HTML EXERCISES:

`index.html`

1. Create a new file and save the file with a `.html` extension, such as `index.html`.

2. Start with the basic HTML structure.

`<!DOCTYPE html>`: Declares the document type.
`<html>`: Root element of an HTML document.
`<head>`: Contains metadata about the document.
`<title>`: Sets the title displayed in the browser's tab.
`<body>`: Contains the visible content of the document.

CODE:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Web Overview</title>
</head>
<body>
</body>
</html>
```

3. Create the header area with navigation.

`<header>`: Defines a header for a document or section.
`<h1>`: Defines a heading.
`<nav>`: Defines navigation links.
`<ul>`: Defines an unordered list.
`<li>`: Defines a list item.
`<a>`: Defines a hyperlink.

CODE:

```
<header>
    <h1>Your Name</h1>
    <nav>
        <ul>
            <li><a href="index.html">Home</a></li>
            <li><a href="about.html">About</a></li>
            <li><a href="contact.html">Contact</a></li>
```

```
        </ul>
    </nav>
</header>
```

4. Create the main content section. Use semantic tags, a paragraph, and a list to list the key aspects of HTML5.

CODE:

```
<main>

    <section>
        <h2>HTML5: The Backbone of the Web</h2>
        <article>
            <h3>What is HTML5?</h3>
            <p>HTML5 is the latest standard for structuring
and presenting content on the web. It provides a rich set
of elements and attributes to create semantic and
accessible web pages.</p>
            <h3>Key Features of HTML5</h3>
            <ul>
                <li>Semantic Elements: For better structure
and SEO.</li>
                <li>Multimedia Support: For embedding audio
and video directly.</li>
                <li>Canvas and SVG: For creating graphics
and animations.</li>
                <li>Local Storage: For storing data client-
side.</li>
                <li>Form Input Types: For enhanced form
functionality.</li>
            </ul>
        </article>
    </section>

    </main>
```

5. Create the aside section, `<aside>` Defines content aside from the page content.

CODE:

```
<aside>
```

```
    <h3>Related Content</h3>
    <ul>
        <li><a href="locations.html">Conference Locations</
a></li>
        <li><a href="cart.html">Shop</a></li>
    </ul>
</aside>
```

6. Create the footer.

CODE:

```
<footer>

    <p>&copy; 2024</p>
</footer>
```

7. Save and View:

- Save the file and open it in a web browser to see the structure.

cart.html

1. Add the basic HTML structure - you can copy it from the previous page.

CODE:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Shopping Cart</title>
</head>
<body>
</body>
</html>
```

2. Create the header.

`<header>`: Defines a header for a document or section.
`<h1>`: Defines a level-one heading.

CODE:

```
<header>
```

```
    <h1>Shopping Cart</h1>
</header>
```

3. Create the main content section.

`<main>`: Defines the dominant content of the document.
`<section>`: Defines a thematic grouping of content.
`<h2>`: Defines a level-two heading.
`<ul>`: Defines an unordered list (empty for now).

CODE:

```
<main>
    <section>
        <h2>Products</h2>
        <ul></ul>
    </section>

    <section>
        <h2>Cart</h2>
        <header>
            <span>0</span> Product(s) in Cart
        </header>
        <ul></ul>
        <footer>
            <button>Empty Cart</button>
        </footer>
    </section>
</main>
```

4. Save and view.

- This is the basic HTML structure. Later we will add JavaScript to dynamically populate the product and cart sections, handle user interactions (adding/removing items, updating quantities), and update the cart total.
- We'll also add CSS to style the layout, colors, fonts, and overall appearance of the shopping cart.

`locations.html`

1. Start with the basic HTML structure. Add the main section and image gallery section. The images are in an `img` folder in the HTML5Solutions folder, and are `austin.jpg`, `berlin.jpg` and `singapore.jpg`. Use `figure` and `figcaption`. At the bottom, put links in a nav tag for previous and next. We will

direct them later with JavaScript, so you can leave the href blank. At the very bottom put three empty `span` tags in a `div`. We will style dots with CSS later. Save and view the page once you are finished.

`<main>`: Defines the dominant content of the document.
`<section>`: Defines a thematic grouping of content.
`<figure>`: Defines self-contained content, often with a caption.
`<figcaption>`: Defines a caption for a figure.
`<nav>`: Defines navigational links.
`<div>`: Defines a generic container for grouping elements.
`<span>`: Defines an inline container for text or other inline elements.

CODE:

```
<main>
    <section>
        <figure>
            <img src="img/austin.jpg" alt="Austin">
            <figcaption>Austin</figcaption>
        </figure>

        <figure>
            <img src="img/berlin.jpg" alt="Berlin">
            <figcaption>Berlin</figcaption>
        </figure>

        <figure>
            <img src="img/singapore.jpg" alt="Singapore">
            <figcaption>Singapore</figcaption>
        </figure>
    </section>

    <nav>
        <a href="#">Previous</a>
        <a href="#">Next</a>
    </nav>

    <div>
        <span></span>
        <span></span>
        <span></span>
    </div>
```

```
</main>
```

about.html

1. We will create a page with the basic structure, header and navigation. This will have an input field for text and a placeholder. We will also add a table with several rows (only one row is pictured below), and two fields each - one for the HTML5 tag you remember and the other for the description of that tag.

<table>: Defines a table.
<thead>: Defines a table header.
<tr>: Defines a table row.
<th>: Defines a table header cell.
<tbody>: Defines a table body.
<tr>: Defines a table row.
<td>: Defines a table data cell.
Optional:
< and > are character entities for angle brackets.
<code>: Makes the content monospaced.

CODE:

```
<main>
    <section>
        <input type="text" id="search" placeholder =
"Search...">
    </section>
    <section>
        <table>
            <thead>
                <tr>
                    <th>Tag</th>
                    <th>Description</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td><code>&lt;header&gt;</code></td>
                    <td>Header for doc or section.</td>
                </tr>
                </tbody>
```

```
        </table>
    </section>
</main>
```

contact.html

1. **Start with the basic structure and navigation, which you can copy. We will make a contact form. Save and test when finished.**

2. **Create the contact form.**

`<form>` defines the form element.
`<label>` defines labels for each input field.
`<input>` defines various types of input fields for user information.
- `type="text"` for single-line text input (name, street, city).
- `type="email"` for email addresses.
- `type="tel"` for phone numbers.
- `type="number"` for numeric input (age) with `min` and `max` attributes for range.
- `type="date"` for selecting dates, with a `max` attribute set.
- `type="text"` for zip code with a `pattern` attribute to restrict input to 5 digits.

`<button>` or `<input type="submit">` defines a submit button.
Optional attributes:
`placeholder` for placeholder text.
`pattern` to add a regular expression.
`required` to ensure the field is filled out - it saves on a lot of JavaScript code.
`min` and `max` for min and max value in number and date fields.
`id` to give each field a unique identifier to get the data later.

CODE:

```
<form>
    <label for="name">Name:</label>
    <input type="text" id="name" name="name"
placeholder="Your Name" required>

    <label for="email">Email:</label>
    <input type="email" id="email" name="email"
placeholder="Your Email" required>

    <label for="phone">Phone Number:</label>
    <input type="tel" id="phone" name="phone"
placeholder="Your Phone Number">
```

```
    <label for="age">Age:</label>
    <input type="number" id="age" name="age" min="18"
max="100" placeholder="Your Age">

    <label for="birthdate">Birthdate:</label>
    <input type="date" id="birthdate" name="birthdate"
max="2005-12-31">

    <label for="street">Street:</label>
    <input type="text" id="street" name="street"
placeholder="Street Address">

    <label for="city">City:</label>
    <input type="text" id="city" name="city"
placeholder="City">

    <label for="zip">Zip Code:</label>
    <input type="text" id="zip" name="zip" pattern="[0-9]
{5}" placeholder="Zip Code">

    <button type="submit">Submit</button>
</form>
```