AP Computer Science Summer Assignment

Due date: September 2015
Instructor: Mr. Alwin Tareen

Part A: Downloading and Installing the BlueJ Integrated Development Environment(IDE)

The following is a step-by-step tutorial for downloading and installing the BlueJ software that we will use in this class for developing Java programs. This tutorial goes through the process with Windows, but Mac users should be able to use this tutorial as well.

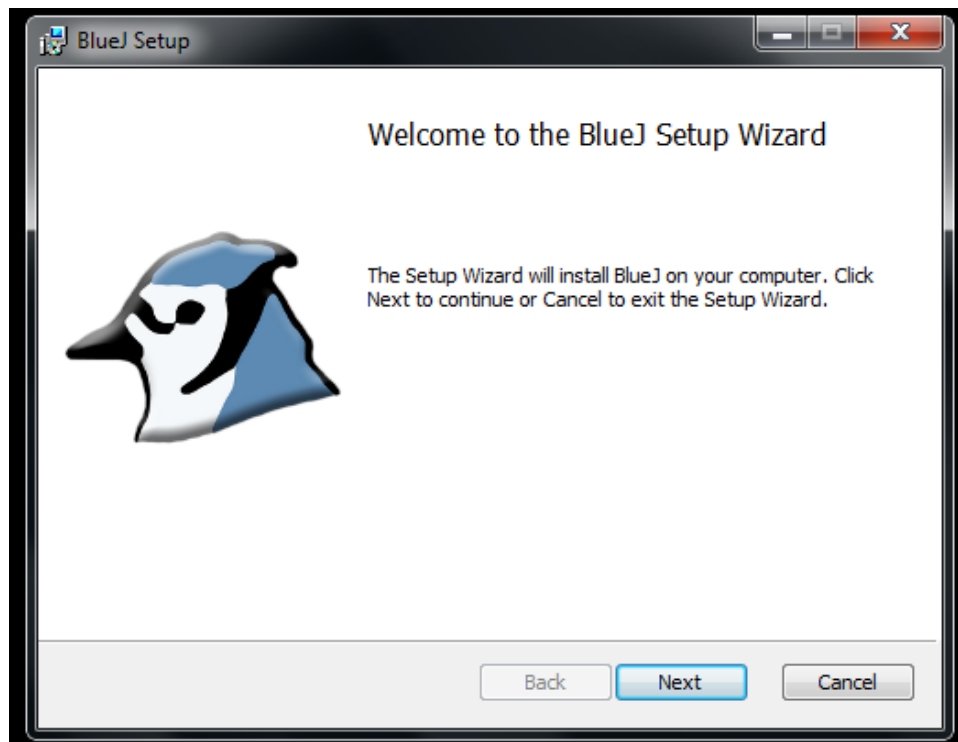1. Point your browser to the following URL: `www.bluej.org` and you should see the following:

2. Scroll down a bit, and you should see the following links. Windows users should download the link `BlueJ Combined Installer(includes JDK)`, and Mac OS X users should download the link that is appropriate for their specific operating system version.

Download and Install
Version 3.1.5, released 29 March 2015 (what's new)

Windows

BlueJ Combined Installer
(includes JDK)

OR

1. Oracle JDK

2. BlueJ Installer

Mac OS X

10.7[Lion].3 and newer:
BlueJ Installer
(includes Oracle JDK)

OR

10.7[Lion].2 and older:
BlueJ Installer
(uses Apple JDK)

3. Once you have finished downloading the file, double click on it. You should see the following dialog window. Click on `Next`.

BlueJ Setup

Welcome to the BlueJ Setup Wizard

The Setup Wizard will install BlueJ on your computer. Click
Next to continue or Cancel to exit the Setup Wizard.

Back   Next   Cancel

4. You should see this window, asking if you want to install just for you, or everyone. Selecting `Install just for you` is fine. Click on `Next`.
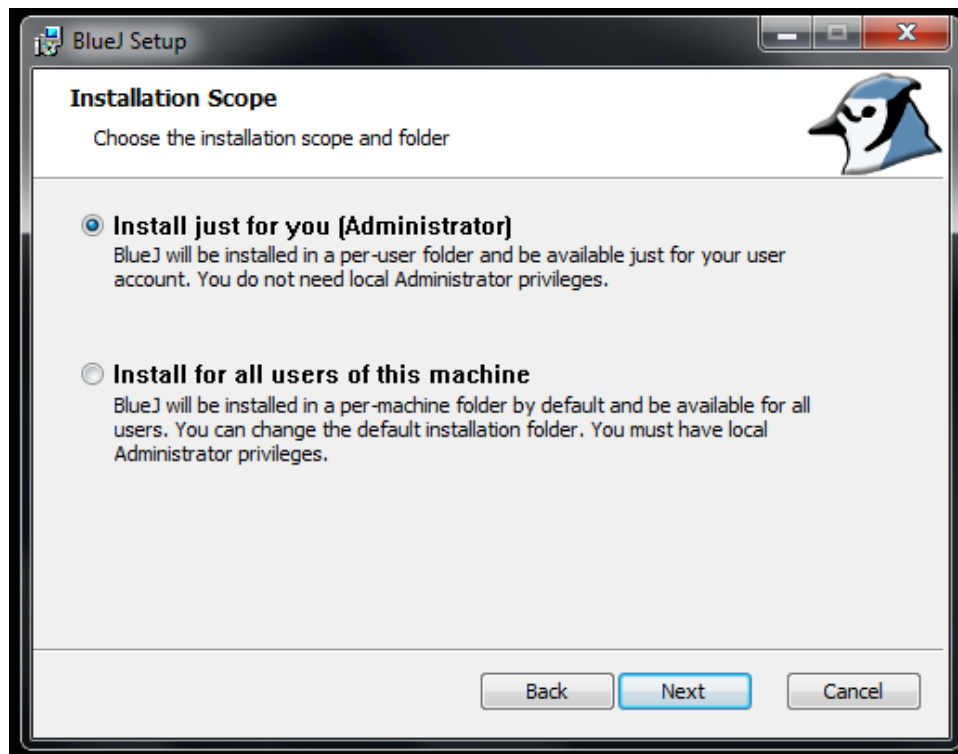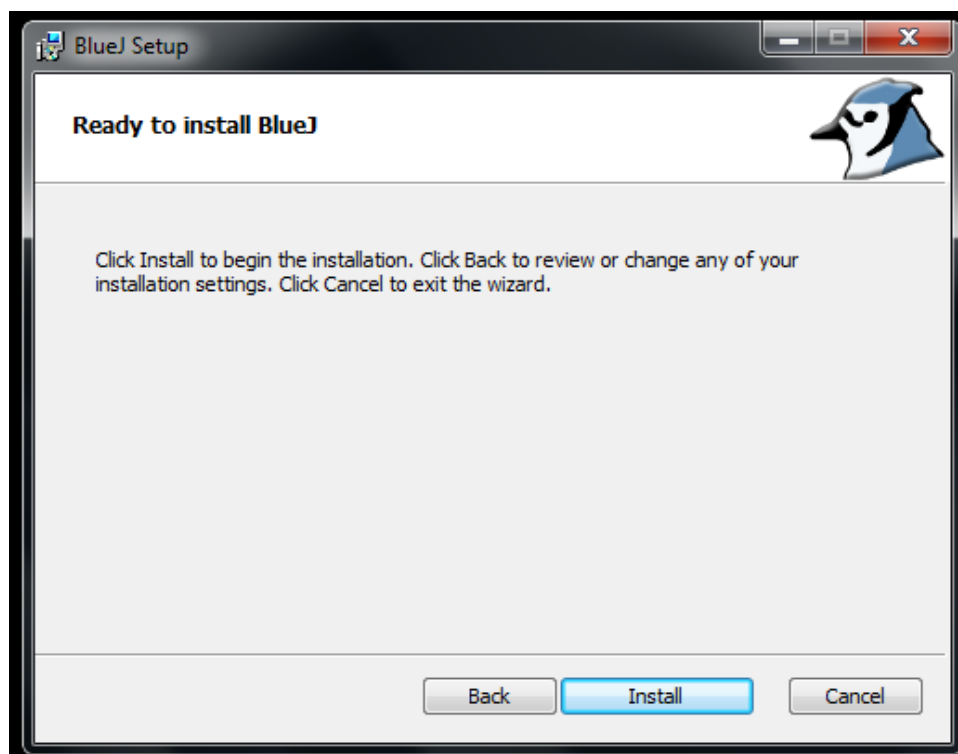

5. Next, you will be asked about file associations and shortcuts. These checkboxes are selected by default, which is fine. Click on `Next`.
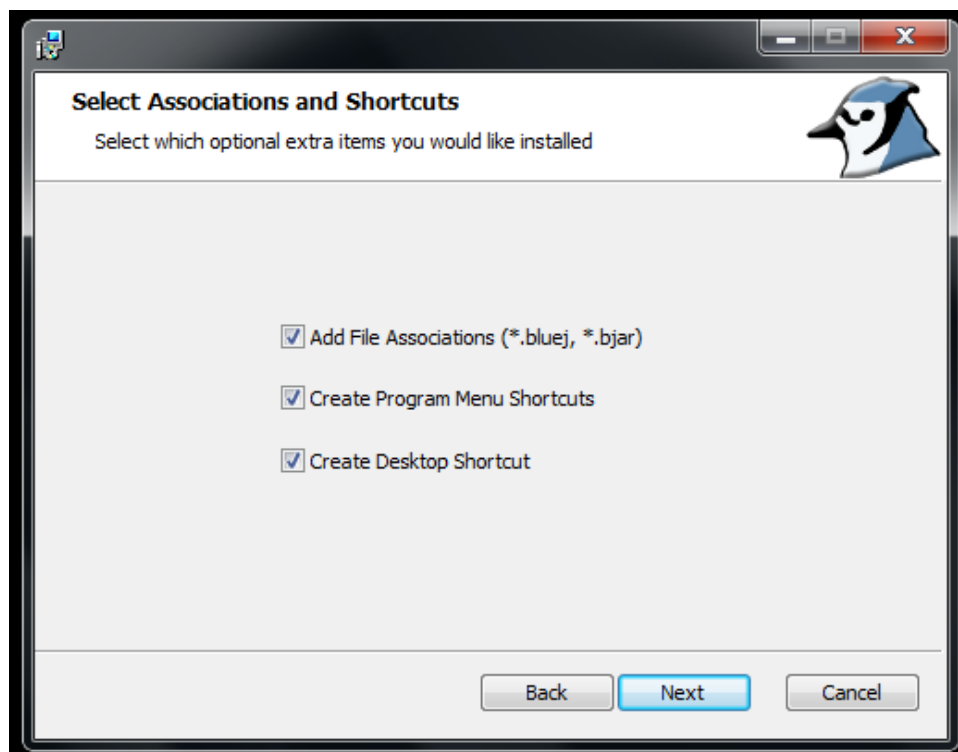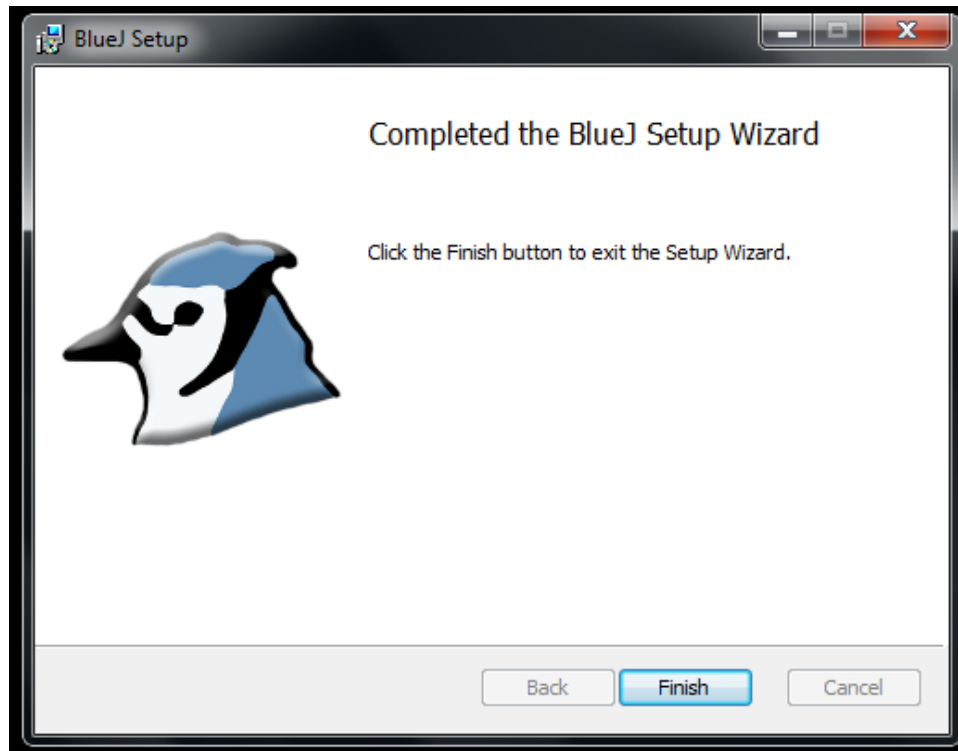
6. Then, you will be asked about a destination folder for the BlueJ program. The default location is fine. Click on `Next`.
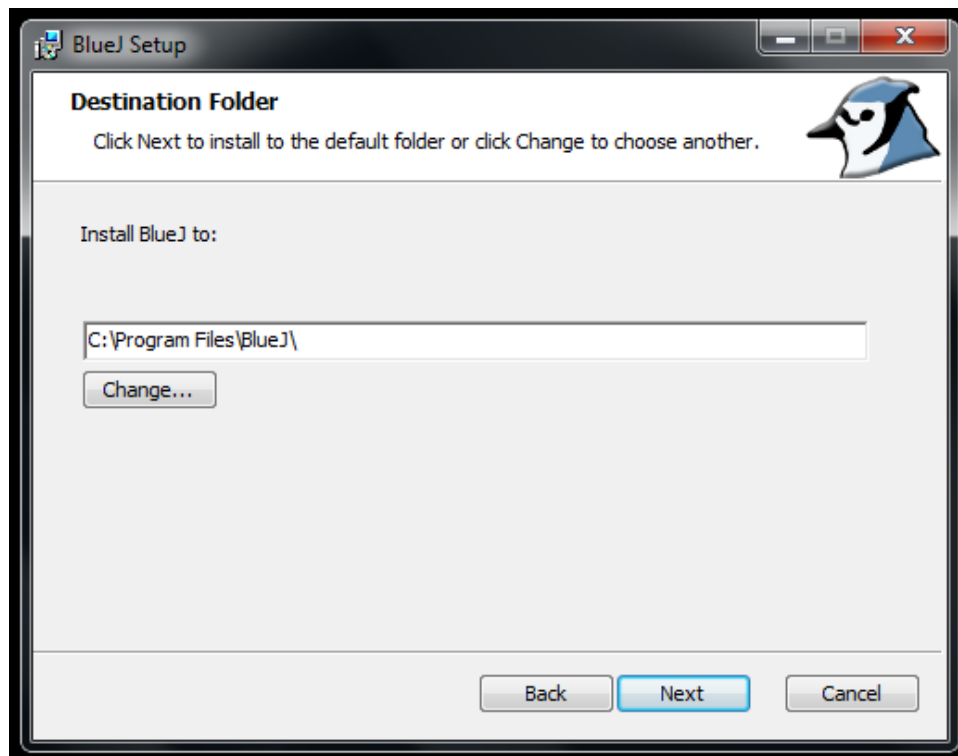
7. You will be presented with a `Ready to install BlueJ` window. All the settings are fine, so click on `Install`.
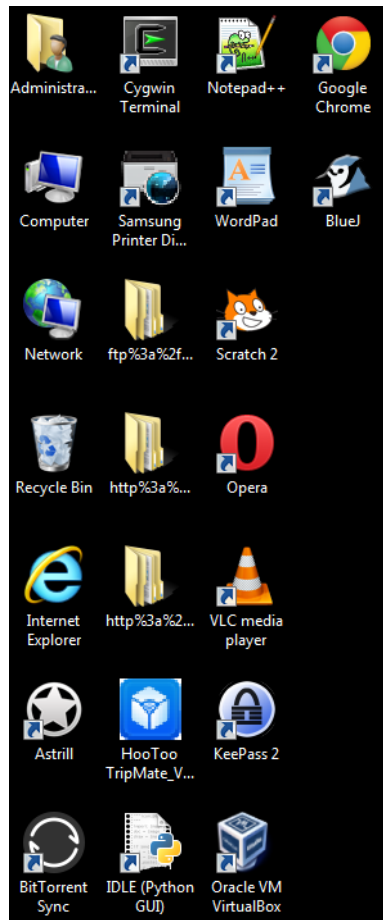
8. Finally, you will see a window comfirming that you have completed the BlueJ Setup Wizard. Click on `Finish`.
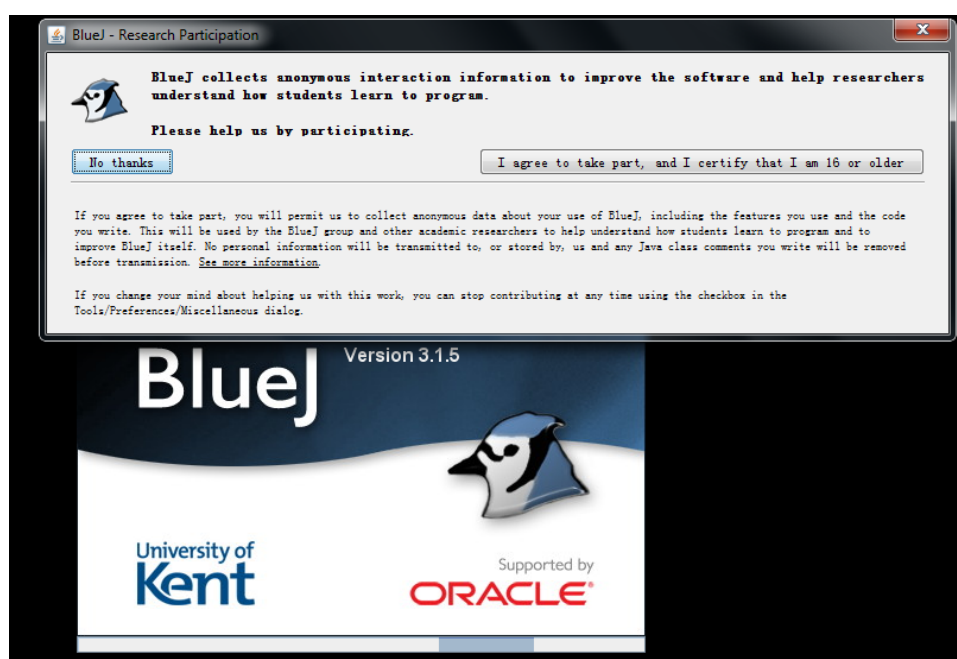
**Part B: Writing, Compiling and Running a Java Program in BlueJ**

1. You should see a BlueJ program icon on your desktop. Double-click this icon to run the program.


2. A window will appear, asking you to participate in some research. Click on No thanks.

3. The BlueJ Integrated Development Environment program will appear. This is the main user interface in which you will develop your Java programs.
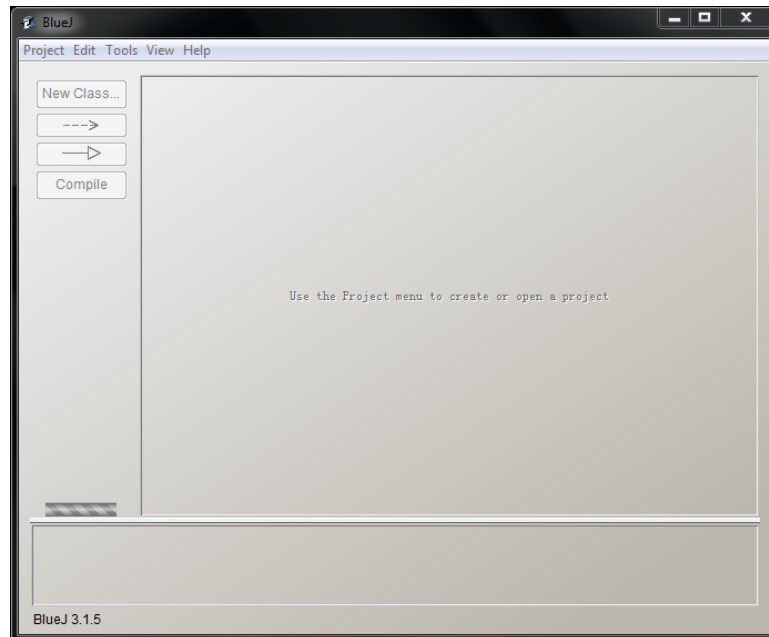

4. Click on the `Project` menu, then select `New Project...` and the `New Project` window will appear. You should choose a suitable place to put your Java programs. I have created a folder called `apCompSci`, and then a subfolder called `blueJCoding`, in which I will place my BlueJ projects. If you wish to create a similar folder scheme, select `Cancel`, then create the necessary folders using Windows explorer, then come back to this `New Project` window. Select the folder location where you want to place your project. I have chosen the location `blueJCoding`. Finally, type the name of your project into the `Folder name:` dialog box. Pay attention to what's happening here. I'm typing in the name of a **folder**, not a **file**, so there won't be any file extensions, such as `.txt` or `.java`, and the name of this particular folder is:
`summerAssignments`
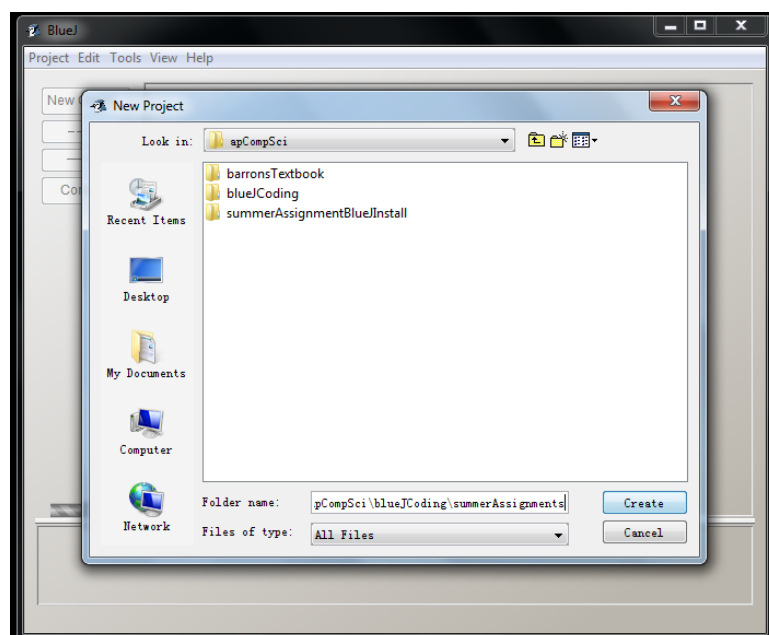
5. After creating a project folder, your BlueJ environment should like like the following screenshot. Now, we are going to create a Java program. Click on the button New Class...
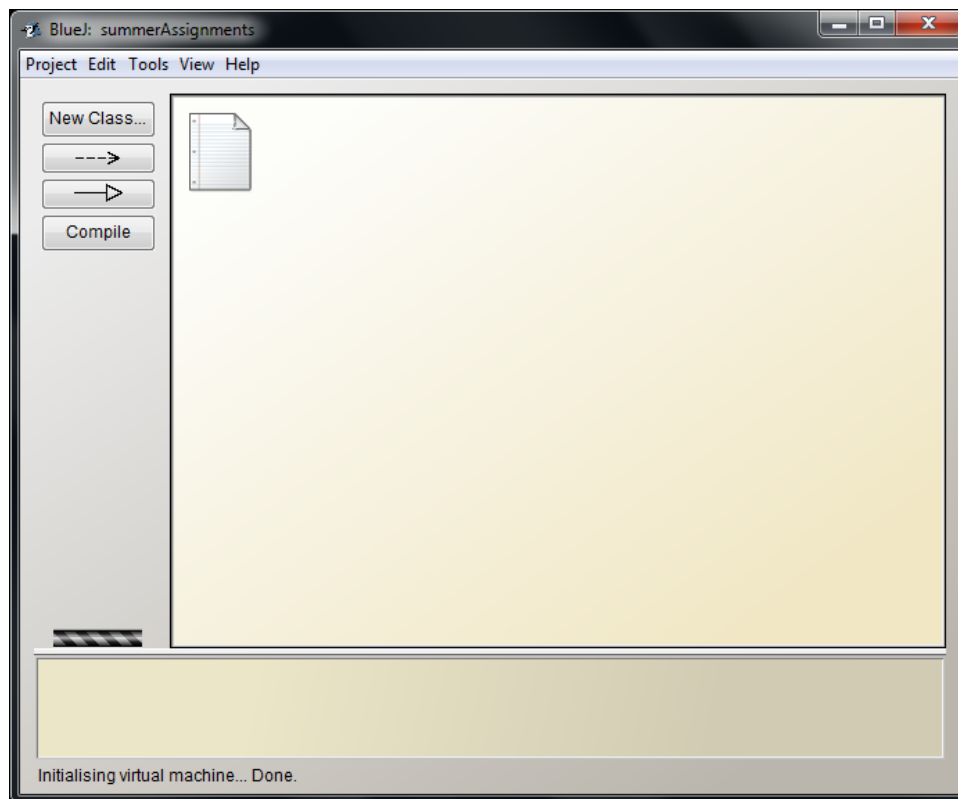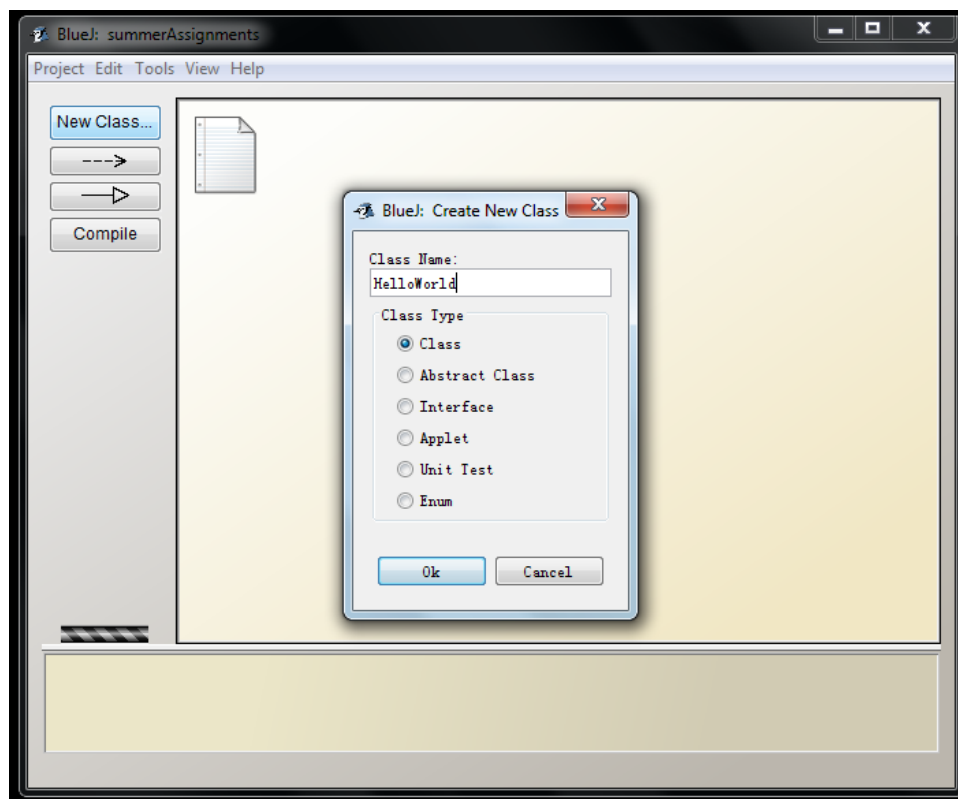

6. The BlueJ: Create New Class window should appear.
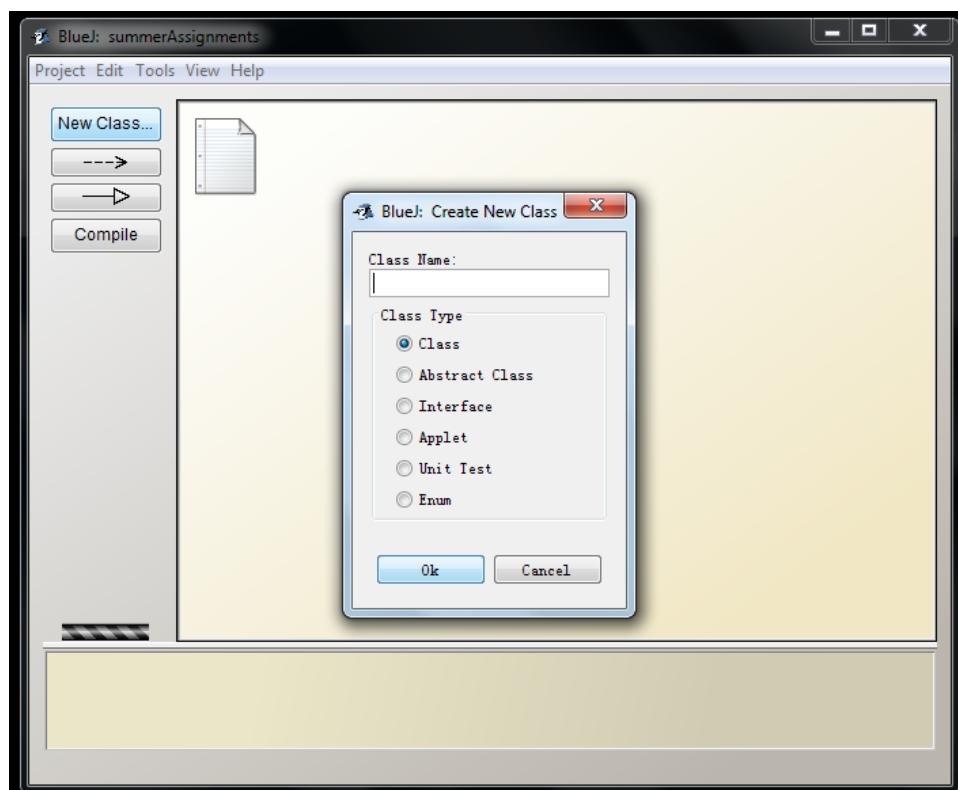
7. Type `HelloWorld` into the `Class Name:` dialog box, then click on `Ok`.


8. The BlueJ environment should look like the following screenshot. Notice that there is a new box in the canvas with the title `HelloWorld`. This represents the `HelloWorld` Java file. Now, we must type a Java program into that file. Double-click on the `HelloWorld` box.
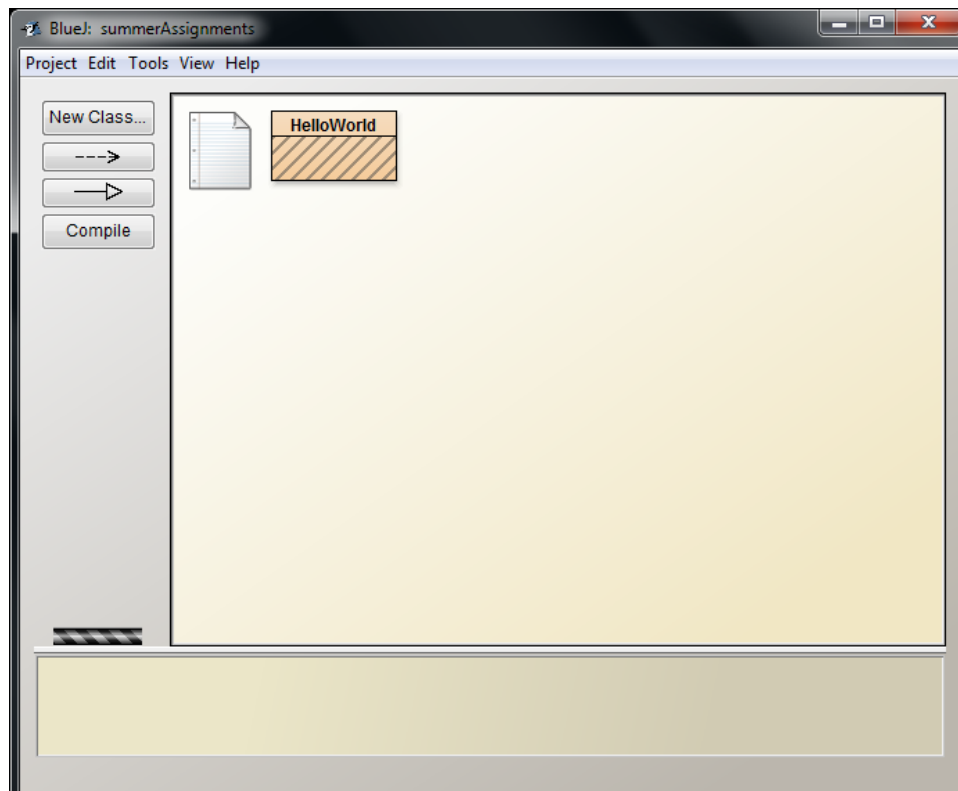
9. A new editor window should appear, which contains the Java program code of the `HelloWorld` file. I have resized the windows so they fit in one screenshot, but you should recognize the BlueJ environment on the left, and the editor window with the `HelloWorld` file on the right. BlueJ places a standard Java program into a new file by default, but we generally don't use this, so we should erase it.
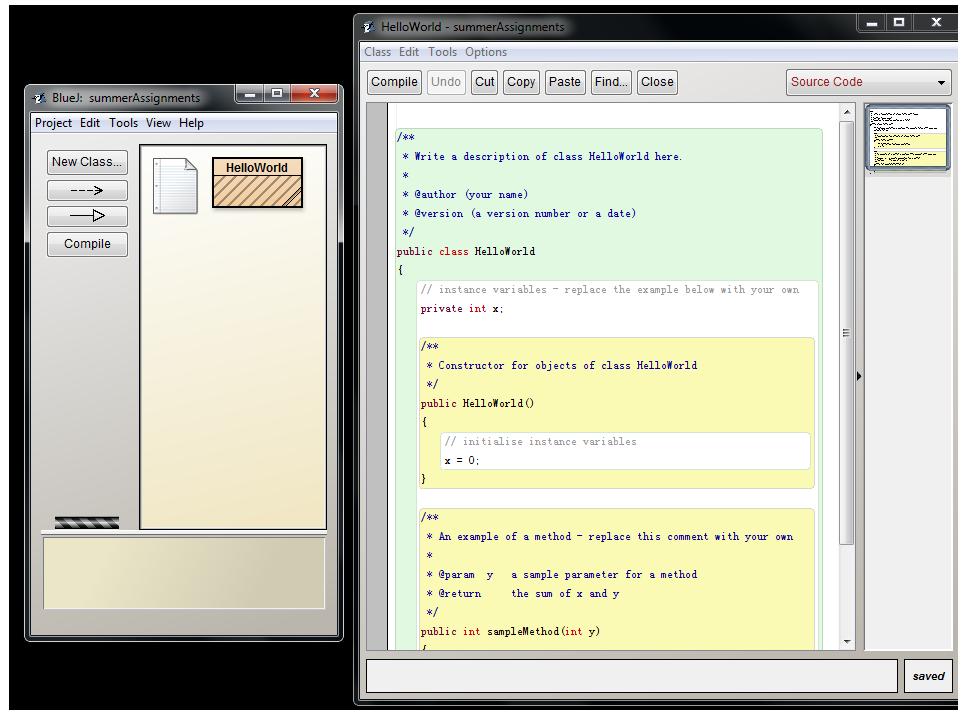
10. Click your cursor in the editor window, then type `Ctrl-a` to select all of the text in that window. You should see everything highlighted in yellow, like the following screenshot.
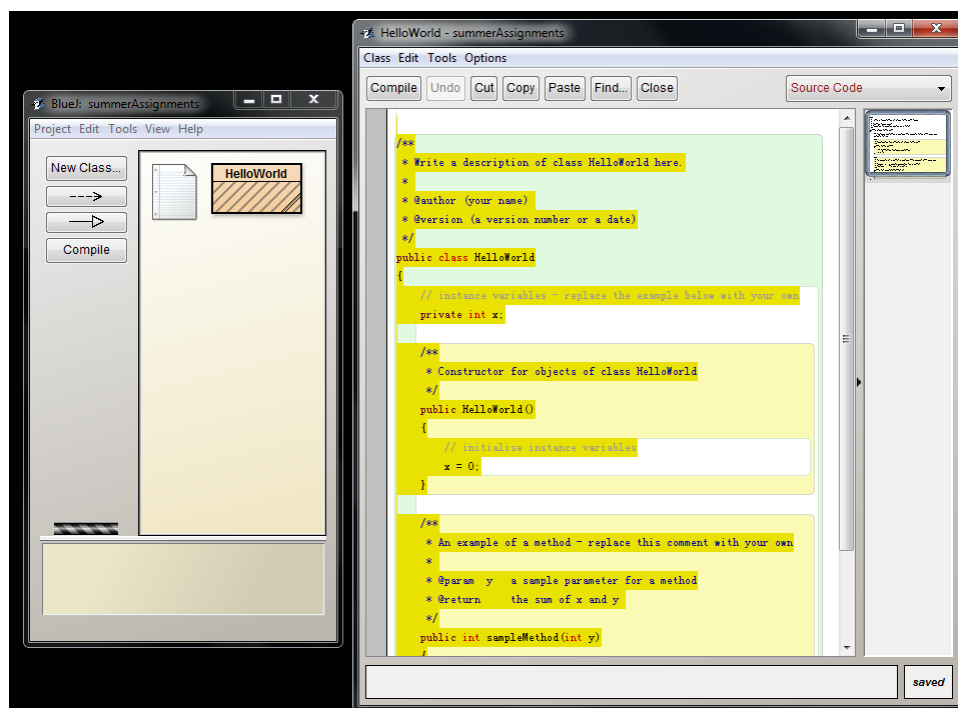
11. Press the `backspace` key to delete the selected text. You should see a completely empty text editor window, as in the following screenshot.
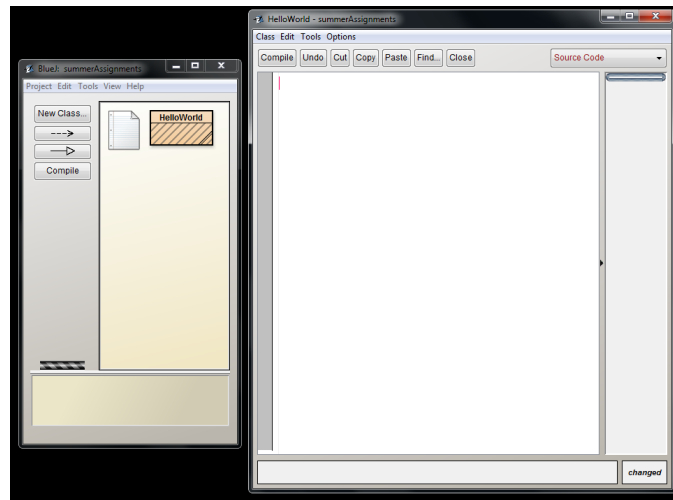

12. I have listed a simple Java program below, now you must type this into the editor window **exactly** as it is stated, with no mistakes, and with all of the strange punctuation characters included. As we will learn, these are actually *control* characters in Java. There is also a strange indentation pattern you must follow, where each indent is 4 spaces.

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello, world!");
    }
}
```
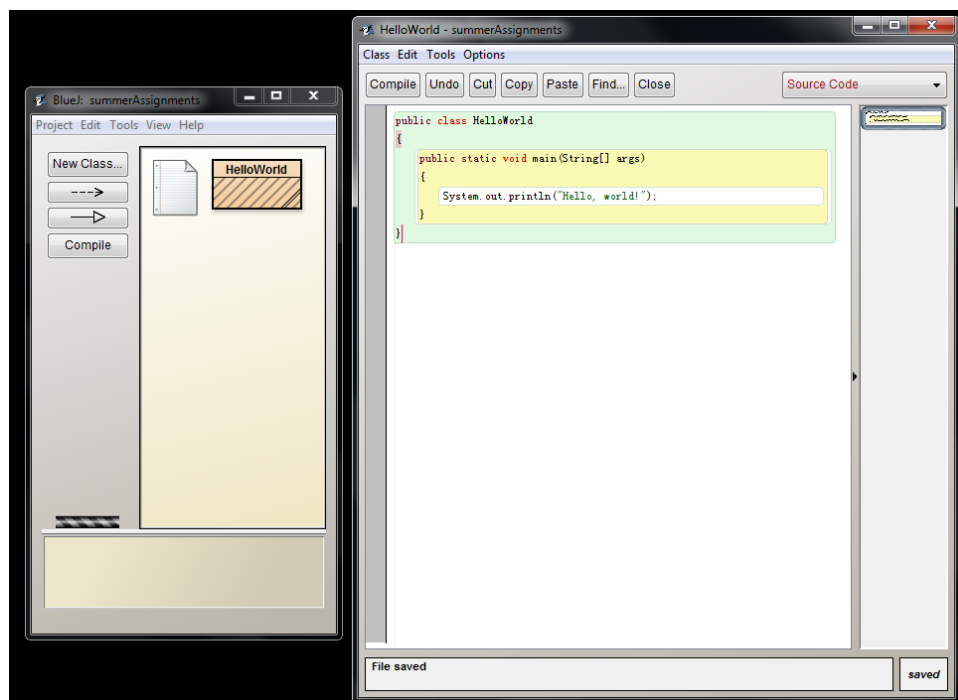
13. After you have typed in the Java program, and are satisfied that there are no errors, you must **compile** the program. Click on the `Compile` button on the text edtior, or click on the one in the BlueJ environment. They do the same thing. If you have typed your program correctly, you should see something like the screenshot below. Notice that the diagonal lines on the `HelloWorld` box are gone, and you see the phrase `Compiling...Done` at the bottom. If, however, you made some mistakes in your program, an error message will appear. You must correct all of the mistakes before the program will compile.
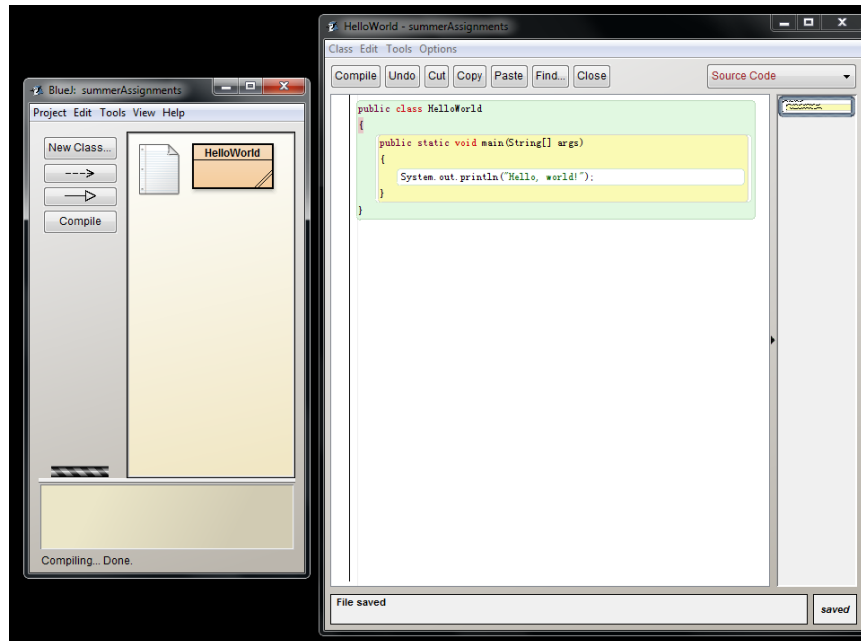
14. Our next step is to run the Java program. `Right-click` on the `HelloWorld` box, and you should see a small dialog box appear. Select the option listed as `void main(String[] args)`, it should be the second one. Then, a `Method Call` dialog box should appear, like in the following screenshot. Click on `Ok`.
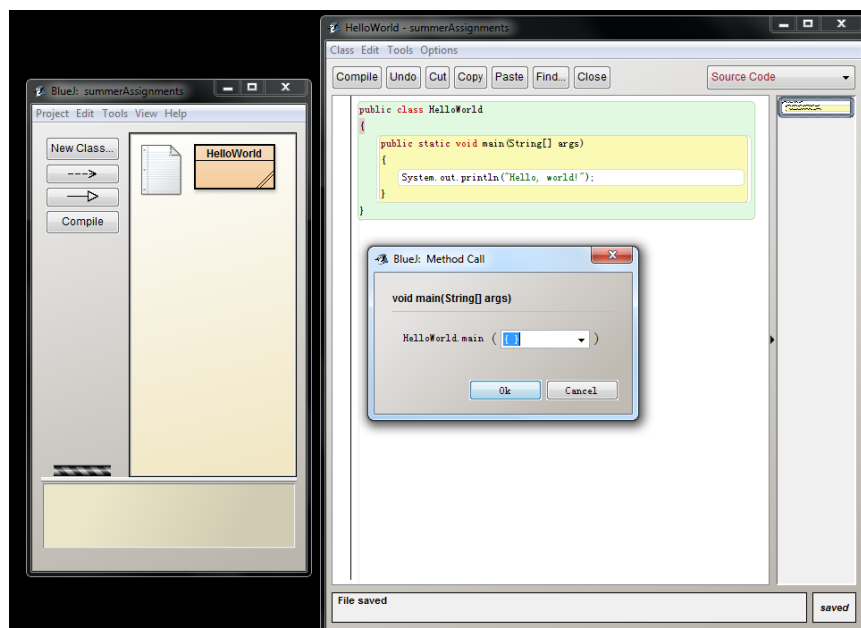
15. After clicking on `Ok`, a `Terminal Window` should appear. This displays the words `Hello, world!` which was exactly what we wanted the program to do. I have resized the `Terminal Window` and placed it at the lower left hand section of the following screenshot. If you get to this step, then congratulations! You have successfully run your first Java program!
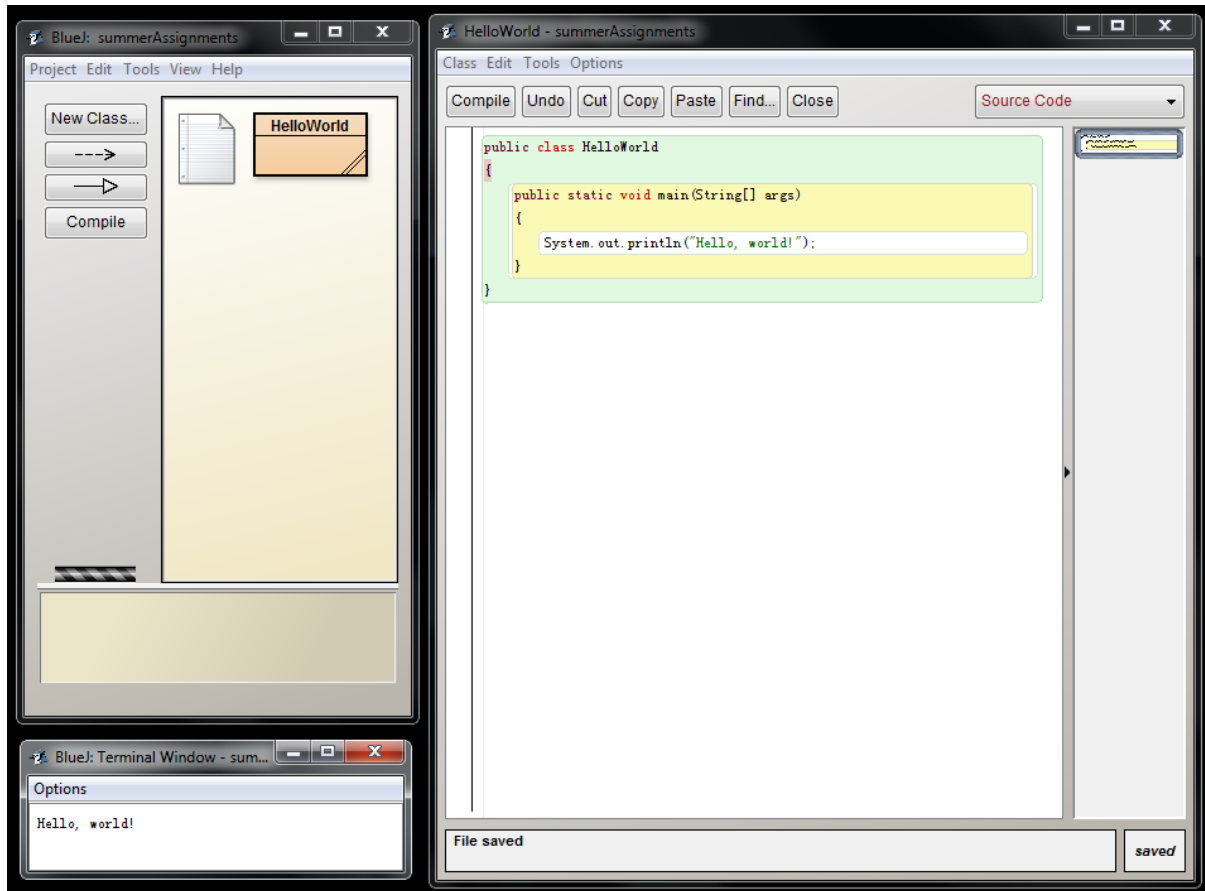
16. Now that you have finished this tutorial, you can use this **write-compile-run** cycle to create your own Java programs. The following are some excellent web resources to guide you. If you are able to gain some programming experience early on, it will be extremely beneficial for you. Good luck!

    www.homeandlearn.co.uk/java/java.html

    interactivepython.org/runestone/static/java4python/index.html

    codingbat.com/java