## Beginner-Level HTML Exercises

1. Create a Basic HTML Page: Create a simple webpage with the title "My First Webpage" and an h1 heading with the text "Hello, World!".

2. Add Paragraphs: Add three paragraphs to the webpage, each containing different text.

3. Insert an Image: Add an image to your webpage using an image URL and provide alternate text (alt) for the image.

4. Create a List: Create an unordered list with three items: "HTML", "CSS", and "JavaScript".

5. Add a Link: Add a hyperlink to your webpage that links to "https://www.example.com".

6. Use HTML Comments: Add a comment at the top of your HTML document explaining its purpose.

7. Add a Table: Create a table with 2 rows and 2 columns containing sample data.

8. Create a Button: Add a button to your webpage with the text "Click Me!".

9. Format Text: Use bold (`<b>`) and italic (`<i>`) tags to format certain words in a paragraph.

10. Create a Form: Create a simple form with an input field for "Name" and a "Submit" button.

11. Add Metadata: Add meta tags to the HTML document to specify the character set (UTF-8) and viewport settings for responsive design.

12. Create a Header and Footer: Create a simple header with an h1 tag and a footer with an h3 tag.

13. Add a Line Break: Use the `<br>` tag to break a line in a paragraph.

14. Create an Ordered List: Create an ordered list with five items, such as steps in a recipe.

15. Create a Navigation Menu: Add a simple navigation menu using an unordered list with links.

16. Use the `<strong>` Tag: Add a strong emphasis to some text within a paragraph.

17. Add a Horizontal Line: Use the `<hr>` tag to add a horizontal line between two sections.

18. Set the Language: Set the language of the document to English using the `lang` attribute.

19. Create a Definition List: Create a definition list with three terms and their corresponding definitions.

20. Add a Video: Embed a video from a URL into your webpage using the `<video>` tag.

---

## Intermediate Level HTML Exercises

1. Create a Multipage Website: Build a basic website with three pages: Home, About, and Contact. Add navigation links to switch between these pages.

2. Create a Contact Form: Create a contact form with fields for Name, Email, Subject, and Message. Include a "Send" button.

3. Use Media Queries: Add media queries to change the background color of the webpage when viewed on devices with a width less than 600px.

4. Add a Video Player: Embed a YouTube video on your page with autoplay and controls enabled.

5. Create a Responsive Image Gallery: Use the `<figure>` and `<figcaption>` elements to create a responsive image gallery with at least four images.

6. Create a Dropdown Menu: Implement a simple dropdown menu using `<select>` and `<option>` tags.

7. Use the `<fieldset>` and `<legend>` Tags: Organize a form using `<fieldset>` and `<legend>` for better accessibility and grouping.

8. Add Audio to Your Webpage: Embed an audio file with controls using the `<audio>` tag.

9. Create a Table with a Caption: Create a table with three columns and four rows. Add a caption at the top of the table.

10. Use the `<iframe>` Tag: Embed a Google Map into your webpage using the `<iframe>` tag.

11. Implement a Basic Layout Using Divs: Create a basic layout with a header, a main content area, and a footer using `<div>` elements.

12. Add Semantic HTML5 Elements: Redesign an existing webpage using semantic HTML5 elements like `<header>`, `<nav>`, `<section>`, `<article>`, and `<footer>`.

13. Create an Accessible Form: Add appropriate labels and `aria` attributes to make a form accessible.

14. Create a Progress Bar: Implement a simple progress bar using the `<progress>` element.

15. Add a Download Link: Create a link that allows users to download a file directly.

16. Use CSS to Style an HTML Document: Add an internal stylesheet to style various HTML elements, such as changing text color and font size.

17. Create a Newsletter Signup Form: Build a form for a newsletter signup with fields for Name and Email and a submit button.

18. Implement a Modal Popup: Create a basic modal popup that appears when a button is clicked.

19. Add Custom Icons Using Fonts: Use FontAwesome or a similar icon library to add icons to your webpage.

20. Use the `<canvas>` Tag: Draw a simple shape (like a rectangle or circle) using the `<canvas>` tag.

## Advanced Level HTML Exercises

1. Create a Full E-commerce Product Page: Develop a fully functional product page for an e-commerce site, including product images, descriptions, pricing, and an "Add to Cart" button.

2. Build a Responsive Navigation Bar: Design a fully responsive navigation bar that collapses into a hamburger menu on smaller screens.

3. Develop a Blog Layout Using Grid or Flexbox: Create a blog layout using CSS Grid or Flexbox with a main content area and a sidebar.

4. Implement an Accessible Accordion: Create an accessible accordion component using HTML, CSS, and JavaScript, with appropriate `aria` roles and attributes.

5. Create a Multi-Step Form: Develop a multi-step form that progresses through several steps before final submission.

6. Build a Custom Video Player: Create a custom video player using HTML5 `<video>` and JavaScript to handle play, pause, and volume controls.

7. Create an Interactive SVG Graphic: Embed an SVG graphic and use JavaScript to make it interactive (e.g., changing colors on hover).

8. Develop a Client-side Validation Form: Implement a form with client-side validation using HTML5 input types and JavaScript.

9. Implement a CSS-based Loading Animation: Create a custom loading animation using only HTML and CSS.

10. Create a Sticky Footer Layout: Build a webpage with a sticky footer that remains at the bottom of the page regardless of content height.

11. Build a Parallax Scrolling Effect: Implement a parallax scrolling effect for background images using HTML and CSS.

12. Design a Webpage with Multiple Sections: Create a single-page application with multiple sections that can be navigated using a sticky sidebar.

13. Create a Complex Form with Validation and Error Handling: Develop a form with complex input types (such as date, range, color) and custom error messages.

14. Implement a Dark Mode Toggle: Add a button to your webpage that toggles between light and dark mode themes using CSS variables.

15. Create a Progressive Web App (PWA): Build a simple PWA with offline capabilities and a manifest file.

16. Build a Dashboard with Charts and Graphs: Use HTML, CSS, and JavaScript libraries like Chart.js to create a dynamic dashboard with charts and graphs.

17. Create a Custom File Upload Interface: Develop a custom file upload button that displays the selected file's name.

18. Implement Infinite Scrolling: Create a webpage that loads more content automatically as the user scrolls down.

19. Develop a Website with Localization Support: Build a multilingual website using the `<lang>` attribute and JavaScript for switching languages.

20. Create an Accessible Drag-and-Drop Interface: Develop a drag-and-drop interface with accessibility support using `aria` attributes and JavaScript.

These exercises will help you practice and improve your HTML skills at various levels of difficulty.