

Advanced Web Development and Web Scraping
Spring 2020
Assignment #5 – Chrome Browser Extension

Note: To turn in this assignment, create one or more zip files of your extensions and e-mail them to me.

1. Create a Chrome browser extension that reformats Eastern's web pages as follows. The extension
 - a. automatically runs on Eastern's pages by using a content script that has match pattern of <https://www.easternct.edu/>*
 - b. changes the color of the menu bar, as in the class example
 - c. hides the image below the menu bar (you will need to hide the second <section> tag)
 - d. displays a *div* with useful links, as in the class example (you may add/change the links)
 - e. may include additional features or style modifications
2. Create a Chrome browser extension that does one of the following when the user clicks on the extension icon:
 - a. changes either all images on the page to an image of your choice (by changing the *src* attribute)
 - b. changes the background image of the page by using JavaScript to change the background-image property (for more information, see https://www.w3schools.com/cssref/pr_background-image.asp).
 - c. Rickrolls a user by changing the *href* of all links on the page (you may optionally limit this to YouTube links). For more information, see this [link](#).

Note that to include images in your extension, you need to save the image in your extension folder (by convention, you should save the image in a folder named *images*). To access the image from your extension, get its location by including the following JavaScript code (in my case the image is stored in *images/picture.jpg*)

```
var imgURL = chrome.extension.getURL("images/picture.jpg");
```

The location will include the ID of your extension. In addition, the image must be listed in your manifest, using

```
"web_accessible_resources": [  
    "images/picture.jpg"  
],
```

3. Modify the highlighter extension as follows:

- a. Allow the user to enter multiple search terms (up to 3), by including a button that lets the user add additional search terms. When this button is clicked, a new input box is displayed and the user can enter an additional search term. However, this button is *disabled* if one or more of the textboxes are empty. If textboxes 2-3 lose focus and are empty, then the textbox is removed (hidden).
- b. The *highlight.js* script finds and highlights multiple keywords. The easiest way to do this is to create a regular expression with all search terms separated by the OR symbol ('|'). In this case, *qry* should have the following form when the regular expression is created: "word1|word2|another word"
- c. In addition to highlighting the words, a summary is displayed on the top right corner of the screen, listing each keyword and the number of matches found.