

**CMPSC 111**  
**Introduction to Computer Science I**  
**Fall 2014**

**Lab 7 for Sections 03 and 04**  
**23 October 2014**  
**Due Thursday, 30 October by 2:30pm**

## Objectives

To gain more experience in writing Java classes and methods, learn about the interactions between Java classes and methods, and practice reusing and extending existing Java programs.

## General Guidelines for Labs

- **Work on the Alden Hall computers.** If you want to work on a different machine, be sure to transfer your programs to the Alden machines and re-run them before submitting.
- **Update your repository often!** You should add, commit, and push your updated files each time you work on them. I will not grade your programs until the due date has passed.
- **Review the Honor Code policy.** You may discuss programs with others, but programs that are nearly identical to others will be taken as evidence of violating the Honor Code.

## Reading Assignment

To learn more about Java classes, methods and their structure, review Sections 4.1–4.5 in your textbook, paying particular attention to details about parameter passing in Java.

## Creating a Java Class and Methods for Your Masterpiece

In this laboratory assignment, you will reuse the programs from the fourth laboratory assignment. As you recall, in `Lab4.java` you have created a Java drawing using the methods from Java’s “Graphics” class, which was called from `Lab4Display.java`. For this laboratory assignment you will create a new Java class, called `Lab7Drawing.java`, and write your own methods that will draw different parts of your masterpiece by reusing statements from `Lab4.java`. In other words, instead of writing all Java commands to produce your Java drawing inside the “`paint`” method in `Lab4.java`, your goal is to separate the parts of your drawing into distinct pieces that can be drawn separately. Please see the course instructor if you have questions about these tasks.

Below is the list of programs you will need to either extend or create:

- **Reuse `Lab4Display.java`:** All you need to do is change the comment header, and rename this program to `Lab7Display.java`. Remember to change the name of the class as well as the name of the file—otherwise, this class will not compile correctly!
- **Create `Lab7Drawing.java`:** This class will represent your “drawable” object from laboratory assignment four. This class should have instance variables (for example, `x` and `y` coordinates), a constructor and methods that draw various parts of the drawing from your laboratory

assignment four submission. For example, if you created an animal in `Lab4.java`, then now you may decide to separate such drawing into three methods, where one method will draw the face, another method will draw the body, and the third method will fill the background and create a visual of the grass. After you create the instance variables, a constructor, and decide which methods you want to create, you can just copy and paste appropriate lines of code from `Lab4.java` into new methods that you will create in `Lab7Drawing`.

- **Modify `Lab4.java`:** First, rename this program to `Lab7.java`. Then, inside the `paint` method you need to create an instance of the `Lab7Drawing` class and call the appropriate methods that you created in `Lab7Drawing` class so that you are able to reproduce your drawing from laboratory four. Please note that the `page` may need to be a parameter passed to your methods. Of course, the methods that you create in your `Lab7Drawing.java` program may have more than one parameter, but they will most likely need to have at least one “`Graphics page`” object. The snippet of the method declaration below, from `Lab4.java`, should remind you where the `page` parameter comes from; please see the instructor if this is confusing.

```
public void paint(Graphics page) { ... }
```

### Program Requirements

- The `Lab7Drawing` class must have at least two instance variables, a constructor, and at least three methods that you create to draw different parts of your masterpiece.
- You must recreate your original graphic. If you would like to add something to it at this time, you may do so. But, please do not create a completely new masterpiece!

Remember to compile three files (`Lab7.java`, `Lab7Drawing.java` and `Lab7Display.java`), you may use the “`javac *.java`” command to compile all the `.java` files in your directory. You only need to run `Lab7Display.java` since it is the one that contains the `main` method, and then look for the pop up window (with a Java symbol). You should compile and run your programs incrementally after creating each method, instead of waiting until you finish creating methods for your entire drawing—this approach will best ensure that you do not create a defective program.

### Required Deliverables

For this assignment you are invited to submit electronic versions of the following deliverables through the Bitbucket repository. As you complete this step, you should make sure that you created a `lab7/` directory within the repository; then, you can save all of the deliverables here.

1. A completed, properly commented and formatted `Lab7.java`, `Lab7Drawing.java` and `Lab7Display.java` program. Please make sure that your programs include the comment header file with the Honor code, your name, date and the description of the program.
2. An output (your drawing) from running `Lab7Display` in the terminal window.

In addition to turning in signed and printed copies of your code and output, please share your program and the output file with me through your Git repository by correctly using the “`git add`”, “`git commit`”, and “`git push`” commands. When you are done, please ensure that the Bitbucket Web site has a `lab7/` directory in your repository with the four files called `Lab7.java`, `Lab7Drawing.java`, `Lab7Display.java`, and `output`. You should see the instructor or a teaching assistant if you have questions about submitting this assignment.