

CS 1100: Web Development: Client-Side Coding / Fall 2016

Assignment 8: Doing Research on the Web

Due: Saturday, November 5, 2016, by 11:59 p.m.

Assignment Goals

The purpose of this assignment is to practice loops and arrays; conditional statements, and string concatenation, while continuing to explore the DOM API. The context of this assignment is basic information extraction techniques, specifically, web page scraping.

Assignment Description

Write a program that does the following:

1. Repeatedly asks the user to provide a CSS selector, finds all the elements on the page that match the provided selector, and then **writes out to the console** how many elements the page contains that match this selector. Specifically, your console messages should include (1) the number of elements found, and (2) the selector provided by the user. **Here's the exact format you should use:**

```
22 elements have been found that match the selector 'p'  
1 elements have been found that match the selector 'h1'  
106 elements have been found that match the selector 'a'
```

2. Once the user types in "quit" (no quotes) into the prompt, the program uses the `document.write` method to write out to the web page the last selected set of elements, **each element on a new line**. Your program should use the `innerText` property for all elements, except anchor elements (i.e., hyperlinks). For anchor elements, you should display the element's text together with the value of its href attribute, **making that value an active link**.

Example. consider a page that contains this content:

Hello, World!

This is some text with [a link to the UNI homepage](#)

This is some other text with

And this is a link to the UNI [Computer Science department](#).

If the last selection was "p" (before the user typed in "quit"), your program should produce this text:

```
Hello, World!
```

```
This is some text with a link to the UNI homepage
```

```
This is some other text with
```

```
And this is a link to the UNI Computer Science department.
```

However, if the last selection was "a", your program should produce this text (**you must use this format**):

```
the text 'a link to the UNI homepage' links to http://uni.edu  
the text 'Computer Science department' links to http://cs.uni.edu
```

You can test your program on any web page by simply copying and pasting it into the console.

NOTES

- To get the tag name of an element, use the `tagName` property
- If you are testing your program using the console, the `document.write()` method will **replace** the content displayed by the web page. If you are testing your program by referencing it from your test web page (i.e., using the `<script>` tag), it will **append** your text to the content displayed on that web page. Both approaches are correct.
- Use variables generously, especially with long string concatenation. For example:
 - Instead of this:

```
document.write("some text" + some-complex-method-call + "more text");
```
 - Use this:

```
var foo = some-complex-method-call;  
var text = "some text " + foo + " more text"; (note the spaces)  
document.write(text);
```

Submit your work

Submit your **hw9.js** file to eLearning.

Grading

This assignment is worth **30 points** (which accounts for approximately **2.7% of your grade**).

1. Part 1. Your program:
 1. Repeatedly asks the user for a selector (**2 points**)
 2. Finds the elements using provided selector (**2 points**)
 3. Console message is correct (**2 points**)
 4. Stops when the user types in "quit" (**5 points**)
2. Part 2. Your program:
 1. Writes out correct set of elements to the page (**5 points**)
 2. Uses correct format for each element using `innerText` (**5 points**)
 3. Uses correct format for anchor elements (**5 points**)
3. Use of good variable names (**2 points**)
4. Code is indented correctly (**2 points**)

To earn partial credit, your program should work without causing JavaScript errors.