Task 2: Mastering CSS Box Model, Margins, and Padding

Objective:

Master the CSS Box Model concept and use margins and paddings in your profile page. This task aims to provide a deep understanding of how HTML elements are structured and styled in terms of spacing, borders, and content organization, which are fundamental for effective web design.

Pre-requisites:

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

Concepts Covered:

- CSS Box Model
- Margins
- Padding

Concepts:

1. **CSS Box Model:**

   The CSS Box Model describes how the space of an element is composed. It includes margins, borders, padding, and the actual content.

```
.box {
    border: 2px solid #000;
    margin: 10px;
    padding: 20px;
    background-color: #f0f0f0;
}
```

2. **Margins:**

   Margins are used to create space around elements, outside of any defined borders. Margins can be set for all four sides (top, right, bottom, left).

```
.box {
    margin-top: 10px;
    margin-right: 20px;
    margin-bottom: 10px;
    margin-left: 20px;
}
```

3. **Padding:**

   Padding is used to create space around an element's content, inside of any defined borders. Padding can also be set for all four sides.

```
.box {
    padding-top: 10px;
    padding-right: 20px;
    padding-bottom: 10px;
    padding-left: 20px;
}
```

Setup:

1. **Install Visual Studio 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. **Exploring the Box Model :**
   - Create a simple HTML page with several `div` elements.
   - Apply different border, margin, and padding values to these elements using CSS.
   - Use the browser's developer tools to inspect how each property affects the element's box.
2. **Margin Collapsing (20 minutes):**
   - Experiment with adjoining elements having margins. Observe how margins collapse/ combine in various scenarios.
   - Test different combinations of margin values (positive and negative) to see the effects on layout spacing.
3. **Padding and Content Area :**
   - Modify the padding of elements and observe how it impacts the content area.
   - Use different background colors to clearly see the effect of padding.

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.

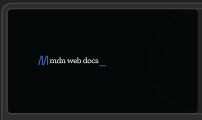
Resources:

- 

  The box model - Learn web development | MDN

  That's most of what you need to understand about the box model. You may want to return...

  https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/The_box_model

- 

  margin - CSS: Cascading Style Sheets | MDN
  The margin CSS shorthand property sets the margin area on all four sides of an element.
  https://developer.mozilla.org/en-US/docs/Web/CSS/margin
- 

  padding - CSS: Cascading Style Sheets | MDN
  The padding CSS shorthand property sets the padding area on all four sides of an element...
  https://developer.mozilla.org/en-US/docs/Web/CSS/padding
- 

  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:

- 

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 6"
   ```

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 2"
git push origin main

# Create a pull request on GitHub
```