CSE 391: Programming for the Internet

Assignment 1: Creating a Personal Web Page

Objective

Your first assignment is to create a personal homepage that will be published on the Web. This page will introduce you to the world and serve as a central hub for web-based applications you develop throughout the semester.

Deadline: As per the lab instructor.

You have full creative freedom over the design and content, but your homepage must demonstrate solid HTML and CSS principles. The primary focus should be on page structure, design, and styling with full CSS. At a minimum, your webpage must include the following elements:

PART 1: Basic Structure & Content (30 points)

1. Your webpage should be divided into multiple sections:
    - A section with personal information (e.g., name, background, etc.).
    - A section with categorized links (e.g., hobbies, sports, extracurricular activities).
2. Implement various types of navigation:
    - Anchor links (for jumping to different sections within the same page).
    - Internal links (to other pages within your website, such as hobbies, experiences, or projects).
    - External links (to relevant outside websites).
3. Your name must be prominently displayed at the top, and your own photo should be included.
4. Clearly labeled sections (e.g., "Personal Info," "About Me," "Education," "Technical Skills").

PART 2: Formatting & Content Enhancement (25 points)

1. Apply text formatting effectively:
    - Use bold, italics, color variations, and different font sizes (without excessive styling).
2. Include a favorite quote (e.g., a poem, movie line, or personal motto).
3. Add at least one list (e.g., top 10 favorite movies, books, or albums).

4. Include at least 5 hyperlinks:
    - One must be embedded within a sentence.
    - One must link to BRAC University.
    - One must link to the CSE 391 Discord page.

PART 3: Layout & Styling with CSS (25 points)

1. Your webpage must contain at least two images:
    - One image must be stored locally.
    - Other images can be linked from the web.
2. Include a properly structured table:
    - Some rows should contain multi-line content (e.g., a short paragraph).
    - Use row and column merging where appropriate.
3. Implement the following CSS effects:
    - Hover effects.
    - Striped rows and table borders.
    - Ensure the table is fully responsive.
4. Use external, internal, and inline CSS effectively:
    - Create a separate CSS file for overall styling.
    - Use internal and inline styles sparingly for minor elements (e.g., bold text, width adjustments).
5. Display the page's location and last modification date at the bottom using JavaScript.

PART 4: Code Quality & Standards (20 points)

1. Ensure your webpage follows correct HTML structure:
    - Must include `DOCTYPE`, `<html>`, `<head>`, and `<body>`.
    - Include appropriate `<meta>` tags (for charset, viewport settings, etc.).
    - Must have a `<title>`, header, body content, and footer with copyright information.
2. Your code should adhere to web standards:
    - Use valid HTML and CSS.
    - Pass W3C validation checks (Strict XHTML 1.0 or 1.1 standard).
    - Use Tidy or Validator (link provided below) for validation.

Validator: W3C Validator

Bonus Challenges (Extra Credit)

1. **Add a contact form** that sends an email when submitted.
2. **Implement a dark/light mode toggle** using CSS.
3. **Create CSS animations** to enhance visual appeal.

Submission Guidelines

- Submit your project files in a compressed `.zip` format.
- Include all necessary files (HTML, CSS, images, JavaScript, if any).
- Ensure your submission is functional and meets the requirements.

Note: Direct AI-generated answers will be easily detectable. Personalize your page, structure it logically, and write meaningful content to get full marks.

Good luck and happy coding!