Setting Up

Log in to your UNIX account. Open up the `public_html` folder, and then the `cmpsci120` folder, as you normally do. If you list files with the `ls -al` command, you should see all the previous work that you have done, but we will not mess with those files. This is a new thing.

Part 1

Use the emacs text editor to create a new file called `CamWatcher.html` (with the exact spelling and capitalization as shown). Type in the HTML document that you see on page 245 of the Companion (JavaScript Camera Watcher). Enter any of the four addresses at the bottom of page 245 in the blank for `MyURL` (they all work as of 11/23/2018) and save the page. Put your name in a comment at the top of the document.

Make certain that the permissions on `CamWatcher.html` are `rw-r--r--` as usual.

Test your page. The URL to this file should be as follows, with your username in the blank (this is the address we will use for testing and grading):

http://elsrv3.cs.umass.edu/~_____/cmpsci120/CamWatcher.html

Correct any errors in emacs and reload the page, as required. Once you have "finished" the page, watch the camera for several minutes to make sure that it is updating. Bear in mind that the four addresses given in the book are in Wyoming, Idaho, Oregon, and Germany, respectively, so if you see a black screen it might be because it is nighttime at the location of the camera. Some of these cameras update only every few minutes, so it might be a while before you notice a change, even though the CamWatcher is pulling the image from the remote server every ten seconds.

Part 2

This is the hard part! Go find a camera on the Web that updates a graphics file periodically (typically a `.jpg` or `.png` file). It cannot be any of the four addresses in the book, and it cannot be any of the WebCams I have on my personal Web site. I have found that traffic cameras are often the best candidates for this kind of program. Note that some camera URLs have a built-in expiration date, and while they work at the time of testing and submission they might not work when we test them; check the URL of the camera to see if it contains anything that might indicate this is a possibility.

What to Turn In

When your camera watching page is visible on the Web and works correctly, send an email to the `literacy@cs.umass.edu` account as usual. The subject line must be set to the exact phrase **CMPSCI 120 ASSIGNMENT #5** and the body of the message must contain your name, your username, and the full Web address to `CamWatcher.html`.

Grading

Points will be removed for the following items:

A.   Non-Working Code. Code must load some image and update it every 10 seconds (even if the image does not change as a result of the update).

B.   Barely working code. Code *might* work if certain items are changed, but graders will be looking at the source to see if all pieces are present and there are no errors reported by the browser.

C.   Invalid camera. Camera must be valid, and not one of my examples or items on my Web page. Note that some camera URLs have a built-in expiration date, and while they work at the time of testing and submission they might not work when we test them.

D.   Bad Indentation. The indentation in the book is two spaces per level in order to get the text to fit on the page. Indentation of 2 or 4 spaces per level is allowed. Note that some browsers in View-Source mess up the indentation; as long as it is "reasonable" and not all starting in column 1 it will be accepted.

E.   Student name missing. The student's name must be inside a `<!--` and `-->` comment at the top of the document.