

# JAVA OOP: THE GUZDIAL-ERICSON MULTIMEDIA CLASS LIBRARY\*

R.G. (Dick) Baldwin

This work is produced by The Connexions Project and licensed under the  
Creative Commons Attribution License 3.0<sup>†</sup>

## Abstract

Learn how to use the Guzdial-Ericson Multimedia Class Library

## 1 Table of Contents

- Preface (p. 1)
- Figures and listings (p. 1)
  - Figures (p. 1)
  - Listings (p. 2)
- Preview (p. 2)
- Discussion (p. 5)
- Two versions of Hello World (p. 7)
  - A text version of Hello World (p. 7)
  - A graphic version of Hello World (p. 8)
- Miscellaneous (p. 9)

## 2 Preface

In the weeks and months to come, I will publish several modules that use a multimedia class library developed and made available by Mark Guzdial <sup>1</sup> and Barbara Ericson <sup>2</sup> of the Georgia Institute of Technology. The purpose of this module is to provide the information that you will need to obtain and use a copy of that library.

## 3 Figures and listings

### 3.1 Figures

- Figure 1 (p. 3) . Result of merging two images.

---

\*Version 1.2: Sep 18, 2012 9:22 pm -0500

<sup>†</sup><http://creativecommons.org/licenses/by/3.0/>

<sup>1</sup><http://www.cc.gatech.edu/~guzdial/>

<sup>2</sup><http://coweb.cc.gatech.edu/ice-gt/8>

- Figure 2 (p. 4) . Input image #1.
- Figure 3 (p. 5) . Input image #2.
- Figure 4 (p. 6) . Putting the library on the classpath.
- Figure 5 (p. 7) . Batch file for text version of Hello World.
- Figure 6 (p. 8) . Hello World in graphics.
- Figure 7 (p. 9) . Batch file for graphic version of Hello World. .

### 3.2 Listings

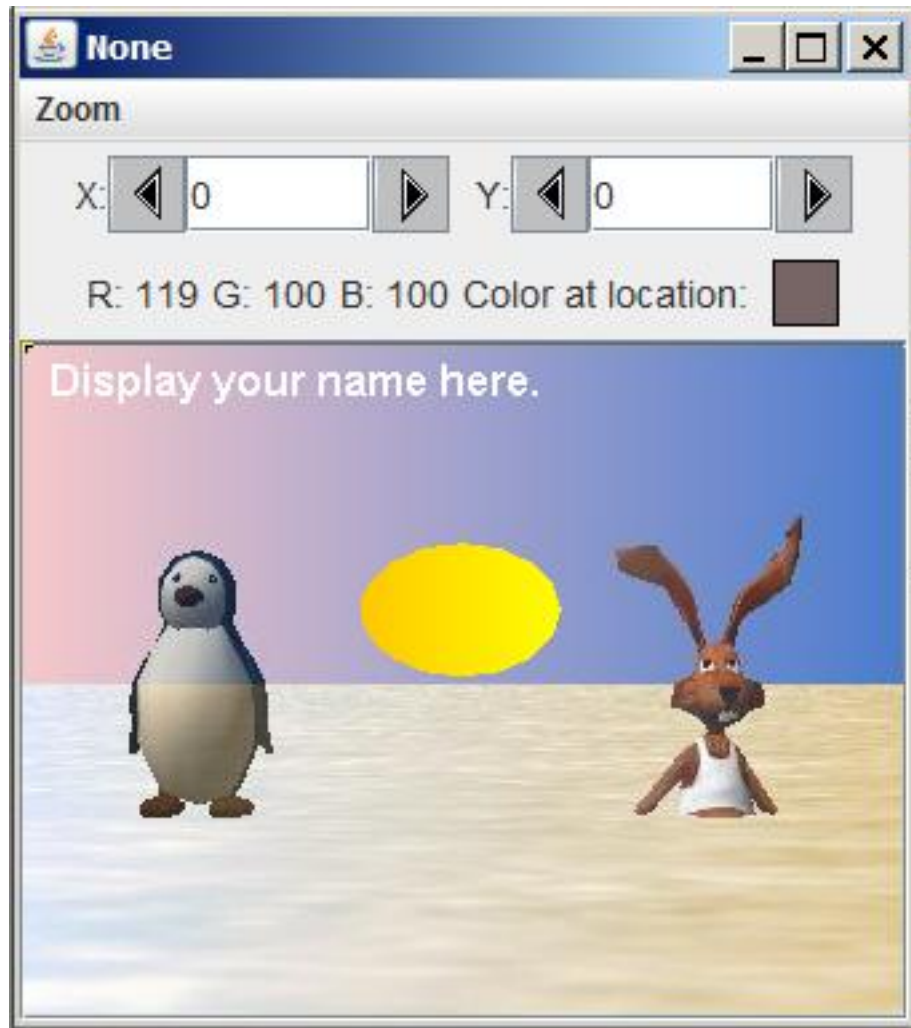
- Listing 1 (p. 7) . A text version of Hello World.
- Listing 2 (p. 8) . A graphic version of Hello World.

## 4 Preview

Among other things, the Guzdial-Ericson multimedia class library makes it practical to write object-oriented programs for the manipulation of images in an interesting and educational way. For example, the image in Figure 1 (p. 3) was produced by manipulating and merging the images shown in Figure 2 (p. 4) and Figure 3 (p. 5) .

---

Result of merging two images.



**Figure 1:** Result of merging two images.

---

One of the input images is shown in Figure 2 (p. 4) .

---

Input image #1.



**Figure 2:** Input image #1.

---

The other input image is shown in Figure 3 (p. 5) .

---

Input image #2.



Figure 3: Input image #2.

---

## 5 Discussion

The Guzdial-Ericson library was originally published by Guzdial and Ericson in conjunction with their book *Introduction to Computing and Programming with Java: A Multimedia Approach*<sup>3</sup>. While the book isn't free, the library is freely available and is published under a Creative Commons Attribution 3.0 United States License<sup>4</sup>.

As of July, 2012, the latest version of the library can be downloaded in a zip file named **bookClasses-3-9-10-with-doc.zip** at <http://home.cc.gatech.edu/TeaParty/47>.<sup>5</sup> Additional information is available at <http://coweb.cc.gatech.edu/mediaComp-plan/101><sup>6</sup>.

### Download the library

---

<sup>3</sup><http://www.pearsonhighered.com/educator/academic/product/0,3110,0131496980,00.html>

<sup>4</sup><http://creativecommons.org/licenses/by/3.0/us/>

<sup>5</sup><http://home.cc.gatech.edu/TeaParty/47>.

<sup>6</sup><http://coweb.cc.gatech.edu/mediaComp-plan/101>

In order to work with the programs in the modules that use the library, you will need to download a copy of the latest version of the library from the site listed above. To guard against the possibility of that link becoming broken at some point in the future, I am depositing a backup copy of the zip file containing the library on the cnx.org site and you can download it here <sup>7</sup> .

#### Prepare the library for use

Once you have downloaded the zip file, you will need to extract the folder named **bookClasses** from the zip file and store it somewhere on your computer's disk.

#### Library documentation

When you examine the contents of the **bookClasses** folder, you will see that in addition to source code and compiled class files for the library, the folder also contains **javadoc** documentation for the library in a folder named **doc** and some other material as well. Go to the **doc** folder and open the file named **index.html** in your browser to view the documentation.

#### Put the library on the classpath

You will need to put the path to the **bookClasses** folder on your classpath in order to incorporate classes from the library into your programs.

---

### Putting the library on the classpath.

```
del *.class
javac -cp .;M:\bookClasses *.java
java -cp .;M:\bookClasses Prob01
```

**Figure 4:** Putting the library on the classpath.

---

For example, Figure 4 (p. 6) shows three commands that you can execute at a Windows command prompt to

- delete the class files from the current folder,
- compile the source-code files in the current folder, and
- execute the compiled version of a program named **Prob01** in the current folder

*(The commands shown in Figure 4 (p. 6) assume that the **bookClasses** folder is in the root directory on the M-drive. The **bookClasses** folder will likely be in a different location on your computer. They also assume that you are compiling and running a program having its **main** method in a class named **Prob01** in a file named **Prob01.java** .)*

Be sure to include the period and the semicolon shown before the M in Figure 4 (p. 6) . This tells the Java compiler and the Java Virtual Machine to search first in the current folder before searching elsewhere on the disk for the required files.

You can also put the three commands in a batch file and run the batch file to avoid having to type the three commands each time you need to compile and run the program. In that case, you should add a **pause** command following the other three commands.

---

<sup>7</sup><http://cnx.org/content/m44148/1.2/bookClasses-3-9-10-with-doc.zip>

If you are using an IDE such as DrJava, JCreator, or JGrasp, see the instructions for the IDE to learn how to set the classpath in the IDE.

## 6 Two versions of Hello World

It is often useful to have the code for a simple program to test your system. This section presents two versions of the classical *Hello World* program. The first version is a simple text version that doesn't use Ericson's library. This version can be used to confirm that the Java Development Kit (JDK) is properly installed on your computer.

The second version is a graphic version that does require Ericson's library. This version can be used to confirm proper installation and use of the library.

### 6.1 A text version of Hello World

The simple program shown in Listing 1 (p. 7) causes the words ***Text Hello World*** to be displayed on the standard output device when the program is compiled and executed.

**Listing 1: A text version of Hello World.**

```
class TextHelloWorld{
    public static void main(String[] args){
        System.out.println("Text Hello World");
    } //end main
} //end class
```

As mentioned above, this program does not require the use of Ericson's library. It can be compiled and executed using a batch file containing the commands shown in Figure 5 (p. 7) . You should make certain that you can compile and execute this program before progressing to the graphic version.

---

**Batch file for text version of Hello World.**

```
echo off
del *.class
javac TextHelloWorld.java
java TextHelloWorld
pause
```

**Figure 5: Batch file for text version of Hello World.**

---

## 6.2 A graphic version of Hello World

Listing 2 (p. 8) contains the code for a graphic version of Hello World.

### Listing 2: A graphic version of Hello World.

```
import java.awt.Color;

class GraphicHelloWorld{
    public static void main(String[] args){
        Picture pix = new Picture(200,100);
        pix.setAllPixelsToAColor(Color.RED);
        pix.addMessage("Graphic Hello World",15,50);
        pix.show();
    } // end main
} //end class
```

The program shown in Listing 2 (p. 8) causes the image shown in Figure 6 (p. 8) to be displayed when the program is compiled and executed.

---

**Hello World in graphics.**



**Figure 6:** Hello World in graphics.

---

As mentioned above, this program does require the use of Ericson's library. It can be compiled and executed using a batch file containing the commands shown in Figure 7 (p. 9) . You should make certain that you can compile and execute this program before progressing to more complex programs involving Ericson's library.



---

### Batch file for graphic version of Hello World. .

```
echo off

del *.class
javac -cp .;M:\bookClasses GraphicHelloWorld.java
java -cp .;M:\bookClasses GraphicHelloWorld
pause
```

**Figure 7:** Batch file for graphic version of Hello World. .

---

(Once again, the commands shown in Figure 7 (p. 9) assume that the **bookClasses** folder is in the root directory on the M-drive. The **bookClasses** folder will likely be in a different location on your computer.)

## 7 Miscellaneous

This section contains a variety of miscellaneous information.

**NOTE: Housekeeping material**

- Module name: Java OOP: The Guzdial-Ericson Multimedia Class Library
- File: Java3000.htm
- Published: July 27, 2012
- Revised: September 18, 2012

**NOTE: Disclaimers: Financial** : Although the Connexions site makes it possible for you to download a PDF file for this module at no charge, and also makes it possible for you to purchase a pre-printed version of the PDF file, you should be aware that some of the HTML elements in this module may not translate well into PDF.

I also want you to know that, I receive no financial compensation from the Connexions website even if you purchase the PDF version of the module.

In the past, unknown individuals have misappropriated copies of my modules from cnx.org, converted them to Kindle books, and placed them for sale on Amazon.com showing me as the author. I receive no compensation for those sales and don't know who does receive compensation. If you purchase such a book, please be aware that it is a bootleg copy of a module that is freely available on cnx.org.

**Affiliation** : I am a professor of Computer Information Technology at Austin Community College in Austin, TX.

-end-