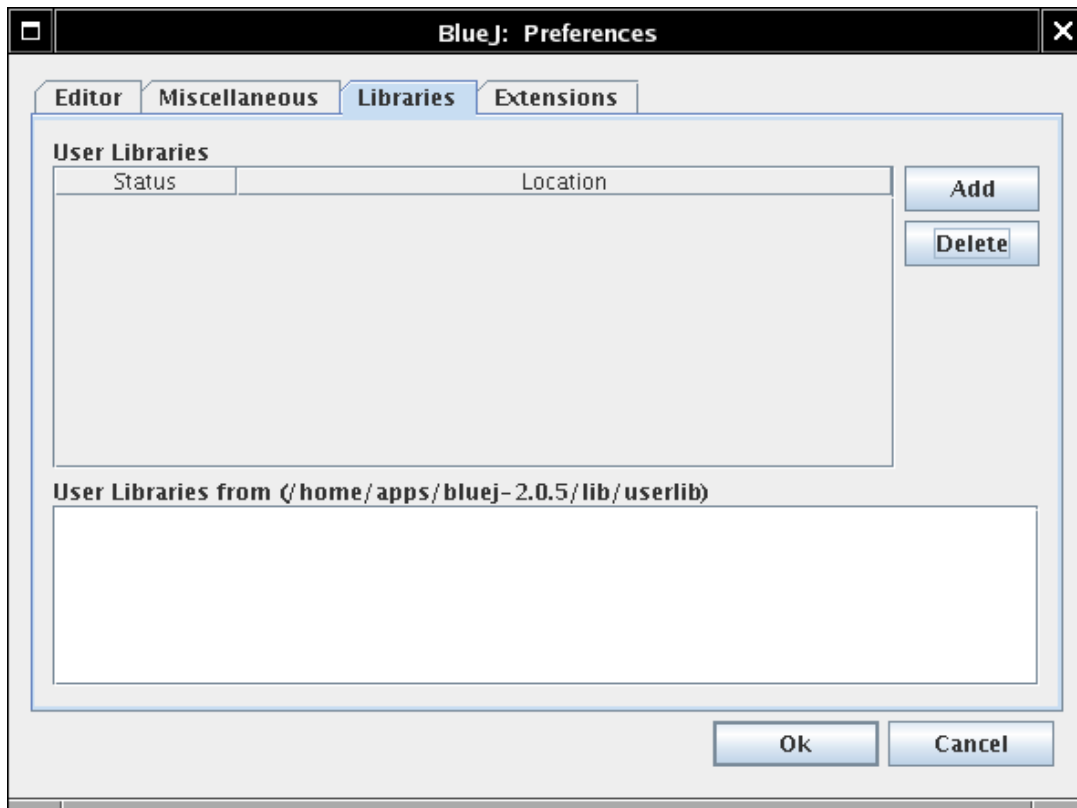


5. Click the **Add External JARs...** button.
6. Navigate to the folder containing `gridworld.jar`. Select `gridworld.jar`.

## BlueJ 2

You need to do the following once.

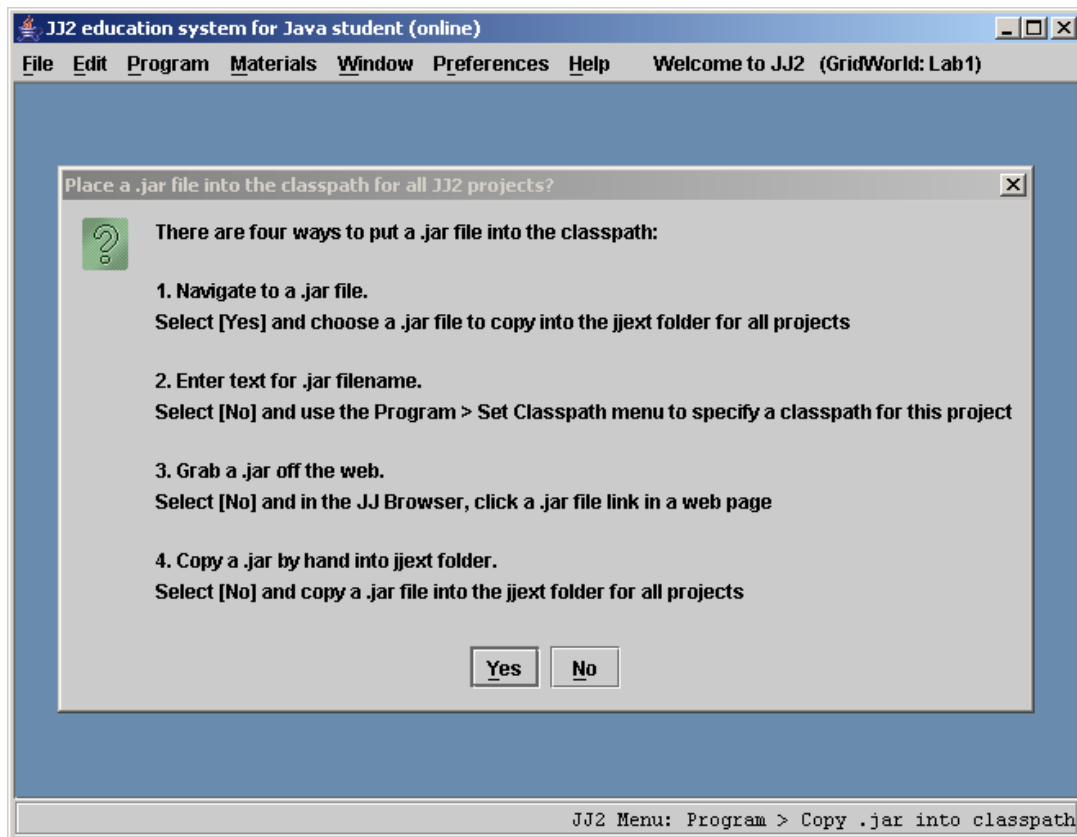
1. Select the menu option **Tools -> Preferences**.
2. In the resulting dialog, click on the **Libraries** tab.



3. Click the **Add** button.
4. Navigate to the folder containing `gridworld.jar`. Select `gridworld.jar`.
5. Restart BlueJ.

## JJ

1. Select the menu option **Program -> Copy .jar into classpath**.
2. A dialog is displayed that describes various options for copying .jar files into the classpath for the current JJ project or for all JJ projects, from a local file or from an Internet download.



3. Click the **Yes** button.
4. Navigate to the folder containing `gridworld.jar`. Select `gridworld.jar`.

## JCreator 3.5 (Windows only)

Create a project with the appropriate runner and other files in it. For example, the project can be the folder `firstProject`. From the menu select **Project -> Project Properties**. Click the **Required Libraries** tab and click **New**. In the **Set Library** dialog box, click the **Add** button, select **Add Archive**, then browse to add the JAR file. Give the new library a name. Finally, click the checkbox for the new library and click **OK**. You can now compile and run the GridWorld files from JCreator.

## TextPad 4.7 (Windows only)

From the TextPad menu select **Configure** → **Preferences** → **Tools**. Select **Compile Java** and in the parameters box type the following.

```
-classpath .;c:\GridWorldCode\gridworld.jar $File
```

Select **Run Java Application** and in the parameters box type the following.

```
-classpath .;c:\GridWorldCode\gridworld.jar $Base
```

You can compile and run the GridWorld code from TextPad. If any of the folder names have spaces, then you must put the whole folder path in double quotes.

```
.;"c:\Program Files\GridWorldCode\gridworld.jar"
```

## 4. Run the sample code.

Each example of code for this case study has a runner class that sets up and starts the simulation. The first such example is `BugRunner`, which is discussed in Parts 1 and 2 of the narrative. The file `BugRunner.java` is found in the `firstProject` folder inside the `projects` folder of the code distribution. Compile and run this class to start the GUI and experiment with the first example. If you get an error message about missing classes, check that you are including the JAR file, as explained in the preceding section.

If you use a development environment such as Eclipse or BlueJ, make sure to *make a project* that contains the `firstProject` folder. In Eclipse, use the **Create project from existing source** option. In BlueJ, use the **Open Non-BlueJ** option.

The example project for Part 2 is found in the `boxBug` folder. Compile the `BoxBug` and `BoxBugRunner` classes and run the `BoxBugRunner`. The examples discussed in Part 4 are found in the `critters` folder. There are two actor classes, `ChameleonCritic` and `CrabCritic`, and three runner classes, `CriticRunner`, `ChameleonRunner`, and `CrabRunner`. Compile the classes and run each of these programs.