

Advanced Web Development and Web Scraping
Spring 2020
Assignment #8 – Selenium Assignment

1. Using Selenium, search for a movie on IMDB (<https://www.imdb.com/>). “Click” on the first result to go to the movie’s page. Then extract its title (including the year) and its rating, and print out the results in the format *Ghostbusters (1984) has a rating of 7.8*. Note: sleep for 2 seconds after searching for the movie, and sleep for 1 second after clicking on the first result (this will help ensure that the page is fully loaded before continuing).
2. Complete the Number Cruncher web page and the corresponding unit testing script by following the instructions below:
 - a. For the Number Cruncher web page,
 - i. Clicking on “Add Numbers” calculates the sum of the numbers and displays the answer in *green* using the format *The sum of the numbers is 10* (this is already done)
 - ii. If the input is invalid, the error message is displayed in *red*.
 - iii. If the user clicks the ‘Reset’ button, the text box is cleared (set to “)
 - b. For unit testing, complete the script to test that the *Add Numbers* buttons works correctly for both valid and invalid numbers, and that the *Reset* button also works.

Note: In Selenium, the preferred way to get the style (CSS value) of an *element* is to use, .e.g. `element.value_of_css_property('color')`. This returns the computed value, which for colors is a string in *rgb* format. In some cases you could use `element.get_attribute('style')` but this returns only the *inline style* of the element. You can find *rgb* color codes at the following website (search for “List of common”):

<https://htmlcolorcodes.com/>