Task 4: Adding Interactive Features to Your Profile Page

Objective:

Implement sorting functionality for skills, add a favorites feature for skill highlights, and show a dynamic count of favorite skills on the profile page. This task aims to enhance the user experience by adding interactivity and dynamic updates to your individual profile page.

Pre-requisites:

- Basic understanding of HTML, CSS, and JavaScript
- Familiarity with a code editor like Visual Studio Code

Concepts Covered:

- Implementing Sorting Functionality for Skills
- Adding a Favorites Feature for Skills
- Showing a Dynamic Count of Favorite Skills

Concepts:

1. **Implementing Sorting Functionality:**

   Create buttons or a dropdown for different sorting options (e.g., sort skills alphabetically). Attach event listeners to these controls that trigger a sort function.

```
<div class="sort-controls">
    <button id="sortSkills">Sort Skills Alphabetically</button>
</div>
```

```
document.getElementById('sortSkills').addEventListener('click', () => {
    profile.skills.sort((a, b) => a.localeCompare(b));
    displaySkills(profile.skills);
});
```

2. **Adding a Favorites Feature:**

   Create a simple array to represent favorite skills. Add buttons next to each skill to add items to the favorites list.

```
let favoriteSkills = [];

function addToFavorites(skill) {
    if (!favoriteSkills.includes(skill)) {
        favoriteSkills.push(skill);
        updateFavoritesCount();
    }
}
```

```
<button class="favorite-button" onclick="addToFavorites('${skill}')">Add to
Favorites</button>
```

3. **Showing a Dynamic Count of Favorite Skills:**

   Create a small badge element that shows the number of items in the favorite skills list. Update this count dynamically as items are added or removed.

```
<div class="favorites-count">Favorite Skills: <span id="favoritesCount">0</
span></div>
```

```
function updateFavoritesCount() {
    document.getElementById('favoritesCount').innerText = favoriteSkills.length;
}
```

Setup:

1. **Install Visual Studio Code (VS Code):**

   Download and install VS Code from Visual Studio Code.

2. **Web Browsers:**

   Use Google Chrome or Mozilla Firefox for viewing your webpage and utilizing their developer tools for debugging.

Tasks:

1. **Implement Sorting Functionality for Skills (10 minutes):**
   - Create buttons or a dropdown for sorting skills alphabetically.
   - Attach event listeners to these controls that trigger a sort function.
   - Example:

```
<div class="sort-controls">
    <button id="sortSkills">Sort Skills Alphabetically</button>
</div>
```

```
document.getElementById('sortSkills').addEventListener('click', () => {
    profile.skills.sort((a, b) => a.localeCompare(b));
    displaySkills(profile.skills);
});
```

2. **Add a Favorites Feature for Skills (10 minutes):**
   - Create a simple array to represent favorite skills.
   - Add buttons next to each skill to add items to the favorites list.
   - Example:

```
let favoriteSkills = [];

function addToFavorites(skill) {
    if (!favoriteSkills.includes(skill)) {
        favoriteSkills.push(skill);
        updateFavoritesCount();
    }
}
```

```
<button class="favorite-button" onclick="addToFavorites('${skill}')">Add to
Favorites</button>
```

3. **Show a Dynamic Count of Favorite Skills (10 minutes):**
    - Create a small badge element that shows the number of items in the favorite skills list.
    - Update this count dynamically as items are added or removed.
    - Example:

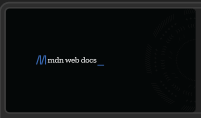
```
<div class="favorites-count">Favorite Skills: <span id="favoritesCount">0</
span></div>
```

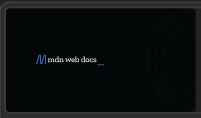
```
function updateFavoritesCount() {
    document.getElementById('favoritesCount').innerText =
favoriteSkills.length;
}
```

Instructions:

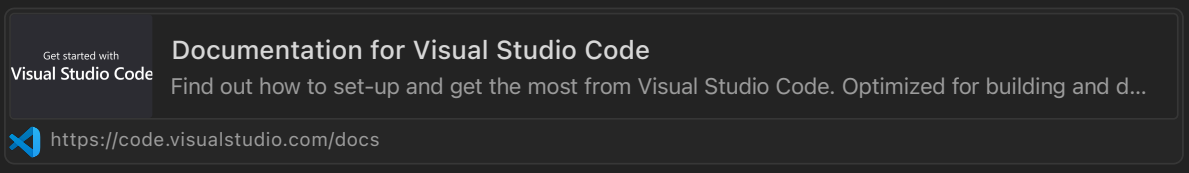
1. Write the required code in `index.html` and `script.js`.
2. Open the `index.html` file in your web browser to ensure the code displays correctly.
3. Use the browser's developer tools to debug and inspect the elements.

Resources:

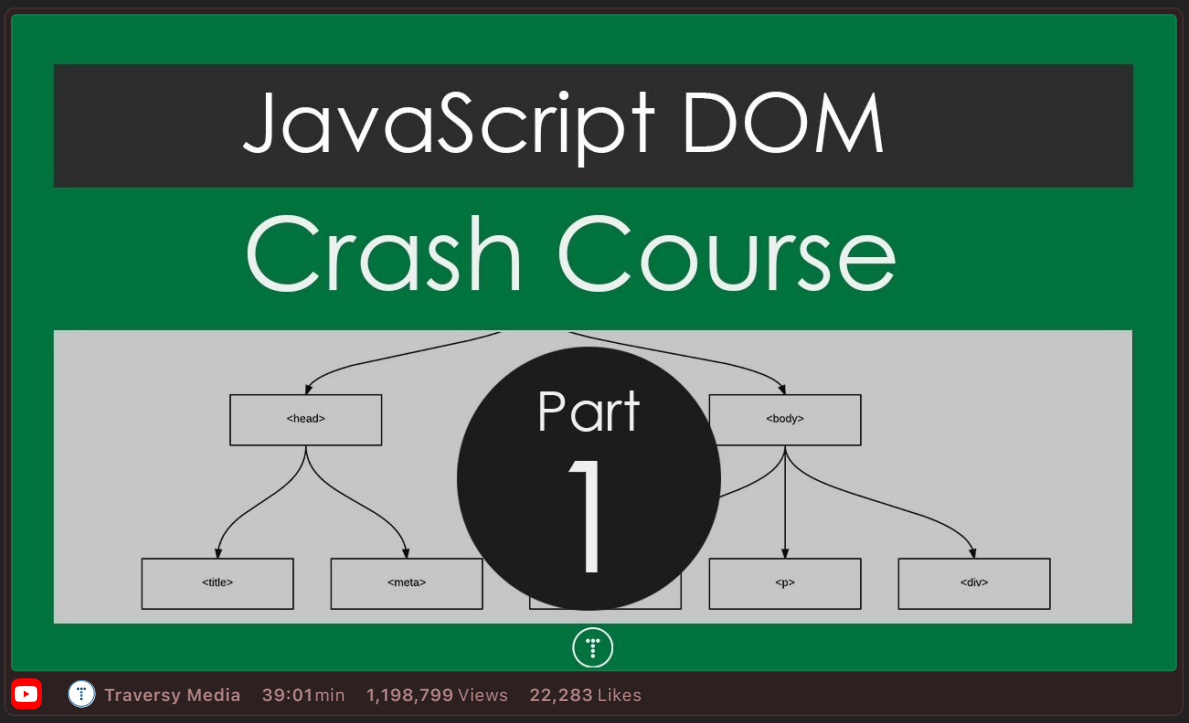
- 

Array - JavaScript | MDN
The Array object, as with arrays in other programming languages, enables storing a collecti...
https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array
- 

Introduction to the DOM - Web APIs | MDN
The Document Object Model (DOM) is the data representation of the objects that compris...
https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction

- 

Videos:

- 

GitHub Instructions:

1. **Open in Visual Studio Code:**

   After clicking on the "Open in Visual Studio Code" button from the GitHub Classroom confirmation page, VSCode will open the repository directly. If prompted, select "Open" or "Allow" to open the repository in VSCode.

2. **Open the Terminal in VSCode:**

   In VSCode, open a terminal by selecting Terminal > New Terminal from the top menu.

3. **Complete the Task:**

   In VSCode, write your solution in the `index.html` and `script.js` files.

4. **Run and Test Your Code:**

   Open your `index.html` file in a web browser to ensure it works correctly. Use the following command:

   ```
   open index.html
   ```

5. **Commit Your Changes:**

In the VSCode terminal, add your changes to git:

```
git add index.html script.js
```

Commit your changes with a meaningful message:

```
git commit –m "Completed task 15
```

"

```
6. **Push Your Changes to Your Repository:**
Push your changes to your forked repository:
```bash
git push origin main
```

7. **Create a Pull Request:**
   Go to your repository on GitHub.
   Click on the "Pull Requests" tab.
   Click the "New Pull Request" button.
   Ensure the base repository is the original template repository and the base branch is `main`.
   Ensure the head repository is your forked repository and the compare branch is `main`.
   Click "Create Pull Request".
   Add a title and description for your pull request and submit it.

Summary of Commands:

```
# Open in Visual Studio Code

# Open terminal in VSCode

# Complete the task by editing index.html and script.js

# Navigate to the directory containing index.html
cd path/to/your/index.html

# Run your code
open index.html

# Add, commit, and push your changes
git add index.html script.js
git commit –m "Completed task 4"
git push origin main
```
```

```
# Create a pull request on GitHub
```