

CS 145: Preassignment 1

libmad

1 Description

Your primary goal in this homework is to acquaint yourself with expressing your deepest algorithmic desires in Java. In this case, your deepest desires are fairly narrow—you long for a Mad Libs application. As you implement this dream, you will learn about the indispensable `Scanner` and `String` classes.

2 Requirements

To receive full credit for this assignment, you must satisfy the following requirements:

- Write all code in a class named `Libmad` in a package named `pre1`. Case and spelling matter.
- Find a chunk of text from some source (Craigslist, a novel, a love letter from a forgotten high school sweetheart, etc.). Selectively strike out *exactly* 10 words from this text.
- Prompt the user to enter replacements for the removed words using commands like “Adjective:” or “Interjection:”. Removed words should be prompted for replacement in the same order in which they appear in the original text.
- Print the reconstructed text—with the replacements properly embedded—to `System.out`, *entirely in lowercase*. (Check out `String`’s `toLowerCase()` method.)

Feel free to create helper methods if you want to jump into a topic covered in the next homework. They could save you a bit of time on this assignment.

3 Example

Suppose our original text is “Excellence. Our measure, our motto, our goal.” We can’t strike out the requisite 10 words here, but let’s do 2: excellence and motto. When we run the `Libmad` class we should see the following interaction:

```
Noun: Hair
Noun: hope
hair. our measure, our hope, our goal
```

4 Submission

This is a preassignment, and preassignments are graded by SpecCheckers, not your instructor or TAs. The good news is you get to tweak your code until you have a perfect score. The bad news is that there's a little setup involved—and you must leave yourself enough time to work out kinks. Deadlines are not extended for botched configurations. Please follow these directions closely to test and submit your work:

1. Right-click on your project and choose Build Path → Add Libraries. Select JUnit, click Next, choose JUnit 4 from the dropdown menu, and click Finish. This only needs to be done once per project. So, if you use the same project for all your code in this class, you'll never need to execute this step again.
2. Download the SpecChecker JAR file.
3. Drag the JAR file onto your `pre1` package in Eclipse. Right-click on it, and select Build Path → Add to Build Path. It will migrate to the Referenced Libraries node in your project.
4. Run the JAR file by selecting it and hitting the Play button in the Eclipse toolbar. Run it as a Java application if prompted.
5. Address any errors. **The score you see will be the score you are assigned in the gradebook.**
6. Once you are satisfied with your score, choose a directory in which `pre1.zip` will be saved. This file contains a compressed version of `Libmad.java`.
7. Drop your submission into `W:\c s\CJohnson\cs145\<YOUR-USERNAME>`. You may overwrite this file as often as you like before the deadline. Just make sure the directory contains only one *.zip file with lowercase `pre1` in its name.