

Name: Andrew Parsons

TMU ID: 500992021

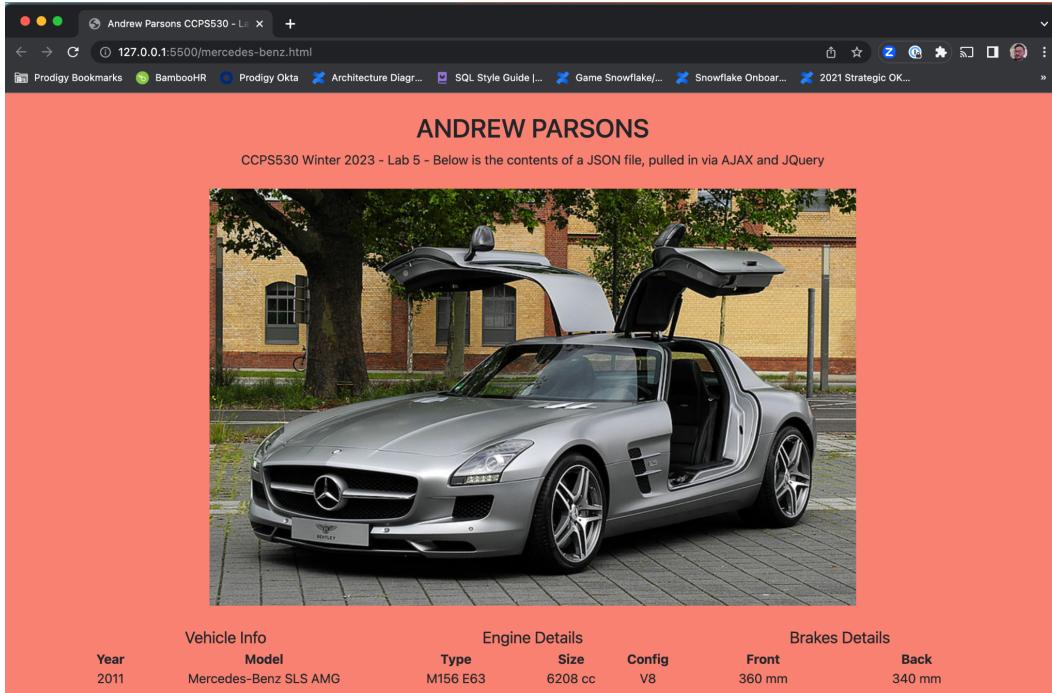
Due Date: Mar 18, 2023

Github Repo Link: https://github.com/Parsonswlu/parsonswlu.github.io/tree/main/CCPS530_Lab5

CCPS 530 - Web Systems Development - Lab 5

HTML Displayed on Browsers

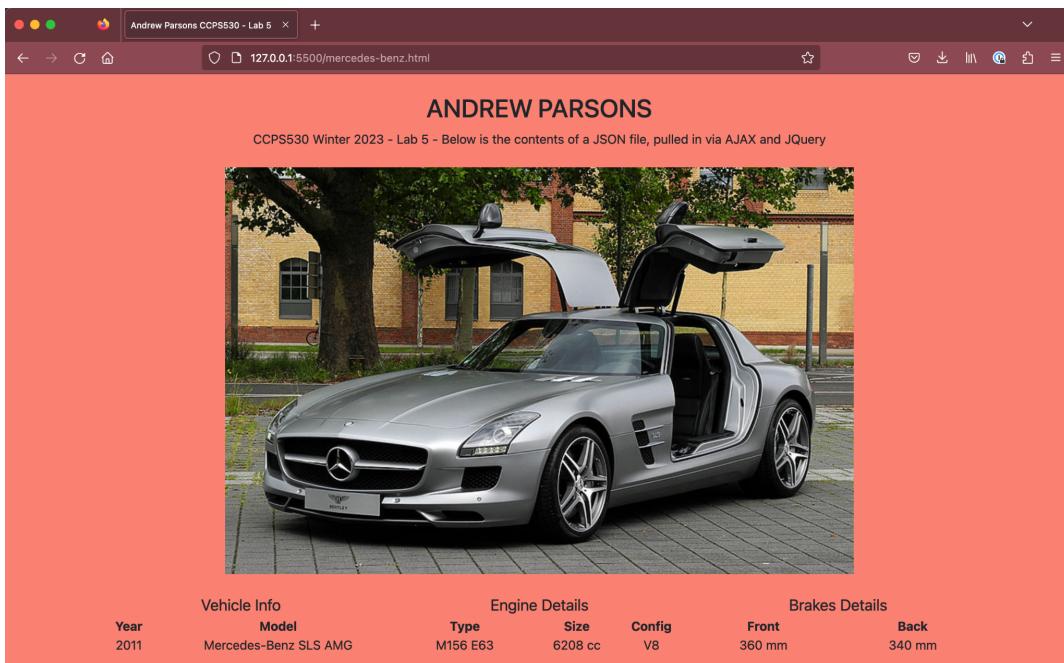
- Google Chrome:



The screenshot shows a Google Chrome browser window with the title "Andrew Parsons CCPS530 - Lab 5". The address bar displays "127.0.0.1:5500/mercedes-benz.html". The page content includes a header "ANDREW PARSONS" and a sub-header "CCPS530 Winter 2023 - Lab 5 - Below is the contents of a JSON file, pulled in via AJAX and JQuery". Below this is a large image of a silver Mercedes-Benz SLS AMG with its gull-wing doors open. At the bottom of the page is a table with vehicle details:

Vehicle Info		Engine Details			Brakes Details	
Year	Model	Type	Size	Config	Front	Back
2011	Mercedes-Benz SLS AMG	M156 E63	6208 cc	V8	360 mm	340 mm

- Firefox:



The screenshot shows a Firefox browser window with the title "Andrew Parsons CCPS530 - Lab 5". The address bar displays "127.0.0.1:5500/mercedes-benz.html". The page content is identical to the Chrome version, featuring the "ANDREW PARSONS" header, the JSON-substituted image of the silver Mercedes-Benz SLS AMG, and the same table of vehicle details at the bottom.

HTML Checker

- Validator Results from <https://validator.w3.org/>

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for mercedes-benz.html

Checker Input

Show source outline image report

Check by No file selected.
Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Document checking completed. No errors or warnings to show.

Source

```
1. <!DOCTYPE html>↵
2. <html lang="en">↵
3. <head>↵
4.   <meta charset="UTF-8">↵
5.   <meta http-equiv="X-UA-Compatible" content="IE=edge">↵
6.   <meta name="viewport" content="width=device-width, initial-scale=1.0">↵
7.   <title>Andrew Parsons CCP530 - Lab 5</title>↵
8.   ↵
9.   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.1.3/dist/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPM0" crossorigin="anonymous">↵
10.  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" integrity="sha384-wvFxpopZZVQGK6TAh5PVlG0fQNHSoD2xbE+QkPxCAF1NEevoEH3Sl0sibVc0QVnN" crossorigin="anonymous">↵
11. </head>↵
12. <body style="background-color: salmon;">↵
13.   ↵
14.   <div class="m-4 text-center">↵
15.     <h2 class="text-uppercase" id="profile">Andrew Parsons</h2>↵
16.     <p>CCP530 Winter 2023 - Lab 5 - Below is the contents of a JSON file, pulled in via AJAX and JQuery</p>↵
17.   </div>↵
```

CSS Checker

- Validator Results from <https://jigsaw.w3.org/css-validator/>

No CSS file utilized for this particular lab.

Technical Report

1. How did you consume and parse JSON?

- I used AJAX and a GET request with dataType: 'json' to do the initial loading of the JSON file, as can be seen in code below:

```
<script>
    $(document).ready(function() {
        $.ajax({
            url: 'ajax/lab5_data.json',
            type: 'GET',
            dataType:'json',
            success: function(data) {
                ...
            },
            error: function(e) {
                console.log(e.message);
            }
        })
    });
</script>
```

- Once the JSON data was extracted, I used JQuery combined with specific class and id values to populate this data into the appropriate areas within the HTML:

```
success: function(data) {
    // load image and relevant title/alt text
    $('.load-image').attr("src", data.imageURL);
    $('.load-image').attr("alt", "Image of a " + data.year + " " + data.name);
    $('.load-image').attr("title", data.year + " " + data.name);
    // populate vehicle details into various fields
    $('#vehicle-year').text(data.year);
    $('#vehicle-name').text(data.name);
    $('#engine-type').text(data.engine.type);
    $('#engine-size').text(data.engine.size);
    $('#engine-config').text(data.engine.configuration);
    $('#brakes-front').text(data.brakes.front);
    $('#brakes-back').text(data.brakes.back);
},
```

2. How long did you spend on this lab? The length of time includes readings and research and code experimentation. State time involved in readings and research as well as code experimentation sessions.

- I spent approximately 2.5 hours working on this lab.
- It took about 2 hours to set up the bootstrap elements and structure in order to load the data from a json file properly.
- It then took about another 30 minutes to do the validation, take screenshots, write up the technical report, etc.