Create a brand new document using the naming convention HW**X-MUid.docx** (example: HW10-johnsok9.docx).

**Objective**: The objective of this exercise is to:
1. Continue working with Ajax
2. Understand external web API documentation
3. Write web clients using JavaScript/jQuery/Ajax

**Submit**: screenshots, your word document, html files and links to your web page(s)

You **may** discuss the concepts with your fellow students
You **may not** show or share code with your fellow students
You **may not** show or share code with internet sources
You **may** discuss this with your instructor or TA.

## Part #1: Create GitLab project
*Estimated time: 10 minutes*
### Exercise:
- Gitlab
  - Make a new hw10 directory in your cse383 projects directory.

## Part #2: Complete the Lab (if not already)
*Estimated time: 1 hour*
### Exercise:

Use your completed lab as a starting point (copy it into the hw10 directory) and update the file info.js that will make appropriate ajax calls to the rest api and update the display.

Note: if your lab worked totally – then copy it and no other work is required for this section

- Add JavaScript/jQuery/ajax code so that the remaining (Process and network usage) boxes are updated with the appropriate information.
  - On the left is a process list.
  - In the middle is a box listing data from the load function.
  - On the right is a box listing current network usage in bytes per second
- Add code to update the time.

- When the ajax call fails, record the event in the log, newest event at the top.
    - note: 10% of the time the calls will fail so that you have something to log

## Part #3: Calculator

*Estimated time: 1-2 hours*

## Exercise:

Create a good-looking bootstrap form with two fields for number (floating point) entry and four buttons (+ - / *).

**When the operator button is pressed, call the ajax api to do the math and display the result.**

**This is an example for the Add operation.  Notice the word Add right before the ? and then the values of n1 and n2.**

https://api.clearllc.com/api/v2/math/Add?
api_key=bed859b37ac6f1dd59387829a18db84c22ac99c09ee0f5fb99cb708364858818&
n1=4&n2=5

Operations: Add, Subtract, Multiply, Divide

Attach screen shots showing multiple calls with the correct answer, and one showing you handled the return of an invalid call (divide by zero)

   i. Process
     1. Call cloud math server
     2. Base url https://api.clearllc.com/api/v2/math/
     3. Commands (case insensitive)
       a. Add
       b. Subtract
       c. Multiply
       d. Divide
     4. Arguments (case insensitive)
       a. N1
       b. N2

   c. Required api_key bed859b37ac6f1dd59387829a18db84c22ac99c09ee0f5fb99cb708364858818

  5. Example Call (note this is only one line, it wrapped in word) https://api.clearllc.com/api/v2/math/Add?api_key=bed859b37ac6f1dd59387829a18db84c22ac99c09ee0f5fb99cb708364858818&n1=4&n2=5

  6. Returns

   a. Result (if good)

    {"result":9}

   b. Error (if bad request)

    {"error":{"code":400,"context":null,"message":"Invalid or missing resource name.","status_code":400}}

   c. Math Error (divide by zero for example)

    {"error":{"code":400,"context":null,"message":"Divide by zero error.","status_code":400}}

## Part #4: Write New Word Document

*Estimated time: 15 minutes*

### Exercise:

- Always make sure it includes your name, class, assignment etc.

- Explain in complete sentences (relating to this assignment):
  - what worked
  - what didn't
  - how long it took
  - What you liked
  - What you didn't

## SUBMIT:

- Word Document
- Screenshots inside word document
  - Attach a screenshot showing GitLab with changes submitted
  - Attach a screenshot of your fully functional lab web page (including output)
  - Attach screenshots showing all four math operations
- Files
  - Attach the html, optional CSS, and JavaScript files
- Paste the URL of your pages into the submission as a comment