

ComS 227: Homework 3
Unforgetter: A Recurring-event Reminder
Due: October 29, *before* 11:59 PM

1 Your Job

Your primary task on this homework assignment is to learn about `ArrayLists` and iteration. Along the way you will learn more about static methods, organizing code into classes, and conditional statements. You will learn these topics by implementing calendar software for reminding users of annual events like birthdays and anniversaries.

There are a number of classes you will need to write to complete this assignment. The more pieces to a problem, the more complex it may seem. However, no single one of them is overly complicated—which is exactly the benefit of object-oriented programming. You need (and are encouraged) to solve only one piece at a time.

Programming with multiple classes is a lot like putting together a puzzle. Simply seeing how they fit together solves much of the problem, but you must give yourself ample time to see this connectivity. **Do not endanger your learning by starting too late or not asking questions.**

2 Package `hw3`

All classes you create for this homework should be in a package named `hw3`.

3 Classes

There are four classes you need to write to make this application meet its specification. They are:

1. **Utilities**. A collection of `static` helper methods.
2. **Event**. A recurring holiday.
3. **EventManager**. Object for dealing with a collection of `Events`.
4. **UnforgetterTUI**. Textual user interface for querying events.

Rather than describe what each class needs to do here, complete Javadoc documentation is provided on WebCT to describe the classes' public interface. Each class depends only on the classes listed above it, so you can be encouraged to complete them in this order. Also, you should test after completing each one before moving on to the next. This way if you find an error, you know it must be in recently-written code.

While you have freedom in how you implement the classes, there are a few guidelines you should follow:

- Do not add any additional `public` methods, variables, or constants to any class. Anything you add should be `private`.
- **EventManager**'s constructor needs to be tagged with special Java code indicating that an error will be generated if a dates file cannot be found. It should look like:

```
public EventManager(String datesFile) throws FileNotFoundException {
```

We will talk more about exceptions later.

- Since the number of events in a file is not known ahead of time, `EventManager` should store the list of `Events` that are read in from the file in an `ArrayList`.
- It is not required that you check for valid input in `UnforgetterTUI` at this point in time. Just make sure your code works when input is valid. Though it is not required, you will benefit from writing code to handle bad input.

A fifth class containing only a `main` method is provided at the end of this document. It will glue your classes together into a working application.

4 Submission

In a class this size, it is extremely important that you submit your assignment correctly to ensure timely grading and feedback. It's so important that you will lose points for incorrectly submitting. Follow these steps to create a proper submission:

1. Select your classes `Event`, `EventManager`, `Utilities`, and `UnforgetterTUI` in Eclipse's Package Explorer. You can select multiple classes by selecting the first and holding down the Control key while selecting the others.
2. Right-click on any of the selected classes, select Export, expand Java, select JAR File, and click Next.
3. Make sure the option regarding exporting class files is unchecked.
4. Make sure the option regarding exporting source files is checked.
5. Click Browse in the dialog that pops up and navigate to the directory where you wish to save your submission. (Take note of this directory. You will need to find it again!) Name your archive `hw3.jar`. *No other names are allowed.*
6. Validate that your archive is in the correct format by running the HW3 Validator on WebCT, pointing it to your `hw3.jar` file, and addressing any errors.
7. Submit your archive on WebCT. Make sure WebCT reports it as submitted! **You should see a yellow confirmation notice in WebCT and receive a confirmation email.** If you do not, your submission has not been received.

5 Extra

You are invited to share your event text files and any testing code on the WebCT discussion forums.

6 Unforgetter

```
package hw3;

import java.io.File;
import java.io.FileNotFoundException;
import javax.swing.JFileChooser;

/**
 * An application for getting reminders of annually recurring events like
 * birthdays, anniversaries, and tax days.
 */
public class Unforgetter {

    /**
     * The application driver. An EventManager is created using the events from
     * the file chosen by the user. Then, a TUI is run to let the user query the
     * events.
     *
     * @param args
     * Unused.
     */
    public static void main(String[] args) {

        JFileChooser chooser = new JFileChooser();
        int selection = chooser.showOpenDialog(null);
        if (selection == JFileChooser.APPROVE_OPTION) {
            File f = chooser.getSelectedFile();
            try {
                EventManager manager = new EventManager(f);
                UnforgetterTUI tui = new UnforgetterTUI(manager);
                tui.listen();
            } catch (FileNotFoundException e) {
                System.err.println("The file " + f.getName() + " couldn't be read.");
            }
        }
    }
}
```