12.4.1 JavaScript Event Listeners

Roza is getting closer to her goal. Let's summarize what she has learned so far.

She is able to create various types of static visualizations with Plotly, such as bar and line charts.

She is also able to perform sophisticated data manipulations under the hood. She can retrieve data from an external JSON file, iterate through objects and retrieve necessary data from them, whether they are object keys or object values. She can also iterate through arrays with methods such as `map()` and `filter()`.

The missing link between the static visualizations and under-the-hood JavaScript data operations is interactivity. It is interactivity that will enable Roza to generate customizable charts dynamically. In order to make interactive visualizations, she'll first need to create JavaScript event listeners.

First we'll help Roza create a very simple dropdown menu, and then build on our skills. Let's look at the contents of the `index.html`:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Events</title>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/d3/5.9.7/d3.min.js"></
  <script src="https://cdn.plot.ly/plotly-latest.min.js"></script>
</head>
```

```
<body>
  <div id="menu"></div>
  <select id="selectOption">
    <option value="option1">First Option</option>
    <option value="option2">Second Option</option>
  </select>
<script src="script.js"></script>
</body>
</html>
```

Note the following:

- This time, there are links to two CDNs: D3 and Plotly.
- The `<select>` tag indicates a dropdown menu. Its `id` is `"selectOption"`.
- The dropdown menu has two options, as indicated by the two `<option>` tags.
- The option values `"option1"` and `"option2"` are internal names for each dropdown menu option.
- `First Option` and `Second Option` are the text displayed in the browser for each menu option.
- A plc tag links to `script.js`, a JavaScript file.

REWIND

The `<select>` tag is used to create a dropdown menu. The `<option>` tag is used to create each menu option.

Now open `script.js`:

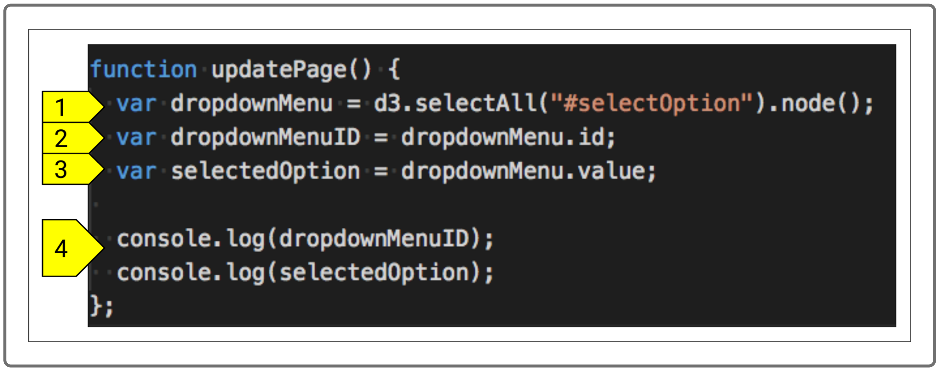
```
d3.selectAll("body").on("change", updatePage);

function updatePage() {
  var dropdownMenu = d3.selectAll("#selectOption").node();
  var dropdownMenuID = dropdownMenu.id;
  var selectedOption = dropdownMenu.value;
  console.log(dropdownMenuID);
  console.log(selectedOption);
};
```

The first line uses the `d3.selectAll()` method to create an event listener. Whenever there is a change to the HTML body, the `updatePage()` function is called. That is, when an event occurs on the page, such as selection of a dropdown menu option, the `updatePage()` function is triggered.

REWIND

`d3.selectAll().on("change", );` creates an event listener that triggers the custom function every time a change takes place to the HTML element specified by `selectAll().`

Let's look at the `updatePage()` function in greater detail:

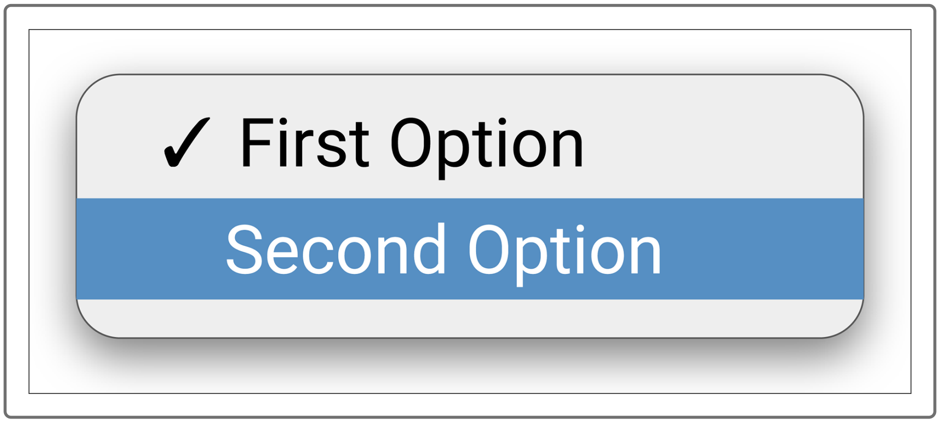


```
function updatePage() {
  var dropdownMenu = d3.selectAll("#selectOption").node();
  var dropdownMenuID = dropdownMenu.id;
  var selectedOption = dropdownMenu.value;

  console.log(dropdownMenuID);
  console.log(selectedOption);
};
```

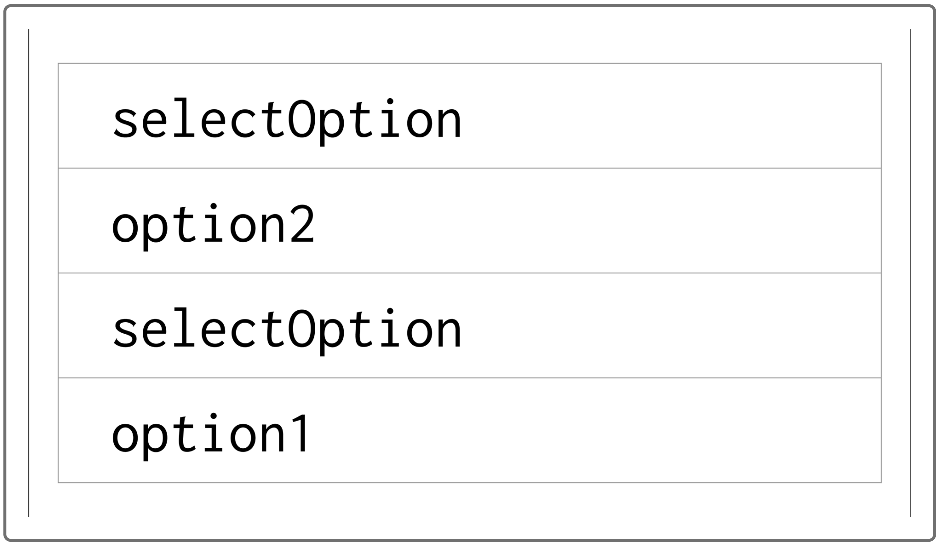
1. The function uses `d3.selectAll()` to select the dropdown menu, which has an `id` of `selectOption`.
2. The `id` of the dropdown menu, `selectOption`, is assigned the variable `dropdownMenuID`.
3. Whenever a dropdown menu option is selected, its value is assigned the variable `selectedOption`. Note that `selectOption` is the `id` value of the dropdown menu, while `selectedOption` is the option that is chosen by the user.

4. Each time `updatePage()` is triggered, the `id` value of the dropdown menu, as well as the value of the chosen menu option, are printed to the browser console.

When we open the browser, we'll see a dropdown menu with two options.



If we open the console, we'll see that every time we toggle between the two menu options, `selectOption` and the option value are printed to the console:



```
selectOption
option2
selectOption
option1
```

Now test your skills in the following Skill Drill.

SKILL DRILL

Create a new directory, containing new `index.html` and `script.js` files. Use the `D3.js` library to create an event listener for a dropdown menu.

Your dropdown menu should contain the following names: Mickey, Minnie, Donald, Goofy. When a character (e.g., Minnie) is chosen from the dropdown menu by a user, the character's name should be printed to the browser console.