HTML Tutorial for Computer Science Students: Junior Comps Tutorial Assignment 2

Charlie Rui Chai
rchai@oxy.edu
Occidental College

1 Introduction

For my Comps Project, I want to build a dining review platform for the Occidental College community. This goal includes developing a web application and running it on a server. This tutorial will guide the reader through the process of building a webpage for restaurant menus on their local server. Restaurant menus are the essential elements of a dining review platform and is one of our main functionalities. We use HTML, CSS and JavaScript as programming languages, and run the webpage on google Chrome using VSCode Live Server extension. This tutorial aims to teach our readers the basics of web development skills and allow them to modify and build their own. In the future, we will learn more advanced skills such as Ajax to asynchronously update webpage content, and the React library to develop more efficient webpages.

2 Methods

To get started, we first install the development environment. We download Visual Studio Code (VSCode) from its website https://code.visualstudio.com. The Live Server extension was installed from VSCode "extensions market" to enable real-time preview of the webpage during the coding process, which simplifies debugging and testing.

For the text and image content, we create a new HTML file and name it Menu.html. HTML is a markup language. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes, and other items. We build a simple but typical layout for the menu page. The HTML file includes a main menu section with div containers for each menu item with appropriate class attributes for CSS styling. Each item within the menu is represented by an article tag, including details such as the item's image, name, price, and description.

To enhance the visual appeal of the webpage, we employ CSS to design visual effects for our webpage. Cascading Style Sheets (CSS) is a style sheet language used for specifying the presentation and styling of a document written in a markup language such as HTML or XML. The CSS file we use (styles.css) is provided by the tutorial repository. It contains global and menu styles with predefined font sizes, colors, borders, and other style variables.

The JavaScript file handles the logic of our webpage and is thus the main focus of the tutorial. To allow interaction between JS and HTML/CSS, we should first understand the Document Object Model (DOM). The DOM is a programming interface for web documents. The DOM represents the document as nodes and objects; that way, programming languages can interact with the page. In our tutorial, an example is the DOM specifies two method in this code snippet, which returns the first or all element within the document that matches the specified selector.

Listing 1: Document method example

```
const sectionCenter = document.querySelector(".
    section-center");
const filterBtns = document.querySelectorAll(".
    filter-btn");
```

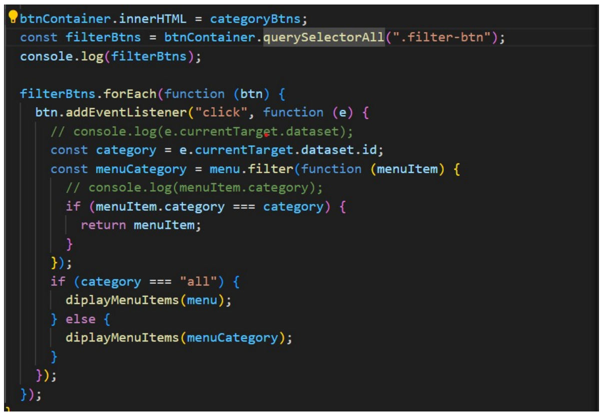
We also need to learn some useful Web APIs. To realize the buttons' functionalities, we use the event listener to "listen" to events like users' click on the button and do something after that. The addEventListener() method of the EventTarget interface sets up a function that will be called whenever the specified event is delivered to the target.

Listing 2: Listeners example

```
filterBtns.forEach(function (btn) {
  btn.addEventListener("click", function (e) {
    console.log(e.currentTarget.dataset);
    const category = e.currentTarget.dataset.id;
  })
})
```

We also need to learn to use functions, the fundamental building blocks in JavaScript. For students with Java experiences, function in JS is similar to function in Java. You must define it somewhere in the scope from which you wish to call it, and it will take some input and return an output. You don't necessarily need to name the function or pass in an input.

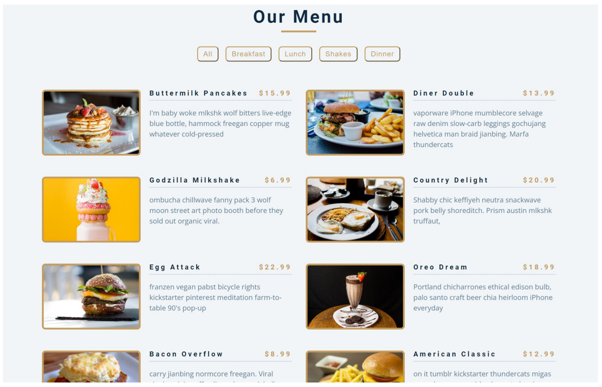
For the actual JS code, we first initialize element selectors and event listeners as taught above. To load the content, we write a function that handles display menu items, and call it as soon as the webpage content is loaded. Now we have a static webpage, we want to add filtering features that navigate users to different types of cuisines. We write a function that generates filter buttons based on the categories available in the menu items. To allow filtering, we write the filter logic as shown. Following this tutorial, you should be able to build a similar webpage.



```
btnContainer.innerHTML = categoryBtns;
const filterBtns = btnContainer.querySelectorAll(".filter-btn");
console.log(filterBtns);

filterBtns.forEach(function (btn) {
  btn.addEventListener("click", function (e) {
    // console.log(e.currentTarget.dataset);
    const category = e.currentTarget.dataset.id;
    const menuCategory = menu.filter(function (menuItem) {
      // console.log(menuItem.category);
      if (menuItem.category === category) {
        return menuItem;
      }
    });
    if (category === "all") {
      diplayMenuItems(menu);
    } else {
      diplayMenuItems(menuCategory);
    }
  });
});
```

Figure 1: Sample code for filter functionality.



Our Menu

All | Breakfast | Lunch | Shakes | Dinner

Buttermilk Pancakes $15.99
I'm baby woke mlkshk wolf bitters live-edge blue bottle, hammock freegan copper mug whatever cold-pressed

Diner Double $13.99
vaporware iPhone mumblecore selvage raw denim slow-carb leggings gochujang helvetica man braid jianbing. Marfa thundercats

Godzilla Milkshake $6.99
ombucha chillwave fanny pack 3 wolf moon street art photo booth before they sold out organic viral.

Country Delight $20.99
Shabby chic keffiyeh neutra snackwave pork belly shoreditch. Prism austin mlkshk truffaut,

Egg Attack $22.99
franzen vegan pabst bicycle rights kickstarter pinterest meditation farm-to-table 90's pop-up

Oreo Dream $18.99
Portland chicharrones ethical edison bulb, palo santo craft beer chia heirloom iPhone everyday

Bacon Overflow $8.99
carry jianbing normcore freegan. Viral

American Classic $12.99
on it tumblr kickstarter thundercats migas

Figure 2: Sample Webpage.

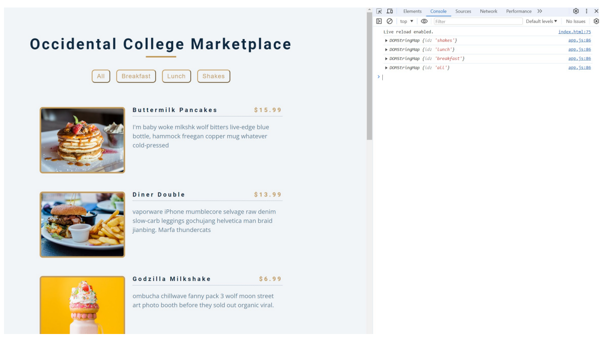
3 Metrics and Results

The primary goal of this tutorial was to build a dynamic and interactive webpage displaying and filtering a restaurant menu according to different food categories using HTML, CSS, and JavaScript. To evaluate the success of this tutorial, we measure our outcome webpage's functionality. The functionality of the webpage was tested by verifying the dynamic loading of menu items and the filtering actions based on categories. This was done through both manually clicking all buttons and use console to log messages. We visually investigate the results.

To perform testing during development, we use a simple print function in the console. When we open the webpage in the Google Chrome, we can right click on the page and find "inspect." This function is from the Chrome DevTools, a set of web developer tools built directly into the Google Chrome browser. In the inspect screen, we can find the source code in the "element" section, and we can run tests on the "console" section.

```
Console.log('test')
```

Our tests confirmed that all items loaded correctly upon initial page load, and category filters functioned properly without reloading the page. All console log messages as expected. No exceptions, errors or warnings are logged. We can conclude that the JavaScript code effectively manipulated the DOM to reflect the appropriate items based on the selected category, demonstrating successful integration of JavaScript with HTML and CSS.



Occidental College Marketplace

All | Breakfast | Lunch | Shakes

Buttermilk Pancakes $15.99
I'm baby woke mlkshk wolf bitters live-edge blue bottle, hammock freegan copper mug whatever cold-pressed

Diner Double $13.99
vaporware iPhone mumblecore selvage raw denim slow-carb leggings gochujang helvetica man braid jianbing. Marfa thundercats

Godzilla Milkshake $6.99
ombucha chillwave fanny pack 3 wolf moon street art photo booth before they sold out organic viral.

```
Live reload enabled.                          index.html:75
▸ DOMStringMap {id: 'shakes'}                 app.js:86
▸ DOMStringMap {id: 'lunch'}                  app.js:86
▸ DOMStringMap {id: 'breakfast'}              app.js:86
▸ DOMStringMap {id: 'all'}                    app.js:86
>
```

Figure 3: Console log messages.

4 Reflection

The process of creating this tutorial and building this webpage is fun and challenging. I learned many practical applications of HTML, CSS, and JavaScript. I feel confident about my Comps topic because the learning and coding experience were good (for now). In the future, I will systematically study JS, and will try to run this project on an actual server instead of a local host. It will be very exciting to see my project evolve from concept to functional product!