I first started by modifying the “ajax” function the “scriptures.js” file. I removed the instance of using the “XMLHttpRequest” and replaced with the code snippet provided in the presentation from day 10, with some modifications. I kept the same parameters that the original “ajax” function had, since those all could still be used in using the fetch API. However, at first the only parameter I used was the “url” parameter. While I would get a response from the URL none of the volumes or books would display on the web page. After browsing the internet trying to trouble shoot my issue, I found this link (https://stackoverflow.com/questions/37663674/using-fetch-api-to-access-json) that made me realize I needed a second .then to use the retrieved JSON data. After adding another .then that called the data, I was able to see the volumes and books. However, the chapter contents still wouldn’t display. The console logged an error that said the json data was seeing an unexpected “<”. That’s when I realized I didn’t have anything in my fetch request to account not only JSON but for HTML as well. Luckily, the MDN site mentioned that “response.text()” can be utilized to read text data. I added another condition that would check if the “skipJsonParse” was true or not, which would determine if the fetch should return “response.json()” or “response.text()”. The full code can be seen below.

```
// https://stackoverflow.com/questions/37663674/using-fetch-api-to-access-json
    ajax = function (url, successCallback, failureCallback, skipJsonParse) {
        fetch(url)
        .then(function(response) {
            if (response.ok && !skipJsonParse) {
                return response.json();
            } else if (response.ok && skipJsonParse) {
                return response.text();
            }

            throw new Error("Network response was not okay.");
        })
        .then(function(data) {
            successCallback(data);
        })
        .catch(function(error) {
            failureCallback(error);
        });
    };
```