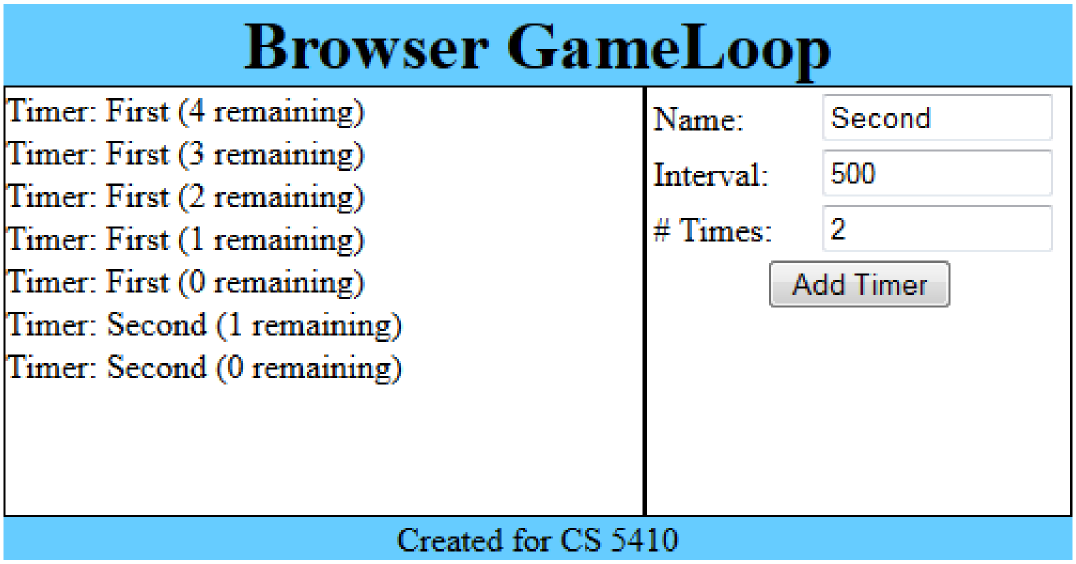
Introduction

You are to write a basic Game Loop. The purpose of this assignment is to help introduce the structure of the game loop and help gain familiarity with the JavaScript language and some basic HTML(for those who don't yet know it). This application provides a simple user interface that allows users to create timer events that are reported a specified intervals for some number of times.

Assignment

Write a program that presents the user with a simple interface, as show in the image below:



Browser GameLoop

```
Timer: First (4 remaining)
Timer: First (3 remaining)
Timer: First (2 remaining)
Timer: First (1 remaining)
Timer: First (0 remaining)
Timer: Second (1 remaining)
Timer: Second (0 remaining)
```

Name: Second

Interval: 500

Times: 2

Created for CS 5410

An event has a name, the interval at which it is reported, and how many times it is reported. The user will enter these details using the controls on the right; the timing interval is in milliseconds. Any number of timers can be active at one time, not just one at a time as shown in the sample above.

The following outlines specific details you must follow for this program...

- The user interface should look substantially similar to the one above. It should have a header and footer, along with an output pane on the left and the event creation controls on the right. Feel free to make it look nicer.
  - The main html file must be named: index.html
  - The page must use an external CSS file for styling, name it whatever you wish.
- Any number of events may be active.
- When an event should be reported, note this to the output window using the format in the example image.
- Submit project in a .zip file (and only a .zip file).

Technical Requirements

I'm allowing a certain amount of hacky looking code to come in for this assignment, as I expect many of you are writing JavaScript/HTML code for the first time. This assignment is mostly to help everyone get a feel for how to do things in a web-based environment, I don't need the code to be perfect, just need you to get on your feet with something. With that said, there are some technical requirements I want followed.

- Use a single JavaScript code file, name it anything you like.
- All JavaScript code must be inside an object/namespace named `Game1`. e.g., `var Game1 = {};`
- Create a function named `gameLoop` in which you update the elapsed time since the last time the function was called.
  - The function will make a call to another function named `update`, which accepts the elapsed time as a parameter. It is in the `update` function where any active timers are updated.
  - The function will make a call to another function named `render`. It is in the `render` function where any timers that need reporting are displayed.
  - This function must use `requestAnimationFrame` to invoke the `gameLoop` function again.
- Can not use `setTimeout`, `setInterval` or anything that causes asynchronous timer callbacks into your code. You have to track how much time has elapsed on each active timer, and use that to know when to report or expire during the update part of your game loop.
- When an event expires (# of times exhausted), it must be removed from the internal data structure(s) holding it.

Relevant JavaScript and HTML Bits & Pieces

The following are some bits of knowledge that may help you put this program together. I'm not telling you exactly how to use them, this is a 5000 level course, I expect you to be able to look at and understand documentation.

- `requestAnimationFrame` invokes the callback function with a `DOMHighResTimeStamp` as a parameter.
- You can use `performance.now()` to obtain the a current high resolution time stamp, for example, say, in the initialization of your code.
- Use the following technique to "output" something to an element...

  ```
  var node = document.getElementById('div-console');
  node.innerHTML += "Hi Mom!";
  ```

- Might want to look at `node.scrollTop` and `node.scrollHeight` to keep the latest timer update visible.
- Consider using the `button` or `input` elements to create the button.