Module 6: HTML Table

6.1 Creating Tables

HTML tables allow web developers to arrange data into rows and columns. A table in HTML consists of table cells inside rows and columns.

The <table> tag defines an HTML table.

An HTML table consists of one <table> element and one or more <tr>, <th>, and <td> elements.

The <tr> element defines a table row, the <th> element defines a table header, and the <td> element defines a table cell.

An HTML table may also include <caption>, <colgroup>, <thead>, <tfoot>, and <tbody> elements.

6.2 Table Header, Body and Footer

Table Header:

The <thead> tag is used to group header content in an HTML table.

The <thead> element is used in conjunction with the <tbody> and <tfoot> elements to specify each part of a table (header, body, footer).

The <thead> element must have one or more <tr> tags inside.

The <thead> tag must be used in the following context: As a child of a <table> element, after any <caption> and <colgroup> elements, and before any <tbody>, <tfoot>, and <tr> elements.

Table Body:

The <tbody> tag is used to group the body content in an HTML table.

The <tbody> element is used in conjunction with the <thead> and <tfoot> elements to specify each part of a table (body, header, footer).

The <tbody> element must have one or more <tr> tags inside.

The <tbody> tag must be used in the following context: As a child of a <table> element, after any <caption>, <colgroup>, and <thead> elements.

Table Footer:

The <tfoot> tag is used to group footer content in an HTML table.

The <tfoot> element is used in conjunction with the <thead> and <tbody> elements to specify each part of a table (footer, header, body).

The <tfoot> element must have one or more <tr> tags inside.

The <tfoot> tag must be used in the following context: As a child of a <table> element, after any <caption>, <colgroup>, <thead>, and <tbody> elements

6.3 Table Rows and Cells

Table Row:

The <tr> tag defines a row in an HTML table

A <tr> element contains one or more <th> or <td> elements.

Table Cells:

An HTML table has two kinds of cells:

Header cells - contains header information (created with the <th> element)
Data cells - contains data (created with the <td> element)

6.4 Table Borders and Styles

Table Borders:

to add a border, we can use the CSS border property on table, th, and td elements:

To avoid having double borders, set the CSS border-collapse property to collapse.

With the border-radius property, the borders get rounded corners:

With the border-style property, we can set the appearance of the border.
With the border-color property, you can set the color of the border.

Table Styles:

To style every other table row element, use the :nth-child(even)

To make vertical zebra stripes, style every other column, instead of every other row.
Set the :nth-child(even) for table data elements:

Use the :hover selector on tr to highlight table rows on mouse over:

6.5 Advanced Table and Features

Adding a caption to our table with <caption>:

we can give our table a caption by putting it inside a <caption> element and nesting that inside the <table> element. we should put it just below the opening <table> tag.

Adding structure with <thead>, <tbody>, <tfoot>:

As our tables get a bit more complex in structure, it is useful to give them more structural definition. One clear way to do this is by using <thead>, <tbody>, and <tfoot>, which allow us to mark up a header, body, and footer section for the table.

These elements don't make the table any more accessible to screen reader users, and don't result in any visual enhancement on their own. They are however very useful for styling and layout — acting as useful hooks for adding CSS to your table.

Using Colspan Attribute:

COLSPAN is the attribute of the <table> tag and can be used on the following elements: <td>, <th> . COLSPAN defines the number of columns a table cell can contain.

Using Rowspan Attribute:

ROWSPAN is the attribute of the <table> tag and can be used on the following elements: <td>, <th> . ROWSPAN defines the number of row a table cell can contain.

Using Id Attribute:

The ID attribute is unique, so it can model a single item on the web page. This attribute is used to indicate a style in a style sheet

Using Scope Attribute:

The SCOPE attribute indicates what is the header cell — a column, a group of columns, a row or row of rows. This attribute is defined by both <td> and <th>

Module 7: HTML5 Features

7.1 New Semantic Elements (Header, Footer, Section, Article, etc.)

Semantic HTML elements are those that clearly describe their meaning in a human- and machine-readable way.

Elements such as <header>, <footer> and <article> are all considered semantic because they accurately describe the purpose of the element and the type of content that is inside them.

Why use semantic elements?

First, it is much easier to read. This is probably the first thing we will notice when looking at the first block of code using semantic elements. This is a small example, but as a programmer we can be reading through hundreds or thousands of lines of code. The easier it is to read and understand.

It has greater accessibility. we are not the only one that finds semantic elements easier to understand. Search engines and assistive technologies (like screen readers for users with a sight impairment) are also able to better understand the context and content of our website, meaning a better experience for our users.

The semantic elements added in HTML5 are:

<section> and <article>
The <section> and <article> elements are conceptually similar and interchangeable. To decide which of these you should choose, take note of the following:

- An article is intended to be independently distributable or reusable.
- A section is a thematic grouping of content.

<header>
The <header> element is generally found at the top of a document, a section, or an article and usually contains the main heading and some navigation and search tools.

<hgroup>
The <hgroup> element should be used where we want a main heading with one or more subheadings.

The <header> element can contain any content, but the <hgroup> element can only contain other headers, that is <h1> to <h6> and including <hgroup>.

<aside>
The <aside> element is intended for content that is not part of the flow of the text.
<aside> is a sidebar to our main content.

<nav>
Before HTML5, our menus were created with 's and 's. Now, together with these, we can separate our menu items with a <nav>, for navigation between our pages.

<footer>
If there is a <header> there must be a <footer>. A <footer> is generally found at the bottom of a document, a section, or an article. Just like the <header> the content is generally metainformation, such as author details, legal information, and/or links to related information. It is also valid to include <section> elements within a footer.

<small>
The <small> element often appears within a <footer> or <aside> element which would usually contain copyright information or legal disclaimers, and other such fine print. However, this is not intended to make the text smaller. It is just describing its content, not prescribing presentation.

<time>
The <time> element allows an unambiguous ISO 8601 date to be attached to a human-readable version of that date.

<figure> and <figcaption>
<figure> is for wrapping your image content around it, and <figcaption> is to caption your image.

7.2 HTML5 Forms (New Input Types, Form Attributes)
HTML5 introduced several new input types and form attributes to enhance the functionality and user experience of web forms.

New Input Types:

Email: <input type="email"> - This type is used for input fields that require an email address. Browsers can provide validation for email addresses entered in this type of input field.

URL: <input type="url"> - This type is used for input fields that require a URL (web address). Like the email input type, browsers can provide validation for URLs entered in this type of input field.

Number: <input type="number"> - This type is used for input fields that require a numerical value. Browsers typically provide UI controls like spinners for selecting numerical values, and they may also restrict input to numbers.

Date: <input type="date"> - This type is used for input fields that require a date. Browsers provide a date picker interface for selecting dates.

Time: <input type="time"> - This type is used for input fields that require a time. Browsers provide a time picker interface for selecting times.

Color: <input type="color"> - This type is used for input fields that require a color. Browsers provide a color picker interface for selecting colors.

New Form Attributes:

autocomplete: The autocomplete attribute controls whether the browser should automatically complete the input values based on the user's previous input in similar fields. It can be set to values like "on" or "off".

novalidate: The novalidate attribute can be added to the <form> tag to disable the browser's built-in form validation. This is useful when you want to implement custom validation using JavaScript.

formaction: The formaction attribute specifies the URL where the form data should be submitted when the form is submitted. This allows you to override the action attribute of the <form> tag for a specific submit button.

formenctype and formmethod: These attributes (formenctype and formmethod) are used to specify the encoding type (enctype) and HTTP method (method) for form submissions. They allow you to override the <form> tag's attributes for a specific submit button.