CSE211 Web Programming - Assignment 1
Comprehensive Web Development: Performance, Semantic Architecture, and HTML5 Implementation

Submission Deadline: Saturday, December 6, 2025, before 11:59 PM
Lab Assessment: During Week 11 Labs (20 points)
Work Mode: Group Assignment - 2 Students per Group
Grade Distribution:

- **Group Work (Submitted Assignment): 80% of total grade**
- **Individual Assessment (Week 11 Lab): 20% of total grade**

Students are ALLOWED and ENCOURAGED to use AI tools (such as ChatGPT, Claude, GitHub Copilot, etc.) as learning aids and guidance tools for this assignment.

1. Instructions

- The total number of pages in this file is 20 pages.
- Students complete the assignment in order.
- All students must have a GitHub account.
- Each group MUST submit the group work deliverables on CANVAS and GitHub.
- Each student **MUST** solve and submit the Lab Component (Individual Assessment) **DURING WEEK 11 LAB**.
- **Failure to attend week 11 Lab and complete the assignment will result in a ZERO in this assessment.**
- **Students are ALLOWED and ENCOURAGED to use AI tools** (such as ChatGPT, Claude, GitHub Copilot, etc.) as learning aids and guidance tools for this assignment.
- However, you MUST:
  - At the top of each main HTML file, include a comment specifying how you used AI
  - Understand every line of code you submit - You will be assessed individually in the lab and must be able to explain your code
  - Write and test your own implementation - AI should guide you, not write your assignment for you
  - Document AI usage in your report - Include a section in your README.txt file titled "AI Tools Usage" that describes:
    - Which AI tools you used
    - How you used them (e.g., "Asked ChatGPT to explain semantic HTML elements", "Used Claude to understand form validation attributes")
    - What you learned from using these tools
    - Confirmation that you understand and can explain all submitted code
- **Acceptable AI Use:**
  - Asking for explanations of HTML concepts & Understanding error messages and validation issues.
  - Learning about semantic HTML best practices, getting guidance on proper HTML structure, and debugging syntax errors with explanations
- **Unacceptable AI Use:**
  - Copying and Submitting code you cannot explain
  - Not documenting AI usage as required
- **Important:** During the lab assessment, you will be evaluated individually on your understanding of the code. If you cannot explain your own code, you will lose significant points regardless of how well the code is written.

2. Assignment description and requirements

This comprehensive assignment integrates modern web development practices with fundamental HTML5 implementation. You will analyze web performance metrics, design semantic architecture, and develop a complete HTML5 website with advanced forms and proper structure.

2.1. General instructions:

This assignment consists of TWO major components

1- Group Work (Submitted Assignment): this component consists of Five parts
   Part 1: Project Setup and Structure.
   Part 2: Core Web Vitals Performance Analysis.
   Part 3: Complete Website Development with Semantic HTML5.
   Part 4: Calculator Implementation.
   Part 5: Validation and Quality Assurance.
2- Individual Assessment

2.2. Group Work (Submitted Assignment) (80% of total grade)

The group work consists of 5 parts and will be submitted as described in the deliverables section on CANVAS and GitHub.

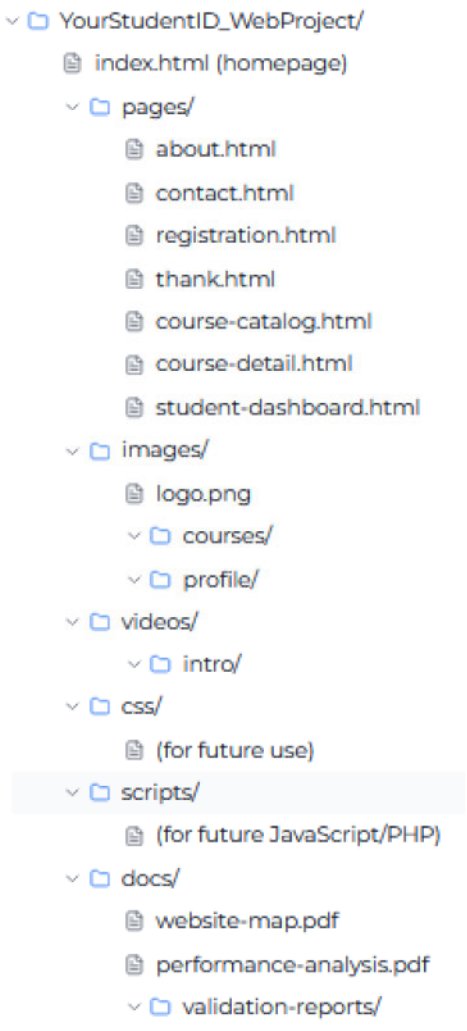
2.2.1. Part 1: Project Setup and Structure (10 points)

2.2.1.1. Logical folder structure (5 points)

Create a professional folder structure for your web project as shown in the following figure:

Root Directory Structure:

- YourStudentID_WebProject (root folder)
  - index.html (homepage - must be in roo t)
  - pages folder (all other HTML pages)
  - images folder (all image files)
  - videos folder (all video files)
  - css folder (empty for now, for future use)
  - scripts folder (for future JavaScript/PHP)
  - docs folder (all documentation and reports)



Requirements:

- Use descriptive, lowercase file names with hyphens, not spaces
- Homepage must be named index.html in root directory
- All other HTML pages must be in pages folder
- Organize assets by type in appropriate folders
- Create docs folder for all documentation and reports
- Use logical subfolders within images folder for organization

2.2.1.2. Website Map and Navigation Structure (5 points)

Create a comprehensive website map illustrating:

- Page hierarchy and relationships between all pages
- Primary navigation structure (main menu)
- Secondary navigation elements (breadcrumbs, footer links)
- Internal links between pages
- External links (to university website, social media)
- Special links (mailto, anchors)

Deliverable Format: Visual diagram created using:

- Draw.io, Lucidchart, Figma, or
- Hand-drawn and professionally scanned, or
- Any professional diagramming tool

Must Include:

- All 7+ pages and their connections
- Link types indicated (internal, external, anchor, mailto)
- Navigation menu structure
- Footer link structure
- Breadcrumb navigation paths

2.2.2. Part 2: Core Web Vitals Performance Analysis (20 points)

Conduct professional-grade performance audits to understand real-world web optimization challenges.

2.2.2.1. Website Selection & Technology Analysis (6 points)

Select and analyze three websites from different domains:

Required Categories:

1. E-commerce Platform (e.g., Amazon, eBay, local e-commerce)
2. News/Media Outlet (e.g., BBC, CNN, Al-Ahram)
3. Educational Institution (e.g., university website, online learning platform)

For Each Website Document:

- Complete homepage URL
- Server location (if identifiable)
- Technology stack identification:
  - Content Management System (WordPress, Drupal, custom)
  - Frontend frameworks (React, Vue, Angular, or none)
  - Identify using browser DevTools (Network tab, Sources tab) or Wappalyzer extension
- Initial measurements (take 3 measurements, calculate average):
  - Page load time (seconds)
  - Total page size (MB)
  - Number of HTTP requests
  - Number of images, scripts, stylesheets

Analysis Method: Use browser DevTools Network tab:

1. Open DevTools (F12)
2. Go to Network tab
3. Hard refresh page (Ctrl+Shift+R)
4. Record "Finish" time, "Transferred" size, and request count
5. Repeat 3 times at different times of day
6. Document with screenshots

2.2.2.2. Core Web Vitals Measurement & Analysis (10 points)

Tools Required:

- Google PageSpeed Insights (https://pagespeed.web.dev/)
- Chrome Lighthouse (built into DevTools)

Measure for Each Website:

A) Largest Contentful Paint (LCP) - 3 points

- Records load time of largest content element

- Target: < 2.5 seconds (Good), 2.5-4.0s (Needs Improvement), > 4.0s (Poor)
- Identify the specific element causing LCP (usually main image or hero section)
- Measure on mobile and desktop
- Document differences and reasons

B) Interaction to Next Paint (INP) - 3 points

- Measures interaction responsiveness
- Target: < 200ms (Good), 200-500ms (Needs Improvement), > 500ms (Poor)
- Test by clicking navigation, buttons during load
- Document any delays experienced

C) Cumulative Layout Shift (CLS) - 3 points

- Measures visual stability
- Target: < 0.1 (Good), 0.1-0.25 (Needs Improvement), > 0.25 (Poor)
- Identify elements causing layout shifts (images without dimensions, dynamic content)
- Document unexpected movement during page load

Required Documentation - 1 point

- Screenshot of PageSpeed Insights results for each site
- Mobile vs Desktop performance comparison table
- Identify top 3 issues flagged by Lighthouse for each site

2.2.2.3. Comparative Analysis & Optimization Strategy (4 points)

Create Comparison Table Including:

- Overall performance scores (0-100)
- Load time comparison
- Core Web Vitals scores
- Page size and request count
- Performance ranking (1st, 2nd, 3rd place)

For the Worst-Performing Website, Provide:

A) Performance Bottleneck Identification (minimum 3):

- Unoptimized images (large file sizes, wrong formats)
- Render-blocking resources (CSS/JS)
- Lack of caching
- Too many HTTP requests
- Large DOM size
- Slow server response time

B) Concrete Optimization Recommendations:

- Image Optimization: Compress, use WebP format, implement lazy loading, specify dimensions
- Code Optimization: Minify CSS/JS, remove unused code, implement code splitting
- Caching Strategy: Browser caching, CDN usage, service workers
- Critical Rendering Path: Inline critical CSS, defer non-critical JS, preload key resources
- Content Delivery: Use CDN, enable compression (Gzip/Brotli)

Deliverable: Professional PDF report with:

- Executive summary & methodology section
- Detailed findings for each website
- Comparison analysis
- Optimization recommendations
- Screenshots and data tables
- Conclusion

2.2.3. Part 3: Complete Website Development with Semantic HTML5 (50 points)

Design and develop a complete, professional website using only HTML5, implementing proper semantic structure, accessibility, and web standards.

Website Theme: EduConnect - Online Learning Platform
Your website represents an online learning platform that helps students discover courses, register, and manage their learning journey.

2.2.3.1. Required Pages and Specifications (35 points)

Page 1: Homepage - index.html (8 points)

Layout Requirements:

- Must use two-column layout
- Left column (main content): 70% width
  - Welcome section with <header>
  - Featured courses section using <section> with multiple <article> elements
  - Latest announcements using <article> elements
  - Statistics section using <dl> (definition list)
- Right column (sidebar): 30% width
  - Search form (text input + submit button)
  - Quick navigation links
  - Marquee displaying latest news (using appropriate semantic tags)
  - Advertisement section using <aside>

Mandatory Elements:

- <header> containing:
  - Logo/site title
  - Tagline
  - Primary navigation menu
- Navigation breadcrumb showing: Home
- <nav> element with links to all pages:
  - About Us
  - Course Catalog
  - Course Details (sample)
  - Student Dashboard
  - Registration
  - Contact Me
  - External link to your university website
- <main> containing two-column content
- <footer> containing:
  - Mailto link to your email
  - Links to social media profiles (LinkedIn, Facebook, Twitter - can be placeholder links)
  - Copyright notice: "© 2025 EduConnect. All rights reserved."
  - Link to "About Us" page
  - Last updated date

Content Requirements:

- Minimum 3 featured courses with images
- Use at least one <figure> with <figcaption>
- Include appropriate headings hierarchy (h1 for page title, h2 for sections, h3 for subsections)

Page 2: About Us - about.html (5 points)

Content:

- Introduction to CSE211 Web Programming course
- Course objectives (use ordered list <ol>)
- Topics covered (use unordered list <ul>)
- Instructor information (use <dl> definition list)
- Course benefits section

Layout:

- Same header and footer as homepage
- Breadcrumb navigation: Home > About Us
- Single column layout with proper semantic sections
- Include at least one image with proper alt text
- Use <blockquote> for course philosophy or mission statement

Page 3: Contact Me - contact.html (4 points)

Content:

- Your email address (as clickable mailto link)
- University address with proper formatting
- Phone numbers (university contact)
- Office hours (use table structure)
- Link to university website (opens in new tab)
- Embedded location map (use placeholder image or iframe if allowed)

Layout:

- Use <address> element semantically
- Contact information organized in <dl> or structured divs
- Same header and footer structure
- Breadcrumb: Home > Contact Me

Page 4: Registration Page - registration.html (10 points)

Must Implement Comprehensive Registration Form Including:

Form Structure: Form must have action attribute pointing to pages/thank.html, method attribute set to get, name attribute as registrationForm, and id attribute as registrationForm

Section 1: Personal Information

Use fieldset element with legend stating "Personal Information"

Must include the following fields:

- Full Name: text input, required, minlength 3 characters, maxlength 50 characters, placeholder text "Enter your full name", title attribute for tooltip, autofocus attribute
- Student ID: text input, required, pattern requiring exactly 9 digits, placeholder "123456789", title explaining format
- Email Address: email input type, required, placeholder "student@example.com", title for valid email, autocomplete set to email
- Password: password input type, required, minlength 8 characters, maxlength 20 characters, placeholder "Minimum 8 characters", title for requirements, autocomplete set to new-password
- Confirm Password: password input type, required, minlength 8 characters, maxlength 20 characters, placeholder "Re-enter your password"
- Date of Birth: date input type, required, max attribute set to ensure 18 years minimum age, title for age requirement, autocomplete set to bday
- Gender: radio button group with three options: Male, Female, Prefer not to say - all marked

as required, grouped in nested fieldset with legend
- Profile Photo: file input type, accept attribute set to image files only, title for upload instructions

Section 2: Academic Information
Use fieldset element with legend stating "Academic Information"
Must include the following fields:
- Current Degree Level: radio button group with options Bachelor's Degree, Master's Degree, PhD - all required, first option checked by default, grouped in nested fieldset
- Major/Department: select dropdown, required, with optgroup elements for faculties:
  - Faculty of Engineering containing: Computer Science, Information Technology, Software Engineering, Computer Engineering
  - Faculty of Business containing: Marketing, Finance, Management, Accounting
  - Faculty of Science containing: Mathematics, Physics, Biology, Chemistry
  - First option as "-- Select Your Major --" with empty value
- Current Semester: number input, required, min value 1, max value 12, placeholder "1-12", title for valid range
- Current GPA: number input, optional, min 0, max 4, step 0.01, placeholder "0.00 - 4.00", title for format
- Previous Programming Experience: checkbox group with options Python, Java, C++, JavaScript, PHP, C#, None - grouped in nested fieldset with legend

Section 3: Course Preferences
Use fieldset element with legend stating "Course Preferences"
Must include the following fields:
- Preferred Learning Mode: radio button group, required, with options On-Campus, Online, Hybrid Mixed - grouped in nested fieldset
- Course Categories of Interest: multiple select element, required, size attribute set to 5, with options: Web Development, Mobile Development, Data Science, Artificial Intelligence, Cybersecurity, Game Development, Cloud Computing, Database Management, DevOps, Blockchain Technology - include help text "Select 1-3 categories (Hold Ctrl/Cmd for multiple)"
- Preferred Class Schedule: select dropdown, required, with options: "-- Select Schedule --" with empty value, Morning 8:00 AM - 12:00 PM, Afternoon 12:00 PM - 4:00 PM, Evening 4:00 PM - 8:00 PM, Weekend Saturday & Sunday
- Primary Course Selection: select dropdown, required, with optgroup elements:
  - Web Development containing: HTML & CSS Fundamentals, Advanced JavaScript, React.js Development, Node.js & Backend
  - Mobile Development containing: Android Development, iOS Development, Flutter Cross-Platform
  - Data & AI containing: Python for Data Science, Machine Learning, Deep Learning
  - First option as "-- Select Primary Course --" with empty value
- Alternative Courses: multiple select element, optional, size 4, with options: SQL & Databases, Git & Version Control, Docker & Containers, AWS Cloud Basics, Web Security, Software Testing - title "Select up to 3 alternative courses"

Section 4: Additional Information
Use fieldset element with legend stating "Additional Information"
Must include the following fields:

- How did you hear about us?: select dropdown with options: "-- Please Select --" with empty value, Search Engine (Google, Bing), Social Media, Friend or Colleague, University Announcement, Online Advertisement, Other
- Special Requirements or Accessibility Needs: textarea element, rows 4, cols 50, maxlength 500 characters, placeholder text, title attribute, with character counter display
- Your Learning Goals and Expectations: textarea element, rows 5, cols 50, maxlength 500 characters, placeholder text about goals

Section 5: Terms and Conditions
Use fieldset element with legend stating "Terms and Conditions"
Must include the following fields:

- Terms Agreement: checkbox, required, with label text "I agree to the Terms and Conditions" with link to terms page
- Privacy Policy: checkbox, required, with label text "I have read and accept the Privacy Policy" with link to privacy page
- Newsletter Subscription: checkbox, optional, with label "Subscribe to our newsletter for course updates and special offers"
- Marketing Communications: checkbox, optional, with label "I agree to receive marketing communications via email"

Form Controls Section:
Must include:

- Hidden field for form version with value 1.0
- Hidden field for submission date with id attribute
- Hidden field for user agent with id attribute
- Submit button with type submit, name submitBtn, id submitBtn, text "Register Now"
- Reset button with type reset, name resetBtn, id resetBtn, title "Clear all form fields", text "Clear Form"
- Additional button with type button, name saveDraftBtn, id saveDraftBtn, text "Save Draft"

Complete Form Requirements:
A) Validation Attributes - Must Include:

- required attribute on all mandatory fields
- pattern attribute with custom regex for Student ID and other fields needing format validation
- minlength and maxlength attributes for text inputs
- min, max, and step attributes for number and date inputs
- type attributes: email, tel, date, number, url, etc.
- placeholder attribute providing helpful examples
- title attribute for tooltip hints on hover

B) Accessibility Requirements:

- Every input, select, textarea must have associated label element using for attribute
- Use fieldset and legend elements to group related form controls
- Include ARIA attributes where necessary such as aria-describedby, aria-label
- autofocus attribute on first field of form
- Logical tabindex for keyboard navigation
- Required field indicators such as asterisk with class="required"

C) User Experience Features:

- autocomplete attributes for appropriate fields: email, name, tel, etc.

- Help text using span class="help-text" for complex fields
- Character counters for textarea fields
- Multiple input types for better mobile keyboards
- Grouped options using optgroup elements in select elements
- size attribute on multi-select for better visibility

D) Form Structure Best Practices:

- Logical grouping of related fields using fieldset elements
- Clear section headers using legend elements
- Proper nesting: div elements for layout, span elements for inline elements
- Hidden fields for tracking and metadata
- Multiple button types: submit, reset, button

Page 5: Thank You Page - thank.html (3 points)

Content:

- Large heading: "Thank You for Registration!"
- Personalized message
- Confirmation message: "We have received your registration. You will receive a confirmation email shortly."
- Summary of submitted information if using GET method, can display query parameters
- Next steps section explaining what to expect
- Timeline of the registration process
- Link to return to homepage
- Link to go to student dashboard
- Contact information for questions

Layout:

- Centered content using semantic structure
- Use section elements for different content areas
- Same header and footer as other pages
- Breadcrumb showing: Home > Registration > Thank You
- Use dl definition list to display submission summary

Special Elements:

- Use time element to display submission date and time with datetime attribute
- Use address element for contact information
- Include reference number, can be simulated

Page 6: Course Catalog - course-catalog.html (8 points)

This page demonstrates advanced semantic HTML5 structure

Layout:

- Header element with page title and navigation
- Breadcrumb showing: Home > Course Catalog
- Main element containing:
  - Aside element on left, 20% width, with filter options:
    - Filter by Category using checkboxes in form
    - Filter by Level using radio buttons: Beginner, Intermediate, Advanced
    - Filter by Duration using select dropdown
    - Filter by Rating using range input or select
    - Apply Filters button
  - Main content area, 80% width, with course grid

Course Display:

- Minimum six courses displayed
- Each course is an article element containing:
  - Course image using figure element and figcaption element
  - Course title using h3 heading
  - Short description using paragraph element
  - Instructor name
  - Duration, level, price displayed using dl definition list
  - Rating using text or symbols
  - "View Details" link to course-detail.html
- Use unordered list ul element for course features or highlights within each article

Additional Required Elements:

- Table showing course comparison with minimum three courses:
  - Columns: Course Name, Price, Duration, Level, Rating, Enrollment
  - Use thead, tbody elements and proper th elements
  - Include caption element for table description
  - Use colspan or rowspan for at least one merged cell
- Search form at top of catalog
- Sorting options dropdown: Sort by Popularity, Price, Rating, Newest
- Results counter showing number of courses displayed

Page 7: Course Detail Page - course-detail.html (4 points)

Content:

- Breadcrumb showing: Home > Course Catalog > [Course Name]
- Header element with course title and subtitle
- Section with id="overview" containing:
  - Course hero image using figure element
  - Detailed description paragraph
  - What you'll learn using ordered list
  - Requirements or Prerequisites using unordered list ul
  - Course level badge
  - Duration and time commitment
- Section with id="syllabus" containing:
  - Section heading
  - Course modules with each module as nested article element
  - Use nested section elements within articles for individual lessons
  - Expandable content structure using heading hierarchy h3, h4, h5
- Aside element containing:
  - Course information box with dl definition list showing: Price, Duration, Level, Language, Certificate offered, Enrollment count
  - Instructor bio with photo using figure element
  - Related courses list with links
  - "Enroll Now" call-to-action button or link
  - Share on social media links
- Section with id="reviews" containing:
  - Section heading
  - Average rating display
  - Student reviews with minimum five reviews, each as article element

- Each review contains: Student name and optional photo, Rating using stars or number, Date using time element with datetime attribute, Review title, Review text, Helpful votes counter

Demonstrate Complex Nesting:

- Article elements containing section elements for syllabus modules contain lesson sections
- Section elements containing multiple article elements for reviews section contains review articles
- Proper heading hierarchy: h1 for page, h2 for major sections, h3 and higher for subsections

Special Requirements:

- Include internal anchor links for quick navigation: Jump to Overview, Jump to Syllabus, Jump to Reviews, Back to Top
- Use dl definition lists for course specifications and instructor details
- Include at least three figure elements with figcaption elements

Page 8: Student Dashboard - student-dashboard.html (3 points)

Content:

- Personal header area with:
  - Welcome message with student name using h1 heading
  - Profile photo placeholder using figure element
  - Quick stats using dl definition list showing: Total Courses, Completed, In Progress, Certificates
- Section with id="enrolled-courses" containing:
  - Section heading using h2
  - List of enrolled courses with minimum three, each as article element
  - Each course article includes: Course thumbnail image, Course name and instructor, Progress indicator using text-based percentage, Last accessed date using time element, "Continue Learning" link
- Section with id="progress-statistics" containing:
  - Section heading using h2
  - Overall statistics using dl definition list: Courses Completed, Courses In Progress, Total Learning Hours, Certificates Earned, Average Quiz Score, Current Streak
- Section with id="upcoming-deadlines" containing:
  - Section heading using h2
  - Upcoming deadlines using ordered list ol by date
  - Each deadline includes: Assignment or quiz name, Course name, Due date using time element, Priority indicator
- Section with id="recent-activity" containing:
  - Section heading using h2
  - Recent activity feed with minimum five activities, each as article element
  - Activity types: Course started, Assignment submitted, Quiz completed, Certificate earned
  - Each activity includes: Activity icon or type indicator, Activity description, Timestamp using time element
- Aside element containing:
  - Recommended courses section
  - Achievements or badges section using list with images
  - Learning goals progress

- Calendar placeholder using table or image
- Quick links section

Layout Requirements:

- Multi-column grid layout using div elements
- Breadcrumb showing: Home > Dashboard
- Same header and footer structure as all pages
- Logical flow of information

2.2.3.2. Universal Page Requirements (15 points)

Applied to ALL Pages:

A) HTML5 Semantic Structure (5 points)

Every page must include:

DOCTYPE and HTML tag: All pages must begin with DOCTYPE html declaration and html element with lang attribute set to "en"

Complete HEAD section: Must contain:

- meta element for charset set to UTF-8
- meta element for viewport with width=device-width, initial-scale=1.0
- meta element for description with page-specific description for SEO
- meta element for keywords with relevant keywords separated by commas
- meta element for author with your full name
- title element with descriptive page title followed by "- EduConnect"

Complete BODY structure: Must contain:

- header element for site header with logo and navigation
- nav element for primary navigation menu
- Breadcrumb navigation using nav element with aria-label="breadcrumb" or div element, except on homepage
- main element for main content area
- footer element for site footer

B) Consistent Navigation Structure (3 points)

Primary Navigation Menu in every page header: Must include links to:

- Home linking to index.html or ../ depending on page location
- About Us
- Course Catalog
- Student Dashboard
- Registration
- Contact Me
- University Name as external link with target="_blank" and rel="noopener noreferrer"

Navigation structure must use nav element with id="main-navigation" and role="navigation", containing unordered list ul with list items li containing anchor links

Breadcrumb Navigation on every page except homepage: Must use nav element with aria-label="breadcrumb" and class="breadcrumb", showing link to Home using greater than symbol to separate from current page name

Footer Links on every page: Footer element must contain:

- div with class="footer-content" containing multiple subsections
- Footer links section with links to: About Us, Contact, Privacy Policy placeholder, Terms of Service placeholder
- Footer contact section with mailto link
- Footer social media section with links to: LinkedIn, Facebook, Twitter - all with target="_blank" and rel="noopener"
- Footer copyright section with paragraph showing "© 2025 EduConnect. All rights reserved." and Last Updated using time element with datetime attribute

C) HTML5 Code Quality Standards (4 points)

1. Proper Nesting (1 point):

- All tags correctly opened and closed in proper order
- No overlapping tags
- Parent-child relationships respected
- Example of correct nesting: section element contains article element which contains h2 heading and paragraph elements, all properly closed

2. Proper Indentation (1 point):

- Consistent 2-space or 4-space indentation throughout
- Child elements indented one level from parent
- Closing tags aligned with opening tags
- Nested elements clearly show hierarchy through indentation

3. Semantic Tag Usage (1 point): Use appropriate HTML5 semantic elements:

- header element for introductory content
- nav element for navigation links
- main element for main content
- article element for self-contained content
- section element for thematic grouping
- aside element for tangentially related content
- footer element for footer content
- figure and figcaption elements for images with captions
- time element for dates and times with datetime attribute
- address element for contact information

4. Proper Attributes (1 point):

Required attributes for specific elements:

- id attribute for unique identification and anchor targets
- class attribute for grouping and styling hooks
- alt attribute for ALL images with descriptive alternative text
- title attribute for additional information
- for attribute on labels matching id of form inputs
- name attribute on form inputs for data submission
- href attribute on links
- src attribute on images, videos, audio
- datetime attribute on time elements
- target="_blank" with rel="noopener noreferrer" for external links

5. HTML Comments: Include meaningful comments:

- Major sections with opening comment: "========== HEADER SECTION =========="
- Complex structures explained
- End of major sections: "========== END HEADER =========="

D) Typography and Content Consistency (3 points)

1. Consistent Font Declaration (1 point): Specify the same font across all pages using inline style attribute on body element, such as style="font-family: Arial, Helvetica, sans-serif;" or on specific elements

2. Proper Heading Hierarchy (1 point):

- One h1 element per page for main page title
- h2 elements for major sections
- h3 elements for subsections
- h4, h5, h6 elements for deeper nesting
- Never skip heading levels, do not go from h2 to h4 without h3

3. Semantic Text Formatting (1 point): Use semantic text elements appropriately:

- strong element for important text, NOT b element
- em element for emphasized text, NOT i element
- mark element for highlighted text
- small element for fine print or legal text
- abbr element with title attribute for abbreviations
- code element for inline code snippets
- blockquote element with cite element for quotations
- pre element for preformatted text

4. Paragraph Usage: All text content must be wrapped in p paragraph tags, except headings, lists, and other specific elements

2.2.4. Part 4: Calculator Implementation (5 points)

Create a functional calculator interface using only HTML5 forms to demonstrate advanced form handling capabilities.

Implementation Requirements:

Create a new page "calculator.html" that contains a scientific calculator with the following operations and features:

- Basic operations: add, subtract, multiply, divide
- Advanced operations: power, square root, percentage
- Memory functions: store, recall, clear using buttons with name and value attributes
- History display using ordered list
- Multiple input validation
- Display area using readonly input
- Number input field with type number
- Operations groups using fieldset elements with legend elements
- Hidden fields for memory value and last result
- Control buttons for clear all and equals
- Calculation history section using fieldset with legend and ordered list

Calculator Result Page: calculator-result.html

Must display:

- main element containing h1 heading "Calculation Result"
- section element with h2 "Your Calculation:"
- dl definition list showing: First Number, Operation, Second Number, Result - all using dt

and dd elements
- paragraph with link back to calculator.html
- paragraph with link to homepage

Requirements for Full mark:
- Proper form structure with fieldsets and legends
- All inputs properly validated with required and other attributes
- Semantic HTML throughout
- Result page displays inputs and calculated result
- Clear user interface
- Accessibility features: labels, ARIA where needed

Note: Since we're only using HTML5, the calculator will submit to a result page that displays the input values. Full functionality would require JavaScript or server-side processing.

2.2.5. Part 5: Validation and Quality Assurance (5 points)

2.2.5.1. HTML5 Validation (3 points)

All pages must pass W3C Markup Validation:

Validation Process:

1. Visit W3C Markup Validation Service at https://validator.w3.org/
2. For each HTML page choose one option:
   - Option A: Upload file directly
   - Option B: Copy and paste HTML code
   - Option C: Validate by URL if deployed
3. Fix ALL errors and warnings
4. Re-validate until you get "Document checking completed. No errors or warnings to show."
5. Take screenshot of successful validation for each page

Create Validation Report PDF Including:

Page-by-Page Validation Results: For each page document:
- Page name such as index.html
- Status: Valid HTML5
- Errors: 0
- Warnings: 0
- Screenshot filename

Documentation Requirements:
- Screenshot of each page's validation result
- List any warnings you received and how you addressed them
- Confirmation statement: "All pages validated with zero errors and zero warnings"

2.2.5.2. Browser Compatibility Testing (2 points)

Test all pages in minimum 2 browsers from this list:
- Google Chrome latest version
- Mozilla Firefox latest version
- Microsoft Edge latest version
- Safari if available on macOS

Testing Checklist for Each Browser:
- Page loads correctly
- All images display properly

- All links work: internal, external, anchors, mailto
- Forms display correctly
- Form validation works
- Registration form submits to thank you page
- Navigation menu is accessible
- Breadcrumb links work
- Tables display properly
- Text is readable and properly formatted
- No console errors in Developer Tools

Create Browser Testing Report PDF Including:
Browser Compatibility Table: Create table with columns for Page Name and each browser tested, showing Pass or Fail status for each combination
Documentation:

- Browser name and version number tested
- Operating system used for testing
- Screenshots of homepage in each browser
- Any rendering differences noted and resolved
- Confirmation that all interactive elements work
- Confirmation that forms function properly

Testing Tips:

- Test on both Windows and macOS if possible
- Check responsive behavior by resizing browser window
- Test keyboard navigation using Tab key through forms
- Verify all form validations trigger correctly
- Check that external links open in new tabs

Issues to Document: If you encounter any issues:

1. Describe the problem
2. Which browsers affected
3. How you resolved it
4. Screenshot of the issue before fix
5. Screenshot after resolution

Final Confirmation Statement: All pages have been tested in [Browser 1] version X.X and [Browser 2] version X.X on [Operating System]. All functionality works as expected with no critical issues.

2.3. Lab Component (Individual Assessment) (20 points)

Conducted During Week 11 Labs

Task 1: Live Performance Analysis Presentation (5 points)

1. **Live Performance Testing Demonstration**
   - Open Google PageSpeed Insights
   - Run live test on one of your selected websites
   - Show the Core Web Vitals results as they appear
   - Explain each metric in your own words
2. **Comparison with Pre-recorded Data**
   - Compare live results with your submitted report data
   - Explain any significant differences such as time of day, server load, etc.

- Discuss what factors might cause variations

3. **Q&A Session**
   - Answer instructor questions about metric meanings and targets, your analysis methodology, and recommendation priorities

Task 2: Live Code Modification (12 points)

Instructor will ask you to perform ONE task.

Possible Task Examples:

1. Redeveloped one of the assignment web pages.
2. Modify one of the pages and convert div-based structure to semantic HTML: Instructor shows div-heavy code, You must replace divs with appropriate semantic elements, Maintain same content, improve structure.
3. Modify one of the pages and add proper breadcrumb navigation to a page that's missing it: Create breadcrumb nav with proper links, Use appropriate HTML structure, Position it correctly in page.
4. Modify one of the pages and add anchor links for in-page navigation: Create navigation menu with 3 anchor links, Link to 3 sections on the page, Add "back to top" link.
5. Develop one page containing HTML5 table: Instructor shows table structure; you must develop the semantic Structural Layout that meets the requirements.
6. Develop one page containing HTML5 FORM: Instructor shows FORM structure; you must develop the semantic FORM that meets the requirements.

Task 3: Push the Live Code on GitHub (3 points)

- Correct Repository Setup.
- Proper Commit Message
- Successful Push with Verification

3. Marking schema:

3.1. Mark Distribution Overview

Total Assignment Value: 10% of the whole course mark

Grade Distribution:

- Group Work (Submitted Assignment): 80% of assignment grade (8% of course)
- Individual Assessment (Week 11 Lab): 20% of assignment grade (2% of course)

3.2. Group Work Breakdown (80 points)

Component	Points	Weight
Part 1: Project Setup and Structure	10	12.5%
- Logical Folder Structure	5	6.25%
- Website Map and Navigation Structure	5	6.25%
Part 2: Core Web Vitals Performance Analysis	20	25%
- Website Selection & Technology Analysis	6	7.5%
- Core Web Vitals Measurement & Analysis	10	12.5%
- Comparative Analysis & Optimization Strategy	4	5%
Part 3: Complete Website Development	45	62.5%
- Page 1: Homepage (index.html)	8	10%
- Page 2: About Us (about.html)	5	6.25%
- Page 3: Contact Me (contact.html)	4	5%
- Page 4: Registration (registration.html)	10	12.5%
- Page 5: Thank You (thank.html)	3	3.75%
- Page 6: Course Catalog (course-catalog.html)	8	10%
- Page 7: Course Detail (course-detail.html)	4	5%
- Page 8: Student Dashboard (student-dashboard.html)	3	3.75%
- Universal Page Requirements	15	18.75%
• HTML5 Semantic Structure	5	6.25%
• Consistent Navigation Structure	3	3.75%
• HTML5 Code Quality Standards	4	5%
• Typography and Content Consistency	3	3.75%
Part 4: Calculator Implementation	5	6.25%
Part 5: Validation and Quality Assurance	5	6.25%
- HTML5 Validation	3	3.75%
- Browser Compatibility Testing	2	2.5%
TOTAL GROUP WORK	**100**	**100%**

3.3. Individual Assessment Breakdown (20 points)

Component	Points	Weight	Duration
Task 1: Live Performance Analysis Presentation	5	25%	10-12 min
- Live Performance Testing Demonstration	2	10%	4-5 min
- Comparison with Pre-recorded Data	2	10%	3-4 min
- Q&A Session	1	5%	3-4 min
Task 2: Live Code Modification	12	60%	40-45 min
- Understanding of Requirements	3	15%	-
- Code Implementation Quality	5	25%	-
- Semantic HTML Usage	2	10%	-
- Code Explanation Ability	2	10%	-
Task 3: GitHub Push	3	15%	3-5 min
- Correct Repository Setup	1	5%	-
- Proper Commit Message	1	5%	-
- Successful Push with Verification	1	5%	-
TOTAL INDIVIDUAL ASSESSMENT	**20**	**100%**	**~60 min**

3.4. Marking Criteria and Standards

3.4.1. Assessment Principles

Intended Learning Outcomes (ILOs) Assessed:

- ILO 3.a.1: Apply HTML5 semantic elements appropriately
- ILO 3.a.2: Implement accessible web forms with validation
- ILO 3.a.3: Analyze web performance using Core Web Vitals
- ILO 3.b.1: Structure information architecture for web applications
- ILO 3.c.1: Validate HTML5 code against W3C standards
- ILO 3.c.3: Test cross-browser compatibility
- ILO 3.d.1: Document technical work and AI tool usage

3.4.2. Mark Award Criteria

Pro-rata Marking: Marks will be awarded proportionally based on:

- Detail and completeness offered
- Evidence of academic ability and technical skills
- Professionalism in presentation and organization
- Quality of documentation and comments
- Understanding demonstrated in individual assessment

3.4.3. Penalty Provisions

Inadequate Evidence of Thoroughness:

- **40% reduction** applied to individual element score
- Applies when work lacks necessary detail, testing, or documentation
- Applied per element, not to entire assignment

Correct Approach but Implementation Errors:

- Maximum **70%** awarded for affected element
- Applies when understanding is evident but execution has mistakes
- Demonstrates partial mastery of concepts

Academic Integrity Violations:

- **Plagiarism or code copying: 0% for entire assignment for both group members**
- Undocumented AI usage: -2 points from group work
- Misrepresentation of work distribution: -2 points from group work
- Reported to university administration per policy

Late Submission Penalties:

- 0-24 hours late: -50% of group work grade
- 24-48 hours late: -75% of group work grade
- More than 48 hours: 0% without prior documented approval and individual assessment cannot be made up (absence = 0%)

Failure to Attend Lab (Week 11):

- Automatic 0% for Individual Assessment component (20 points)
- No make-up opportunity except documented emergency
- Must be reported BEFORE scheduled lab time

4. Feedback given to students in response to assessed work.

- Specific oral feedback on the assessed components.
- Feedback for the HTML will be placed on the coursework assessment sheet returned with the coursework mark.
- During contact hours, students will receive oral generic comments on every part of their assignment..
- If students need more input, they are encouraged to speak with the teaching staff.

5. Deliverables and Submission Requirements

All groups must submit two synchronized copies of their work:
On CANVAS (zipped folder)
On GitHub (private repository)
Failure to submit on both platforms = See Section 3.4.3 for penalties.

CANVAS Submission Requirements

Complete Project Folder (Zipped)
Named: GroupNumber_StudentID1_StudentID2_Assignment1.zip
Example: Group05_2021170123_2021170456_CSE211_Ass1.zip
Required Folder Structure:

```
GroupNumber_StudentID1_StudentID2_ CSE211_Ass1/
├── Part1_Performance_Analysis/
│   ├── Performance_Analysis_Report.pdf
│   └── screenshots/
│       ├── website1_pagespeed_mobile.png
│       ├── website1_pagespeed_desktop.png
│       ├── website2_pagespeed_mobile.png
│       ├── website2_pagespeed_desktop.png
│       ├── website3_pagespeed_mobile.png
│       └── website3_pagespeed_desktop.png
├── Part2_Website_Project/
│   ├── index.html
│   ├── pages/
│   │   ├── about.html
│   │   ├── contact.html
│   │   ├── registration.html
│   │   ├── thank.html
│   │   ├── course-catalog.html
│   │   ├── course-detail.html
│   │   └── student-dashboard.html
│   ├── images/
│   │   ├── logo.png
│   │   ├── courses/
│   │   └── profile/
│   ├── videos/
│   │   └── intro/
│   ├── css/
│   │   └── (empty - for future use)
│   └── scripts/
│       └── (empty - for future use)
├── Part3_Documentation/
│   ├── Website_Map.pdf
│   ├── Validation_Report.pdf
│   ├── Browser_Testing_Report.pdf
│   └── AI_Tools_Usage_Documentation.pdf
├── Part4_Calculator/
│   ├── calculator.html
│   └── calculator-result.html
└── README.txt
```

---------- END ----------