CSC207 Lab 3 — Java Classes

To earn lab marks, you must arrive on time and actively participate for the entire session.

1 Overview

This week, you are going to implement two Java classes. You’ll also document the classes by providing doc
comments for the Javadoc tool.

2 Log in and get things set up

s1 drives and s2 navigates.

1. In your subversion repository, create a lab3 directory.
2. Start Eclipse and select lab3 as a workspace to work in today.
3. Create a new Java Project called Week3Lab and a new package named week3lab.
4. In package week3lab, create classes named Author and Book as described below.

3 Implement class Author

Implement the Author class. This class should have two private instance variables of type String — one
for the author’s first name and another for the author’s last name.

It should also have the following methods (use Eclipse to generate these for you!):

1. A constructor that takes a first name as the first argument and a last name as the second argument.
2. Getters and setters for the two instance variables called getFirstName(), setFirstName(), getLastName(),
   and setLastName().
3. Override toString() so that it returns a String representation of this Author. For example, if the
   author is named Russel Winder, it returns the string "Winder, Russel".

4 Add Javadoc comments to class Author

“Javadoc” comments are comments written in a particular location and with particular notation that are
then used to generate HTML documentation for your own code in the same style as the Java API. For
example, here is Javadoc for a method in a class called Person:

/**
 * Returns the age of this Person on date date.
 *
 @param date the date relative to which this Person’s age is to be computed.
 @return the age, in years, of this Person on date date, or 0 if this Person was
 not yet born at that time.
 */
public int ageAt(String date) {
The first sentence is a summary. Every parameter must have an \texttt{\@param} tag that is followed by the name of the parameter and what it means. If the method returns a value, it must have an \texttt{\@return} tag that specifies what it returns. If you need to look up more information about Javadoc, this page is useful: \url{http://www.oracle.com/technetwork/articles/java/index-137868.html}

1. Write Javadoc for each of your methods in class \texttt{Author}. Notice that simply typing \texttt{/**} in Eclipse, and hitting return, causes Eclipse to lay out the structure of the Javadoc for you. Very handy!

2. Use Eclipse’s help feature to find the menu item that will generate HTML documentation based on the contents of your Javadocs, and then generate the documentation.

3. Look in the directory that you are using for your Eclipse workspace. In the same directory as your \texttt{bin} and \texttt{src} directories, you will now find a new directory called \texttt{doc}. Look inside it for a file called \texttt{index.html}. Open it in a browser and voila — beautiful documentation!

5 \textbf{Implement class Book, including Javadoc comments}

\textbf{Switch roles!}

Implement the \texttt{Book} class and include Javadoc comments.

This class should have four \texttt{private} instance variables: a \texttt{String} to store the book title, an array of \texttt{Authors} to keep track of the book’s authors, a \texttt{String} for the book’s ISBN, and a \texttt{double} for the book’s price.

It should also have the following methods:

1. A constructor that takes a book’s title, authors, ISBN number, and price, in this order.

2. Getters and setters for the instance variables: \texttt{getTitle()}, \texttt{setTitle()}, \texttt{getAuthors()}, \texttt{setAuthors()}, \texttt{getISBN()}, \texttt{setISBN()}, \texttt{getPrice()}, and \texttt{setPrice()}.

3. Override \texttt{toString()} so that it returns a string representation of the \texttt{Book} as follows: for a book titled “Developing Java Software” with authors Russel Winder and Graham Roberts, ISBN number 978-0470090251, and price 79.75, it returns the string "Developing Java Software (978-0470090251, $79.75, by Winder, Russel and Roberts, Graham)".

4. A \texttt{main} method that creates a \texttt{Book} titled “Developing Java Software” with authors Russel Winder and Graham Roberts, ISBN number 978-0470090251, and price 79.75, and then prints the \texttt{Book}’s string representation to standard output (using \texttt{System.out.println}).

Use Eclipse to generate the HTML documentation from your Javadocs.

---

Show your work (including your Javadoc!) to your TA.

---

Using a text editor (e.g., \texttt{nedit}, \texttt{kate}, or \texttt{pico}), in your \texttt{lab3} directory create a file named \texttt{partner.txt}. In that file there should be exactly one line and that line should contain only the CDF username of s2 (assuming you are using s1’s repository during this lab).

---

Add and commit the directories \texttt{Week3Lab}, \texttt{src}, and \texttt{week3lab}, and the two Java files to your repository. \textbf{Do not include} the \texttt{bin} folder or any of the “hidden” files generated by Eclipse.