

HTML & CSS Assignment – Week 1

Assignment Questions (20 Questions)

Section 1:

Question 1: Create an HTML page with a proper document structure that includes a navigation menu with 5 links (Home, About, Services, Portfolio, Contact). Use semantic HTML5 elements appropriately.

Question 2: Build a data table that displays information about 5 different products with columns for Product Name, Category, Price, Stock Status, and Actions. Include proper table headers.

Question 3: Create a registration form with the following fields: Username, Email, Password, Confirm Password, Date of Birth, Gender (radio buttons), Country (dropdown), Terms & Conditions (checkbox), and a Submit button. Use appropriate input types and include placeholder text.

Question 4: Design an article page using semantic HTML5 elements (article, header, footer, aside, section). The article should have a title, author info, publication date, main content with at least 3 paragraphs, and a sidebar with related links.

Question 5: Create an ordered list of your top 5 favourite movies, and within each movie item, create an unordered list showing 3 reasons why you like it. Demonstrate nested lists properly.

Question 6: Build a contact page footer that includes three columns: Company Info (with address, phone number, email using address tag), Quick Links (list of links), and social media (list of social media links). Use semantic elements appropriately.

Question 7: Create an HTML page with at least 6 different heading levels (h1-h6) organized in a hierarchical structure that represents a book's table of contents.

Section 2: CSS Styling Basics

Question 8: Style a button with the following specifications: custom background color, white text, padding, rounded corners and appropriate margins.

Question 9: Create three div boxes demonstrating the CSS Box Model. Each box should have different margin, padding, and border values. Display the actual measurements by adding text inside each box explaining its box model properties.

Question 10: Design a card component (300px × 400px) with an image at the top, a title, a description, and a button at the bottom. Use appropriate backgrounds, borders, padding,

and margins. The card should have a subtle border and apply a different border style other than solid.

Question 11: Create a webpage with three different CSS styling methods: inline CSS for one element, internal CSS for another section, and external CSS for the rest of the page. Document which method you used where in comments.

Question 12: Style a navigation menu where links change color on hover, have a different color when visited, and display with custom fonts. Use at least 3 different font properties (font-family, font-size, font-weight or any other 3 of your preferences).

Section 3: Layout & Positioning

Question 13: Create a webpage header that remains fixed at the top of the page while scrolling. The header should contain a logo on the left and navigation links on the right. Use appropriate positioning.

Question 14: Design a layout with a sticky sidebar on the left (200px width) that stays visible while scrolling, and main content on the right that scrolls normally. Demonstrate the difference between fixed and sticky positioning.

Question 15: Create four 150x150 boxes: move the relative box 20px right/down, place the absolute box at the bottom-right of its parent, and keep the fixed box stuck to the bottom-left of the screen while scrolling.

Question 16: Create a navigation menu once using display:inline (all links in one row) and again using display:block (links stacked vertically), showing the difference clearly.

Section 4: Flexbox & Grid

Question 17: Create a photo gallery using CSS Flexbox with 6 images. The gallery should:

- Display 3 images per row on desktop
- Center the images horizontally and vertically within their containers
- Have equal spacing between images
- Include links on images and it should navigate to Wikipedia where the information shall be available

Question 18: Design a dashboard layout using CSS Grid with the following sections:

- Header (spanning full width)
- Sidebar (left side, 1/4 width)

- Main content area (center, 2/4 width)
- Widget panel (right side, 1/4 width)
- Footer (spanning full width)

Use grid-template-columns/rows for the layout.

Section 5: Responsive Design

Question 19: Create a responsive navigation menu that:

- Displays horizontally on desktop (screens wider than 768px)
- Converts to a vertical stacked menu on mobile (screens 768px or smaller)
- Changes the background color at the medium breakpoint (768px-1024px)

Use media queries to achieve this.

Question 20: Build a responsive card grid that displays:

- 4 cards per row on large screens (>1200px)
- 3 cards per row on medium screens (768px-1200px)
- 2 cards per row on small screens (480px-768px)
- 1 card per row on mobile (<480px)

Use either Flexbox or Grid, and implement with media queries.

FINAL PROJECT: Personal Portfolio Website

Project Overview

Create a fully responsive personal portfolio website that showcases your skills, projects, and contact information. This project will combine everything you've learned over the past 5 days.

Pages Required (Minimum 3 pages)

1. Home Page (index.html)
2. Projects/Portfolio Page (projects.html)
3. Contact Page (contact.html)

Technical Requirements

HTML Requirements:

- Proper HTML5 document structure on all pages (DOCTYPE, html, head, body)
- Use of semantic elements (header, nav, main, section, article, footer, aside)
- A navigation menu that appears on all pages linking to each page
- At least one properly structured table on Projects page showing project details
- A contact form with validation attributes (required, email type, etc.)
- Organized use of headings (h1-h6) maintaining proper hierarchy
- Both ordered and unordered lists where appropriate
- Proper use of links, including external links opening in new tabs

CSS Requirements:

- External CSS stylesheet(s) linked to all pages
- Demonstration of the Box Model (proper use of margin, padding, border)
- Custom color scheme (minimum 3-4 colors used consistently throughout)
- Custom fonts (use at least 2 different font families)
- Box shadows and border-radius for modern styling
- Use of different background properties (colours, gradients)
- Proper use of CSS selectors (class, ID)

Layout Requirements:

- **Fixed/Sticky Navigation:** Header/navigation bar that stays visible while scrolling
- **Flexbox Implementation:** Use Flexbox for at least 2 sections Examples: navigation menu, skills section, footer, image gallery
- **CSS Grid Implementation:** Use CSS Grid for the projects gallery (minimum 6 project cards)
- **Positioning:** Demonstrate at least 2 different positioning types appropriately (relative, absolute, fixed, or sticky)

Responsive Design Requirements (MANDATORY):

Must be fully responsive with media queries for THREE breakpoints:

- **Mobile:** Less than 768px

- **Tablet:** 768px to 1024px
- **Desktop:** Greater than 1024px

The layout must adapt gracefully:

- Navigation converts to mobile-friendly vertical layout
 - Grid/Flex layouts adjust column counts based on screen size
 - Font sizes scale appropriately for readability
 - Images resize properly and maintain aspect ratio
 - Padding and margins adjust for smaller screens
-

Content Guidelines:

Home Page Must Include:

- Hero section with your name, photo/image, and a tagline/introduction
- About Me section with a paragraph about yourself
- Skills section displaying your technical skills (use Flexbox or Grid for layout)
- Education or Experience section (use a table or structured layout)
- Call-to-action button or link directing to projects or contact page

Projects Page Must Include:

- Page heading and brief introduction
- Minimum 6 project cards displayed in a responsive CSS Grid
- Each project card must contain:
 - Project image or placeholder
 - Project title
 - Brief description (2-3 sentences)
 - Technologies/skills used (displayed as a list or tags)
 - Links or buttons (even if they are placeholder links)
- A table listing all projects with columns: Project Name, Category, Technologies, Completion Date

Contact Page Must Include:

- Page heading
- Contact form with the following fields:
 - Full Name (text input, required)
 - Email Address (email input, required)
 - Phone Number (tel input, optional)
 - Subject (text input, required)
 - Message (textarea, required)
 - Submit button
- Your contact information section displaying:
 - Email address
 - Phone number
 - Location/Address
- Social media links (LinkedIn, GitHub, etc.) with proper tags
- All form inputs must have appropriate labels and placeholder text

All Pages Must Have:

- Consistent header with logo/name and navigation menu
- Consistent footer with:
 - Copyright information
 - Social media links
 - Additional navigation links (optional)
- Attractive design with matching color scheme and typography
- Proper spacing and alignment throughout