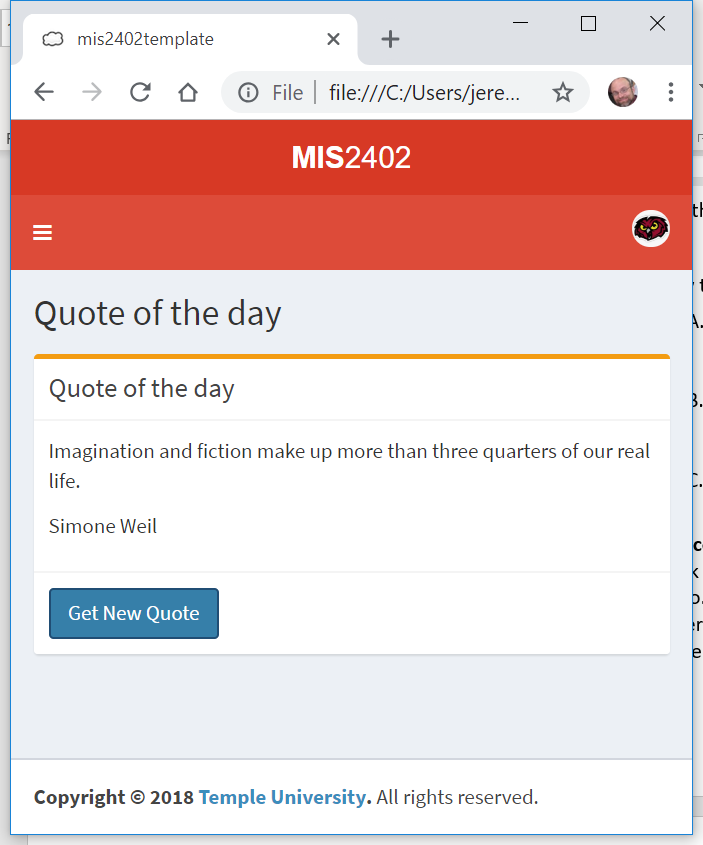
# MIS2402 Assignment – AJAX 1

## Instructions (follow along in class)

1. Please note - this assignment requires you to be working at a computer with a working internet connection.
2. Download the mis2402template02.zip and set up the mis2402template02 folder in your workspace.
3. In the mis2402 application template, find the navigation section. Note the existence of the wrapper div tags referenced there.
4. Inspect app.js. Notice again that *some* of the functionality has been removed from the jQuery ready handler and placed in the “supporting functions” section at the top of app.js. (See the functions navigationControl.) This change allows us to keep the “ready” logic as simple as possible. It also creates the opportunity to reuse functionality written in the “supporting functions” section.
5. Visit this URL - <https://misdemo.temple.edu/dailyquote/> . This resource returns a simple JSON object of length 1. Every time you refresh the page you get a new (randomly selected) quote. Notice that what is being returned is **not** html. It is JSON. It is a structured text file of daily quote data being delivered over the internet.
6. IMPORTANT NOTE – This resource just happens to be hosted from misdemo.temple.edu – it really could be coming from any provider who cared to offer it … either here on campus, or on the other side of the world. This page is providing data to our application. It is a very simple form of Web API. For some simple examples of a web API that could be used for practical purposes see: <http://goqr.me/api/> or <https://restcountries.eu/rest/v2/all> or <https://misdemo.temple.edu/states/> .
7. Now, go back to app.js and look at the quote logic. This code runs when the document is ready, and it retrieves a simple object (think “array”) from the <https://misdemo.temple.edu/dailyquote/> resource. The daily quote resource changes every time you reference it. The dailyquote resource could possibly return many quotes, but it is currently set to return a simple object of length one. Notice that we get to choose what the data being returned to the callback function will be called. Here, we chose the name “a\_gaggle\_of\_quotes” … it is an intentionally silly name. The point is that this is a simple object that could hold many things… a lot, a gaggle, a swarm, a population… many things.
8. Improve the content of our quote wrapper div so that the author name is displayed along with the quote as well. Your results should look like this:

|  |  |
| --- | --- |
|  | How to do it …   1. Look at the JSON resource first, so you know the name of the author attribute. 2. Look at your html in the quotes wrapper, make a <p> tag to hold the author. Give it a meaningful id. 3. Add code to your existing getJSON callback function to populate the new tag.   **Notice** that steps A,B,C require you to think holistically about what you are trying to do. Where does the data come from? Where do I want it to go? How will I get it there? |

1. Now, we will make a “Get New Quote” button. The button should initiate a call to the getJSON method and put a new quote and author in to place.



To do this we will need to:

* 1. Create a button in HTML and give it an id.
  2. create a click() event handler for the button.
  3. Execute the exact same logic as found in the last step.
  4. DISCUSS: Should you rewrite the $.getJSON statement again? Or should you choose to create a supporting function? Which approach is better?

**CONTINUED…**

**On your own**

1. Find “on\_your\_own\_json.zip”. Using the URLs previously mentioned in this document, can you write a loop that retrieves all 50 states and appends them to a table on the page?
2. With “on\_your\_own\_json2.zip” write code that retrieves the name of one country, the country with largest population.

Here are some hints:

* 1. You will need to loop through all the countries in order to find the most populated country.
  2. Inside the code block of the “for” loop, create one variable to hold the name of the country and another to hold the population.
  3. Use a conditional statement inside the loop.

**Upload your work**

This assignment requires you to upload three separate folders. The folders mis2402template02, on\_your\_own\_json, on\_your\_own\_json2 and all their respective contents should all be found in your wwwroot folder on the class server.

Upload your work to the class server on or before the due date / time. This assignment will be graded.

**How will this assignment be graded?**

|  |  |
| --- | --- |
| Item | Point Value |
| All files / folders uploaded correctly | 10 |
| mis2402template02 Get Quote works | 30 |
| on\_your\_own\_json All 50 states retrieved and displayed. | 15 |
| on\_your\_own\_json HTML table rendered correctly | 15 |
| on\_your\_own\_json2  Solution must use a loop and a conditional statement to determine the answer. | 30 |