

# ICT-4570 Homework 6

## Purpose

This exercise allows you to experience AJAX using JavaScript. In it you will asynchronously access a JSON resource, parse the return, and filter and display data as a table.

## What to Hand In

Canvas submission instructions:  
Please combine multiple files into a single "zip" archive, and save it in a location that you will remember. When you are ready to submit the assignment solution, open the assignment in Canvas, and click on the Submit for Evaluation button at the top, attach the file, and click Submit for Evaluation at the bottom.

## President by AJAX

Name:  Took Office:

Number	Name	Birthday	Took Office	Left Office
1	George Washington	1732-02-22	1789-04-30	1797-03-04
22	Grover Cleveland	1837-03-18	1885-03-04	1889-03-04
24	Grover Cleveland	1837-03-18	1893-03-04	1897-03-04
38	Gerald Ford	1913-07-14	1974-08-09	1977-01-20
41	George H. W. Bush	1924-06-12	1989-01-20	1993-01-20
43	George W. Bush	1946-07-06	2001-01-20	2009-01-20

## Problems

- I. Retrieve the JSON resources at the provided URL, extract the ones that match the criteria provided, and display them as a table on your web page.
  1. Create a basic web page from the course template (which we have been using all along), and external JavaScript and CSS files
  2. Add an input to the page which will be used to match the President's name.
  3. Add an output element on the page suitable for receiving the table result you will build.
  4. Examine the data source at  
`http://schwartzcomputer.com/ICT4570/Resources/USPresidents.json`.  
The structure will be needed to understand how to extract data from it later.
  5. Set up a `XMLHttpRequest` in a method to retrieve this resource
  6. Set the timeout on the request for 30 seconds (30000 ms). This should be sufficient time to retrieve the resource, and will give you an indication of a typo or other problem in doing so.
  7. Create a function to handle the data once it comes back in the asynchronous request.
  8. Ensure the function does the following, using other functions as appropriate:
    - a. Parse the result using `JSON.parse` and assigning it to a local variable.
    - b. The function should create a table with the number, name, birthday, entered office, and left office headers
    - c. The function should add all President information which passes the filter to the table. If the filter is blank or can't be interpreted, show all the Presidents.
- II. Stretch goal: Create inputs for number, name, and birthday, and filter on each, as above.
- III. Stretch goal: Build a "subtable" of vice-presidents in a cell on each displayed President's row. Give it a heading of vice president, and include the vice president name (or names) for the subtable rows.

## Notes

- I recommend that you create a simple function first to process the return value, just to make sure you actually received it. After that, you can spend some more time refining the function to do precisely what you'd like.
- Start with the "empty" filter that allows all Presidents to be displayed; then filter by name.

- Question for thought: Why does this work at all? How can you have your local web page access a remote resource?
- The section on *Building a Table* at the end of the Document slides may prove useful in actually building the data table.

## Evaluation

Criteria	Weight
Basic HTML5 template HTML file with suitable content	15%
Basic HTML5 template CSS file with suitable content	15%
XMLHttpRequest set up properly	15%
XMLHttpRequest response handler set up properly	15%
Table elements built properly	15%
Proper elements are displayed after a query	10%
Screen output provided	15%
Stretch goal: Multiple filters	+10%
Stretch goal: Vice Presidents	+10%