Spring 2017 CS151 Section 6
Instructor: Dr. Angus Yeung
Assignment 3
Soft copy due: Refer to Canvas
No hard copy submission is required.

[Important] Please delete all package statements from the source file. Do not zip your workspace. That means when I unzip the file, I should be able to see the directories 5.2, 5.4, 5.8, 5.10, 5.12, and mvc which immediately contains `.java` files without subdirectories. Since the violation of this guideline will greatly degrade the efficiency of grading procedure, I decided to enforce it by stipulating a penalty on the case of violation.

Submission: Create a directory named as the question number and save the required solutions in the directory. For example, create a directory named 5.2 and save `answer.txt` in the directory. Do the same for the rest of questions.

Zip the directories that contain your solutions into hw3.zip and submit it on Canvas.

Problems:

- 5.2: `ObjserverTester`.java and all required classes to run the program. Make sure to follow the Observer pattern. Note: The solution of 5.1 is available at the book publisher's web site.

  Exercise 5.2. Improve Exercise 5.1 by making the graph view editable. Attach a mouse listener to the panel that paints the graph. When the user clicks on a point, move the nearest data point to the mouse click. Then update the model and ensure that both the number view and the graph view are notified of the change so that they can refresh their contents. *Hint:* Look up the API documentation for the `MouseListener` interface type. In your listener, you need to take action in the `mousePressed` method. Implement the remaining methods of the interface type to do nothing.

- 5.4: `SliderTester`.java and all required classes to run the program.

  Exercise 5.4. Implement a program that contains a slider and a car icon. The size of the car should increase or decrease as the slider is moved.

- 5.8: `MailSystemTester`.java and all required classes to run the program. Use two telephone handsets as 5.6 described and add scroll bars to the text areas of the telephone frame. Note: The solution of 5.7 is available at the book publisher's web site and you may use it.

  Exercise 5.8. Add scroll bars to the text areas of the telephone frame.

- 5.10: Write your answer in answer.txt.

  Exercise 5.10. You can give a title to a border by using the `TitledBorder` class. For example,

  ```
  panel.setBorder(new TitledBorder(new EtchedBorder(),
        "Select one option"));
  ```

  Which pattern is at work here? Explain.

- 5.12: `DecoratorTester` java and all required classes including `EncryptingWriter` and `DecryptingReader`. Test your classes in the `DecoratorTester`.

  > Exercise 5.12. Supply decorator classes `EncryptingWriter` and `DecryptingReader` that encrypt and decrypt the characters of the underlying reader or writer. Make sure that these classes are again readers and writers so that you can apply additional decorations. For the encryption, simply use the *Caesar cipher*, which shifts the alphabet by three characters (i.e., A becomes D, B becomes E, and so on).

- mvc: Write `MVCTester`.java. When the program starts, the initial screen displays a button labeled "add", a blank text area, and a text field. A user places a line in the text field and clicks on the add button. Then, the text area displays the line. Each time the user enters a new line in a text field and clicks on the add button, the text area is updated displaying previously entered lines and the new line. The following picture shows the snapshot of the program output right after two lines are added.

  To get a credit, the following requirements have to be satisfied.

  - The program follows the MVC model.
  - Listeners are implemented in an anonymous class.
  - Model is a separate class from the client (test) program.
  - Indication of which part of your program serves as model, controller or view.

  Think of the responsibilities of model, view and controller. I will see if the responsibilities are appropriately placed.