Task 5: Mastering CSS Selectors on Your Profile Page

Objective:

This task involves applying various types of CSS selectors including ID selectors (`#id`), class selectors (`.class`), attribute selectors (`[attribute=value]`), and element selectors (like `p` , `div` , etc.). Your challenge is to use these selectors to style different components of a profile page, understanding how each selector targets and applies styles to specific elements.

Pre-requisites:

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

Concepts Covered:

- ID Selectors
- Class Selectors
- Attribute Selectors
- Element Selectors

Concepts:

1. **ID Selectors:**

   The ID selector is used to style a specific element on the page using the `id` attribute. An ID should be unique within a page.

   ```css
   #header {
       background-color: #ff5733;
       color: #fff;
       padding: 10px;
   }
   ```

2. **Class Selectors:**

   The class selector is used to style multiple elements by applying the same class attribute to them. This allows for grouping and collectively styling elements.

   ```css
   .highlight {
       background-color: #f0f0f0;
       color: #333;
       padding: 5px;
       border-radius: 4px;
   }
   ```

3. **Attribute Selectors:**

   Attribute selectors target elements based on their attributes and attribute values. They are useful for styling elements like input fields based on their type.

```
input[type="text"] {
    border: 2px solid #ff5733;
    padding: 5px;
}
```

4. **Element Selectors:**

   Element selectors style HTML tags directly. They apply styles to all instances of a specified element type.

   ```
   p {
       font-size: 16px;
       line-height: 1.5;
       color: #333;
   }
   ```

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. **ID Selectors :**
   - Use the ID selector to style a specific element on the page. For example, apply a unique style to the header or a specific section.
   - Example:

     ```
     <div id="header">John Doe's Profile</div>
     ```

     ```
     #header {
         background-color: #ff5733;
         color: #fff;
         padding: 10px;
         text-align: center;
     }
     ```

2. **Class Selectors :**
   - Apply class selectors to group multiple elements and style them collectively. Experiment with applying the same class to different types of elements and observe the results.
   - Example:

```
<p class="highlight">Welcome to my profile page.</p>
<div class="highlight">This is a highlighted section.</div>
```

```
.highlight {
    background-color: #f0f0f0;
    color: #333;
    padding: 5px;
    border-radius: 4px;
}
```

3. **Attribute Selectors :**
   - Utilize attribute selectors to apply styles to elements based on their attributes, such as styling all input fields of a certain type.
   - Example:

```
<input type="text" placeholder="Enter your name">
<input type="password" placeholder="Enter your password">
```

```
input[type="text"] {
    border: 2px solid #ff5733;
    padding: 5px;
}
```

4. **Element Selectors :**
   - Use element selectors to style HTML tags directly, like paragraphs, headings, and links.
   - Example:

```
<p>This is a paragraph.</p>
<h1>Profile Heading</h1>
<a href="#">This is a link</a>
```

```
p {
    font-size: 16px;
    line-height: 1.5;
    color: #333;
}
h1 {
    font-size: 2em;
    color: #ff5733;
}
a {
    color: #ff5733;
    text-decoration: none;
}
a:hover {
    text-decoration: underline;
}
```

Instructions:

1. Write the required code in `index.html` and `styles.css` .
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:

- **CSS selectors - CSS: Cascading Style Sheets | MDN**
  The CSS selectors module defines the patterns to select elements to which a set of CSS ru...
  https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Selectors
- **ID selectors - CSS: Cascading Style Sheets | MDN**
  The CSS ID selector matches an element based on the value of the element's id attribute. I...
  https://developer.mozilla.org/en-US/docs/Web/CSS/ID_selectors
- **Class selectors - CSS: Cascading Style Sheets | MDN**
  The CSS class selector matches elements based on the contents of their class attribute.
  https://developer.mozilla.org/en-US/docs/Web/CSS/Class_selectors
- **Attribute selectors - CSS: Cascading Style Sheets | MDN**
  The CSS attribute selector matches elements based on the element having a given attribut...
  https://developer.mozilla.org/en-US/docs/Web/CSS/Attribute_selectors
- **🤷 Page not found | MDN Web Docs**
  The MDN Web Docs site provides information about Open Web technologies including HTML, C...
  https://developer.mozilla.org/en-US/docs/Web/CSS/Element_selectors
- Get started with Visual Studio Code
  Documentation for Visual Studio Code
  Find out how to set-up and get the most from Visual Studio Code. Optimized for building and d...
  https://code.visualstudio.com/docs

Videos:

-

  Web Dev Simplified 19:38 min 441,212 Views 17,475 Likes

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 `styles.css` 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 styles.css
   ```

   Commit your changes with a meaningful message:

   ```
   git commit -m "Completed task 9"
   ```

6. **Push Your Changes to Your Repository:**

   Push your changes to your forked repository:

   ```
   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 styles.css

# 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 styles.css
git

 commit -m "Completed task 5"
git push origin main

# Create a pull request on GitHub
```