CMSC 122: Project 3

HTML, CSS & JavaScript

Overview

In this project, you will use HTML, CSS, and JavaScript to create a small webpage that plays an interactive game with the user. The rule of play is roughly:

- The game consists in a series of questions where the user is given the name of a University and is asked to choose its matching Mascot from a dropdown list of Mascot names.
- The user is awarded one point for each correct response, loses one point for each incorrect response, and is awarded zero points for choosing "Don't know" or just not choosing any of the options in the dropdown list before proceeding to the next prompt.
- At any point during the play, the user may "exit" the game (by clicking on the Exit Button) where the score is shown.
- After four questions, the game will be exited automatically; again, the score is then shown.
- If the user "reloads" the page, the game starts afresh.

Things to note:

It is quite possible that the user could obtain a minimum score of -4 or a maximum score of 4. Choosing "Don't know" (or not choosing any option before proceeding to the next question) results in no points being added or deducted, and so a score of 0 points is also possible. Again: at any point during the play of the game, the user may hit the exit button which should result in the game ending (by changing what is displayed on the screen) and the score being shown.

A substantial part of your grade is based upon how well you use JavaScript, CSS, and HTML to create a "dynamic" web-page, i.e., a HTML document whose content is generated or modified based upon user's actions.

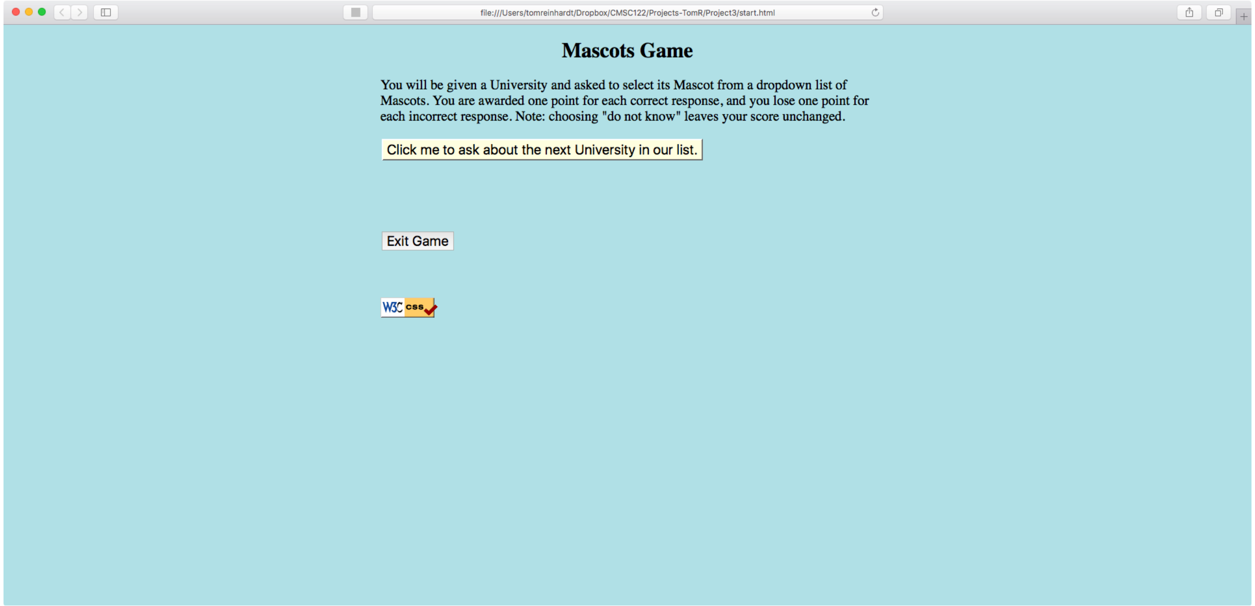
It is NOT acceptable to have a number of webpages, such as p1.html, p2.html, etc. one for each possible set of user interactions.

Instead, you must use JavaScript to dynamically generate the appropriate content that reflects users' interactions at that point in time.

The following set of screen shots illustrate several possibilities.

Screen 1 (The Initial Screen)

The game begins with this screen. Note, you are free to choose you own CSS styles, but your text and game play must be clearly readable or you will suffer massive point deductions.


file:///Users/tomreinhardt/Dropbox/CMSC122/Projects-TomR/Project3/start.html
Click me to ask about the next University in our list.
Exit Game
W3C css

Mascots Game

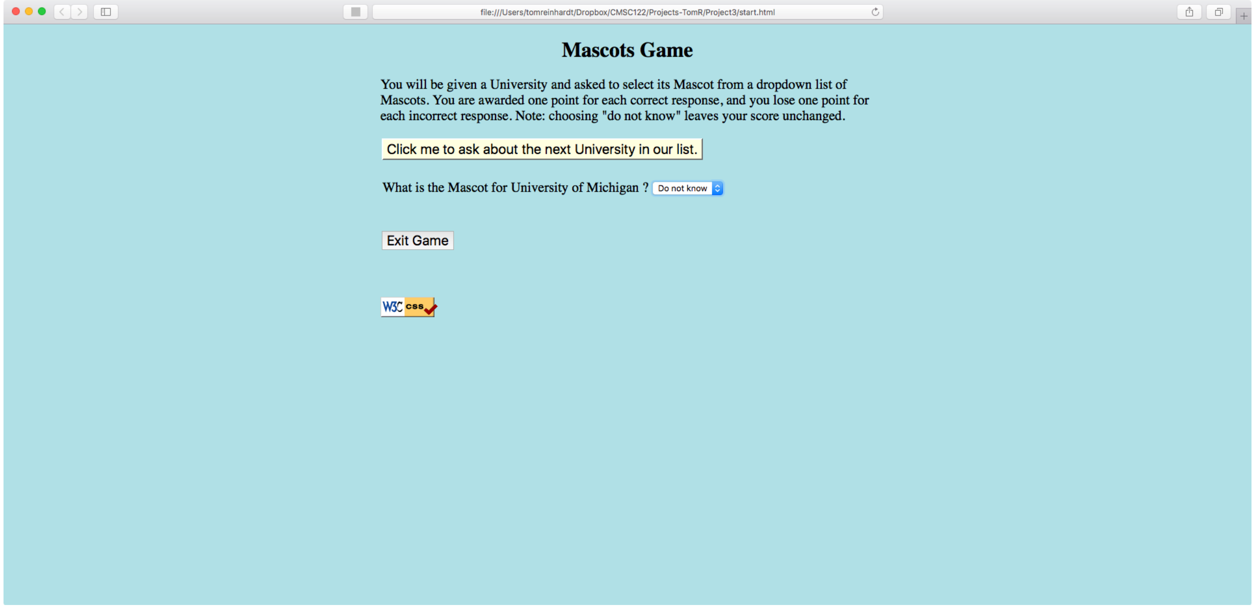
You will be given a University and asked to select its Mascot from a dropdown list of Mascots. You are awarded one point for each correct response, and you lose one point for each incorrect response. Note: choosing "do not know" leaves your score unchanged.

Moreover, all HTML and CSS must validate (HTML to version 5 and CSS to version 3). In my sample solution, I include the CSS validation information at the bottom of my webpage, but you do not need to do this; the Graders will submit your CSS file for validation.

Next, we present a series of “interaction screens.” Again: every interaction basically reuses the original webpage by executing some JavaScript. I use a combination of Class and ID element maps in conjunction with the necessary “event” handlers, such as when the user selects a Mascot name from the drop down box, or when they click on the Exit button.

The bulk of your logic will consist in managing the dropdown list of Mascot Names, evaluating the user’s response, and scoring. Perhaps the most difficult of these will be managing the dropdown box.

Consider the following problem: What happens if the user chooses nothing from the dropdown box but continues to click through to the next question? Clearly, the score should remain unchanged, but the “number of attempts” that appears in the final score report needs to indicate the total number of questions seen before either exiting the game or exhausting the total number of possible questions. Think about how you might do this!


file:///Users/tomreinhardt/Dropbox/CMSC122/Projects-TomR/Project3/start.html

Mascots Game

You will be given a University and asked to select its Mascot from a dropdown list of Mascots. You are awarded one point for each correct response, and you lose one point for each incorrect response. Note: choosing "do not know" leaves your score unchanged.

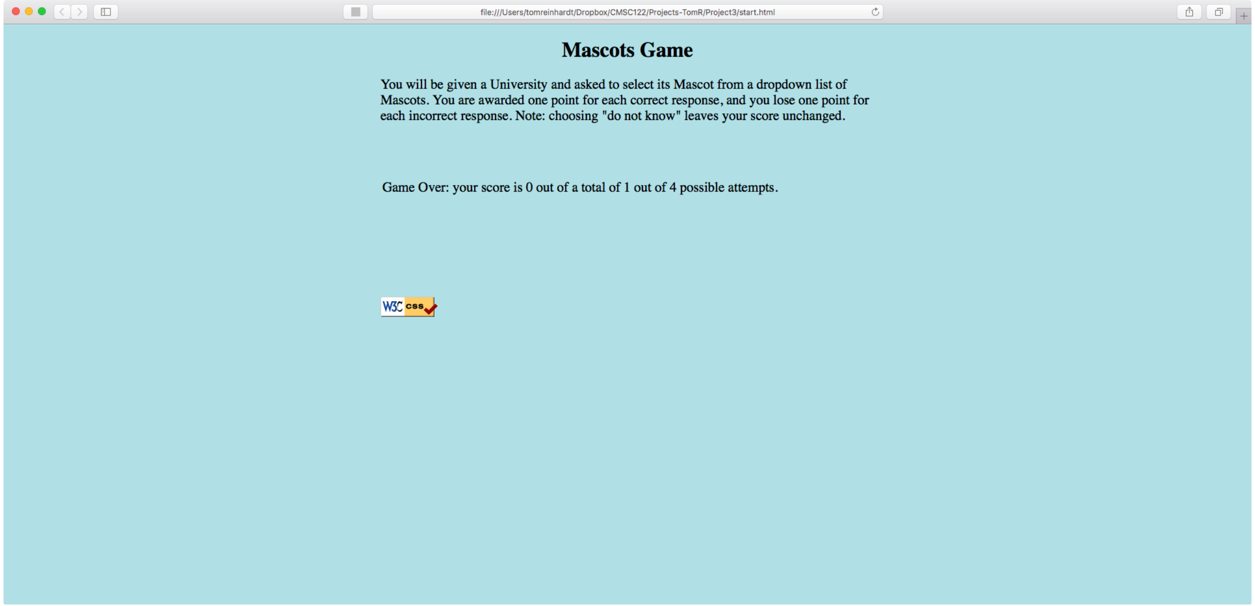
Click me to ask about the next University in our list.

What is the Mascot for University of Michigan ? Do not know

Interaction Screen(s)

Here is the “initial prompt.” Note that your dropdown box must allow for the user to state that they “Don’t know”—in which case no points are deducted for the question.

If the user clicked “Exit” at this moment, they would be taken to the “reporting” display, which is illustrated by the next screenshot:

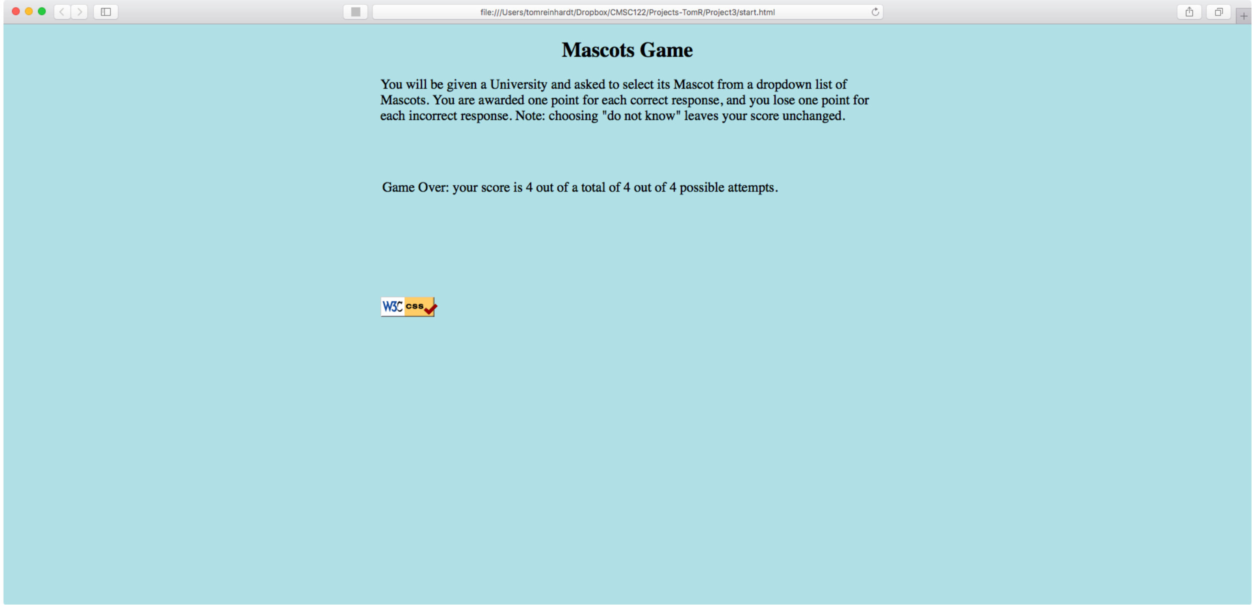

file:///Users/tomreinhardt/Dropbox/CMSC122/Projects-TomR/Project3/start.html

Mascots Game

You will be given a University and asked to select its Mascot from a dropdown list of Mascots. You are awarded one point for each correct response, and you lose one point for each incorrect response. Note: choosing "do not know" leaves your score unchanged.

Game Over: your score is 0 out of a total of 1 out of 4 possible attempts.

Suppose that the user correctly completed all of the questions (which, in this implementation means that they responded to all four Universities.) Then you might display a screen that reads as


file:///Users/tomreinhardt/Dropbox/CMSC122/Projects-TomR/Project3/start.html
W3C css

Mascots Game

You will be given a University and asked to select its Mascot from a dropdown list of Mascots. You are awarded one point for each correct response, and you lose one point for each incorrect response. Note: choosing "do not know" leaves your score unchanged.

Game Over: your score is 4 out of a total of 4 out of 4 possible attempts.

Notice: This shows that the user obtained their score by answering all of the available questions as opposed to skipping to the end.

Important things to note at this point …

- The game ends when either
  - All questions have been presented, or
  - The user clicks on the “Exit” button.
- At the end of the game, the prompt button, as well as the question and dropdown box are no longer visible, meaning that the game is over. Reloading this page at this time restarts the game.
- The “score” must include
  - The number of cumulative points (determined by the number of correct responses - number of wrong answers);
  - The number of questions answered (recall that the user may exit at any time), and
  - The total number of possible questions. In this assignment, you should provide at least the following four:
    - University of Wisconsin —Badgers
    - University of Nebraska —Cornhuskers
    - University of Michigan —Wolverines
    - University of Oklahoma —Sooners

Special Note

Later in this course, you will re-visit this game but with some “improvements” requested by the customer, such as the addition of a dynamic scoring table. This is done to provide you with a “authentic” coding experience. It’s rare that you ever write something “once.” If your implementation is successful, the customer always wants another, but with more “features.” If your implementation is less than successful, then you are either out of work or you are tasked with revising (fixing) the product but within a narrow timeframe. In either eventuality, I strongly suggest that you document all files with this in mind.

Implementation details

In addition to the details presented above, make sure that you pay attention to the following:

- Your CSS file must be separate from your HTML file.
- Your JavaScript file may be separate or may be included within the HTML file. Note, if you choose to include the JavaScript within the HTML file, make sure that the HTML validates to version 5.
- All files must be adequately documented.
- Again, only one HTML page is acceptable for this assignment. All changes should be reflected on that page by a combination of JavaScript interactions.
- Your HTML page should use at least the following HTML elements: headings, page title, <div>, <span>, <input>, <select>, <option>, most likely a simple <table>, including <th>, <tr>, <td>, <thead> and <tbody> tags. You will also need at least one <button> tag to handle game exits.

Submission details

As always, all files must be zipped into one directory named per the class convention: `YourNameP3.zip`, where YourName is replaced with `LastNameFirstName`. For example, if your name is `John Smith` then `SmithJohnP3.zip` is the expected submission name.