Assignment Requirements:
Use JavaScript and HTML to create a text adventure (which may optionally include images). Your text must be displayed in an HTML label tag. Player choices must be displayed on HTML buttons. For example:

See the class website for an interactive version of the above “Adventure”.

For each different button the player might select, your program must use JavaScript to change the label text and the button text to a new game state.

Your adventure must have at least 15 different screens (sets of text and options) where screen has at least 2 options or is an end screen (zero options). The shortest path from the start screen to an adventure end screen must be at least 4 screens (counting the start and end screen).

Paths from the start screen to the end screen may join, but must not contain any loops. That is, once the player sees a particular screen, it must be
impossible for the player to ever see that screen again (unless the player reloads the page).

Your text adventure may be on any topic you like. The example above starts a fantasy adventure. Another very different option might be a car diagnostic tutorial. For example:

The car fails to start. Which best describes your observation?
1. The car made no noise at all.
2. The car made a small click - click - click sound.
3. The car started to turn-over, but never caught.
4. The car started, but, then quit.

Grading Rubric [20 points total]:
[1 Point]: Submit in Blackboard a file named lab5_yourname.zip of your project folder including the .html AND ALL images and any other resources your program needs (it is optional to include p5.js as the graders all have this file and it is always needed).

[4 Points]: The buttons use JavaScript to change what is displayed.
[4 Points]: There are at least 15 different screens (sets of text and options).
[4 Points]: There are at least 2 options on every non-end screen.
[3 Points]: The shortest path from the start screen to an adventure end screen is at least 4 screens (counting the start and end screen).
[2 Points]: There at least two images that load on different screens. These two images must NOT show on all screens, only show on some screens. Of course, you are welcome to have other images and some of the extra images may, if you want, show on all screens.
[2 Points]: The adventure (story, tutorial, ...) makes sense.

NOTE: The example uses the draw() method to display slashes in the canvas when the player dies. However, drawing in the canvas is not a requirement. You may, if you want have a totally empty draw() method.

NOTE: The goal of this lab is for you to demonstrate mastery of changing html form JavaScript. If you want to demonstrate this mastery in a different way (such as dynamic graphs or web queries) then propose to me (joel) what it is you want to do. I want you to do something you find interesting.