

# CS1322 Homework 3

## Due: Thursday, January 27 @ 6:00 pm

### Introduction

In this homework, you will utilize the Scanner class in order to read input from the command prompt. You will also create an applet like the ones you saw in Breakout 2.

### General

Up to now, we have not required any code comments just for simplicity. Comments are very important, however, for software maintenance and understanding of your code. There is an old adage in development that says, "Comment your code so that someone who has no idea what the code does can understand it, because in six months that someone will be you!"

In this assignment we will begin requiring you to comment your code and we will gradually build this level up over the term. For HW3 we will begin to "javadoc" our code. Javadoc is an application that is part of the JDK that you downloaded. You can run it on the command line by typing "javadoc \*.java" which will create documentation for all the classes in your current directory. If you are using JGrasp, you can invoke javadoc by using **Project->Generate Documentation** from the pull-down menu. Javadoc creates the nice web-page based documentation that we have been using in class (the API).

Remember from class we said that javadoc comments were a special form of comment. Javadoc recognizes a comment that starts with `/**` as a javadoc comment. For each new program assignment we will introduce new javadoc features. For HW3 we will create the following javadoc comments:

#### The class header:

Start your class header with the javadoc comment `/**`

Then write a short description of the class. Note that each line starts with an asterisk (\*) and space.

End the header comment with `*/`

For example:

```
/**
 * This is HW3, problem 1.
 * This class rids the Earth of bad guys.
 * @author James Bond gtg007m
 */
```

You should also place good comments into your code when the algorithm you are using is not obvious.

For the statement

```
int perimeter = a + b + c;
```

a good comment might be:

```
// calculate the perimeter by summing the lengths of the sides
```

A bad comment would be:

```
//add a b and c together
```

A frequent syntax problem developers have is mismatched braces. One commenting technique that can really help you is to comment the close brace with a short description of what it is enclosing.

For example:

```
    } // end switch(x)  
  } // end main method  
} // end class def Demo
```

We'll be looking at more javadoc and commenting issues in future HWs. For now, if you have more questions, check out the javadoc info page under the Java Resources page on the class website, or talk to your TA.

### Assignment 3.1: Inputting data from the command console

Write a class called TimeConvert, which when executed, prompts the user for to enter a number of seconds, and then it prints out the equivalent number of days, hours, and minutes. You do not have to worry about excess seconds. A sample run is given below.

```
Enter total number of seconds  
84939  
84939 seconds is equal to:  
    0 days  
   23 hours  
   35 minutes
```

Once again you may code this assignment in the main method. However, later on we will learn how to implement this code in a method.

**Hint:** Use the Scanner class to read input from the command console.

## Assignment 3.2: Creating a business card

Write a java applet that displays a business card of your own design. The card should include both graphics and text. At a minimum, your text should include your name. The graphics should include at least 3 or 4 different shapes from the Graphics class. You should also use at least 3 different colors on your card. Make the business card (applet) 400 pixels wide by 200 pixels tall, an aspect ratio similar to a real card. Make sure to include an HTML file that displays your applet in a web page. We will put some of the "coolest" cards off our class web page (with the student's permission of course).

In order to view your applet, you can run it as an applet from inside JGrasp, but you should also embed it in a web page for WWW viewing. You can do this in a web page editor or even a simple text editor. Since you may not know how to code html, the only line of code needed is given below:

```
<applet code="BusinessCardApplet.class" height=200 width=400></applet>
```

assuming that your applet file is called BusinessCardApplet.java.

### Turn-in Procedures

After you have finished the above assignments, turn them via [Webwork](#) You will be submitting three files:

- TimeConvert.java
- BusinessCardApplet.java
- The html file you created for viewing your applet