

Assignment 5 – Literature Database

Maximum Points = 50

The purpose of this lab is to introduce you to the creation of objects of a user-defined class and to the Scanner, File, and Applet classes in the Java programming language. In this lab you will write a program that reads data from a file and then generates an applet that displays a listing of the novels and their data from the file.

- ❖ Your program must read the data from a file called novels.txt [HINT: You will need to use the Scanner class described on page 1018 (example from keyboard on page 157)] and example with a File class described on page 997 (examples in CougarView's Instructor Links and Files in the Scanner and URL Dissector folders in the Solutions folder.) NOTE: you will need to use a delimiter as illustrated in the examples to separate the data components.
- ❖ The data in the file will consist of five (5) novels (each novel will consist of a title, author's name, date published, and genre) where each component will be separated by the ampersand (@) symbol as shown below:

Time Machine@H. G. Wells@1895@Science Fiction
Gulliver's Travels@Jonathan Swift@1726@Satire

- ❖ Your program must include a class that will hold the Novel data (complete with constructor(s), get and set methods for each instance variable, and a toString method that returns the contents of the instance variables). Include an additional method getAge (returns the age of the novel from the date published.)
- ❖ Once your program has collected the data, you need to display the listing of the five novels using the Applet class described on page 991 (examples on page 63, in CougarView's Instructor Links and Files in the Applet folder in the Solutions folder, and <http://csc.columbusstate.edu/summers/NOTES/JAVA.HTM>) [NOTE: Applets use a method *init* instead of the *main* method]
- ❖ Your listing should have a title at the top in a larger font (and possible different font face and color) along with a graphic. The listing of the five novels should display the data from the file, PLUS the computed data for each novel.
- ❖ List somewhere on the bottom of the "page" the number of novels as well as the average age of the novels.
- ❖ **Modularize your program to minimize the amount of changes you would need to make if we change the number of novels.**

EXTRA CHALLENGES:

- add images.
- Ask the user for the file name.

Make sure that your program uses proper indentation and complete documentation. See <http://csc.columbusstate.edu/summers/NOTES/1301/style.htm> for guidelines.

The program heading should occur at the top of the program and should include:

```
/**
 * PROGRAM SPECIFICATIONS
 * NARRATIVE DESCRIPTION:
 *
 * @author (your name)
```

* @version (date)
*/

(Due before class on Wednesday, October 17, 2012) Submit your .java files containing your program and your timesheet documenting your time to the dropbox in WebCT..

Grades are determined using the following scale:

- Runs correctly.....:___/10
- Correct output.....:___/10
- Design of output.....:___/8
- Design of logic.....:___/10
- Standards.....:___/7
- Documentation.....:___/5

[Grading Rubric](#) ([Word document](#))