HTML

<title>Title Name</title>

➜ Put a title for this webpage

<meta name = "description" content="This is the content"/>

➜ This information will be showed when searching about this webpage is a search engine results.

<!--Comment here ->

➜ To create a comment

Text Styling

<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>

➔ To create headings with difference size

<p>Paragraph here</p>

➔ Create section for paragraph

Bold Text

➔ To make text bold

Bold Text but is important

➔ The string refers for search engine that the text inside it is important.

<mark>Highlighted text</mark>

➔ To highlight the text inside it

<u>Underline Text</u>

➔ To make underline

~~Deleted Text~~

➔ Make line through text (deleted text)

H<sub>2</sub>O

➔ Make the text inside it as small text shifted to bottom. (H_2O)

n power 2 = n<sup>2</sup>

➔ Make the text inside it shifted to top (as a power of a number n^2)

Hyperlinks

`<a href = "link">text to be shown in webpage</a>`

➔ To make a hyper link

`<a href = "#paragraph1">Go to paragraph 1</a>`

➔ We can move to a specific paragraph in the same page using paragraph id

`<a href = "page2.html#p1>Go the paragraph in another page></a>`

➔ You can move to a paragraph in another page.

`<a href = "mailto:shawky.ebrahim2514@gmail.com>Send To Me</a>`

➔ When click (Send To Me) it will be redirected to the email

Target attribute

➔ Decide where to redirect to the link.
➔ By default, the link will be open in the same tab.

`<a href = "link" target="_blanck">`

➔ Open the link in a new tab.

Images

`<img src = "image path"/>`

➔ We can use the **src** as a link or a path in the system

`<img src = "../img.png" />`

➔ When using the system path and want to go to one step back from your current folder, use (../)

Alt attribute

`<img src"path" alt="this is the alternative text" />`

➔ We can use the "**alt**" attribute to replace the text with an alternative text in case of this image doesn't exist or cannot open it.

Lists

- ➔ To make an unordered list
- ➔ In the tags we use list items:

- ➔ To make a nested unordered list, we can inside the list item make another unordered list

- ➔ To make an order list

- ➔ We can use attribute (reversed) to reverse the order of this list
- ➔ We can use attribute (start = "100") to start the order of this list by a specific number.

Order the list

- ➔ By default, this list will be order as a number ordering.
- ➔ To specify the type of this order list, use attribute (type = "A") or any available type.
- ➔ We can enforce a sub list to have to ordered from a specific value

```
<li>First</li>
<li>Second</li>
<li value="60">Third</li>
<li>Fourth</li>
<li>Fifth</li>
</ol>
```

- ➔ This will make the first two items 1 and 2
- ➔ And from the third item, it will have the order = 50, and the other item below it will follow this new order.

Description List

<dl></dl>

- ➔ To create a description list, same as the unordered list, but without any bullets.
- ➔ In the description list, we define a description items and description for each item
- ➔ To create description items:
  - ▪ **<dt></dt>**
- ➔ To create description for each item:
  - ▪ **<dd></dd>**

```
<dl>
 <dt>HTML</dt>
 <dd>Hyper Text Markup Language</dd>
 <dt>CSS</dt>
 <dd>Cascading Style Sheets</dd>
 <dt>JS</dt>
 <dd>Java Script</dd>
</dl>
```

Tables

Each table can be separated into three parts or less

➔ (**thead** => table head)
➔ (**tbody** => table body)
➔ (**tfoot** => table foot)
➔ In each section we can create any number of rows using table row tag:
■ **<tr></tr>**
➔ In each table row, we can create table data cell in each column using:
■ **<td></td>**
➔ In the table head section, we can use table head cell instead of table data cell:
■ **<th></th>**
➔ We can use tables without defining any section, directly create table rows and table data cells.
➔ We can create a caption section that define the title or caption for this table.
■ **<caption></caption>**
➔ To merge two cells or more, we can use the attribute:
■ **colspan ="numberOfMergedCells"**

```
<table border="1">
 <caption>List of Employees</caption>
 <thead>
 <tr>
  <th>ID</th>
  <th>First Name</th>
  <th>Last Name</th>
  <th>Salary</th>
 </tr>
 </thead>
 <tbody>
 <tr>
  <td>1</td>
  <td>Shawky</td>
  <td>Ebrahim</td>
  <td>12000</td>
 </tr>
 <tr>
  <td>2</td>
  <td>Mohammed</td>
  <td>Essam</td>
  <td>11000</td>
 </tr>
 <tr>
  <td>3</td>
  <td>Belal</td>
  <td>Ahmed</td>
```

```
  <td>10000</td>
 </tr>
 </tbody>
 <tfoot>
 <tr>
  <td colspan="3">Total salary</td>
  <td>33000</td>
 </tr>
 </tfoot>
</table>
```

Text

➜   This is an inline object, that can put text in it, it is useful to style a specific text using CSS

➜   To make a new line using HTML

<hr/>

➜   To make a horizontal row line

<div></div>

➜   Create a container

Entities of HTML

To make the symbol (<)

➔ <

➔ Lt = less than.

To make the symbol (>)

➔ >

➔ Gt = greater than.

To make a copy right symbol

➔ ©

More Entities from here: HTML Entities (w3schools.com) HTML Symbols (w3schools.com)

HTML Good Styles

```
<!DOCTYPE html>
<html lang="en-us"/>
<head>
  <meta charset="UTF-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <title>Webpage Title</title>
</head>
<body>
</body>
</html>
```

```
<html lang="en-us"/>
```

- ➔ To declare the language of the Web page.
- ➔ This is meant to assist search engines and browsers.

```
<meta charset="UTF-8"/>
```

- ➔ To ensure proper interpretation and correct search engine indexing

```
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
```

- ➔ This gives the browser instructions on how to control the page's dimensions and scaling.

Use Lowercase Element Names

```
<p>Paagraph</p>
```

not

```
<P>Paragraph</P>
```

Close All HTML Elements

```
<html>
<head>
  <title></title>
</head>
<body>
  <p></p>
</body>
</html>
```

Use lowercase attribute names

`<img src="path"/>`

not

`<img SRC="path">`

Always quote attribute values

`<input type"text"/>`

not

`<input type=text>`

Always Specify alt, width, and height for Images

Use blank lines and identation if needed

Don't omit <html>, <head> and <body>

Close Empty HTML Elements

`<img src="path"/>`

not

`<img src="path">`

HTML Comments

```
<!-- This is a comment -->
<!--
    This is a long comment example. This is a long comment example.
    This is a long comment example. This is a long comment example.
-->
```

Use Lower Case File Names

Short CSS rules can be written compressed, like this:

```css
p.intro {font-family:Verdana;font-size:16em;}
```

Long CSS rules should be written over multiple lines:

```css
body {
  background-color: lightgrey;
  font-family: "Arial Black", Helvetica, sans-serif;
  font-size: 16em;
  color: black;
}
```

➔ Place the opening bracket on the same line as the selector
➔ Use one space before the opening bracket
➔ Use two spaces of indentation
➔ Use semicolon after each property-value pair, including the last
➔ Only use quotes around values if the value contains spaces
➔ Place the closing bracket on a new line, without leading spaces

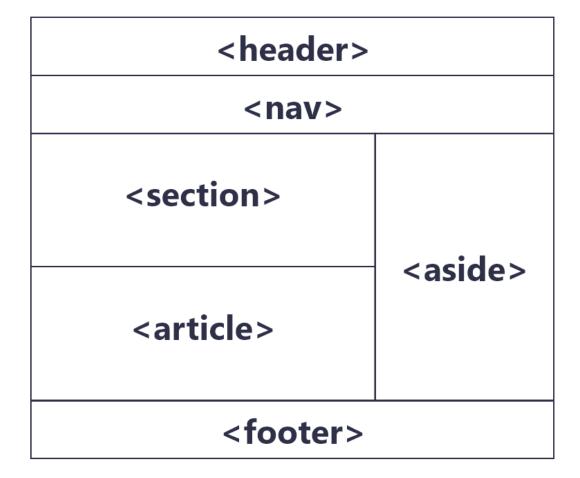More about Good HTML Styles: HTML Style Guide and Coding Conventions (w3schools.com)

shawkyebrahim2514

HTML Semantic Element

In HTML there are some semantic elements that can be used to define different parts of a web page

```
┌─────────────────────────────────────────┐
│              <header>                    │
├─────────────────────────────────────────┤
│               <nav>                      │
├──────────────────────┬──────────────────┤
│                      │                  │
│     <section>        │                  │
│                      │    <aside>       │
├──────────────────────┤                  │
│                      │                  │
│     <article>        │                  │
│                      │                  │
├──────────────────────┴──────────────────┤
│              <footer>                    │
└─────────────────────────────────────────┘
```

Define a section in a document.

<section></section>

Can be used in:
➔ Chapters
➔ Introduction
➔ News Items
➔ Contact Information

Specify independent, self-contained content.

<article></article>

Can be used in:
➔ Forum posts
➔ Blog posts
➔ User comments
➔ Product cards
➔ Newspaper articles

represent a container for introductory content or a set of navigational links.

<header></header>

contains:
➔ one or more heading elements (<h1> - <h6>)
➔ logo or icon
➔ authorship information

Define a footer for a document or section.

Contains:
- ➜ authorship information
- ➜ copyright information
- ➜ contact information
- ➜ sitemap
- ➜ back to top links
- ➜ related documents

Define a set of navigation links.

<nav></nav>

Define some content aside from the content it is placed in (like a sidebar).

<aside></aside>

For Figures

Specifie self-contained content, like illustrations, diagrams, photos, code listings, etc.

<figure></figure>

Define a caption for a <figure> element.

<figcaption></figcaption>

```
<figure>
  <figcaption>Figure Caption</figcaption>
  <img src="path">
</figure>
```

Specify details that the user can open and close on demand

`<details></details>`

➔ Is often used to create an interactive widget that the user can open and close.
➔ By default, the widget is closed. When open, it expands, and displays the content within.

`<summary></summary>`

➔ Is used in details tag to specify a visible heading for the details

HTML 5 semantic elements: HTML Semantic Elements (w3schools.com)

HTML Form

<form></form>
➜ Identify the container in which we will put the input fields in

<input />
➜ Create form field

<label></label>
➜ Create a label for a text field

Specify the type of each text field (text, password, ..etc)
➜ By using the attribute (**type="text"**) in the input tag

Make a submission button for a form

<input type="submit"/>

Enforce user to fill input field
➜ Add the attribute (**required**) in the input tag

Add a hint text to an input field
➜ Use the attribute (**placeholder = "text"**).
➜ This placeholder text cannot be selected or modified by the user.
➜ The text will disappear when the user starts typing in the field.

Make a default value for a text field
➜ Use the attribute (**value="defaultValue"**).
➜ This text can be selected and modified by the user.

Specifie where to send the form data when the user submits the form
➜ Use the attribute (**action="path"**) in the form tag

Access each field that in the form after submitting the request

➜ We must include the attribute (**name = "name"**)
➜ The name attribute is used as a reference when the data is submitted.
➜ When submitting a form, the data is packaged as a series of name-value pairs, so the name attribute is important and should never be omitted.

Specify the method to send the data through

➜ Use the attribute (**method = "methodName"**)

There are 2 values of the methods:
1- GET
➜ This will send the values of the fields in the url of the website. (not secure)

2- POST
➜ This will send the values of the fields to the server side without passing it to the url of the website.

<input type="hidden" value = "value"/>

➜ This field used to put a specific value to be accessed in the server side that user cannot view it in the webpage.

<input type="reset" value = "value"/>

➜ Create a reset button that resets all form fields to its default value.

Color Field

`<input type="color" value = "value"/>`

➔ Create a color field that user can choose any color from a color panel.

Range of Values

`<input type="range"  name = "range" min = "0" max = "100" step = "20" value = "0" />`

➔ Make a range field that user can choose a number from 0 to 100 and can move 20 steps forward or backward
➔ The default value of this range = 0.

`<input type = "number" min = "0" max = "10" step = "1" />`

➔ Create a number field that allows only to write numbers on it.
➔ It restricts users to enter number that between **min** number and **max** number.
➔ We can modify the number of steps that each value can be increased if user use the keyboard to increase the number by the **step** attribute.

Prevent users to modify any field

➜ Add the attribute (**readonly**) .
➜ Users can select it and copy it, but cannot modify it.
➜ The data of input field with this attribute can be accessed from the server side.

We can add the attribute (**disabled**) to also prevent users to modify it

Difference

➜ The **readonly** attribute makes an input field read-only, which means that the user cannot modify its value, but the value is still sent to the server when the form is submitted.
➜ The **disabled** attribute, on the other hand, disables an input field, which means that the user cannot modify its value and the value is not sent to the server when the form is submitted.

Add the attribute (autofocus)

➜ Can be used to automatically focus on an input field when the page loads.
➜ This means that the input field will be selected and ready for the user to start typing in without them having to click on it first.

Restrict length for input value

➜ We can restrict users to enter a specific text within range of lengths using attributes (**minlength** and **maxlength**)

Single Selection

```
<input type = "radio" name = "countries" values = "Egypt" />
```

➜ The radio allows users to only choose one option from multiple options.
➜ This will make one radio button without any name describe It
➜ To make a name to describe it, create a label after each radio with its name.
➜ All radios within the same options' list, have the same attribute (**name**), this allow backend developer to access the selected option using this name attribute.

Make a default selected option

➜ Use the attribute (**checked**)

Select option through its label

➜ Use attribute (**id**) for the radio button.
➜ And attribute (**for**) for the label with the same id value.

```
<input type = "radio" name = "countries" values = "Egypt" id = "eg"/>
<label for="eg">Egypt</label>
```

Multiple selection

```
<input type = "checkbox" id = "eg" value = "Egypt" name = "countries"/>
<label for = "eg">Egypt</label>
```

Selection with drop-down list

➜ To create a selection options, use:

```
<select></select>
```

➜ In the select tag, we can define multiple options using: **<option></option>**.
➜ To separate options into groups, put it in the tag:**<optgroup></optgroup>**.
➜ To label an option group to be view to the user, use the attribute (**label = "groupName"**) for this **optgroup** tag.
➜ We can make the user choose multiple options by using the attribute (**multiple**) in the select tag.
➜ To make a specific option to be selected once the user loads the page, use the attribute (**selected**) in its **option** tag.

Text Area

➜ Use the tag **<textarea></textarea>**
➜ It is a resizable text area

➜ To make a upload file area, use the input tag with the attribute (**type = "file"**)

Search Field

➜ Use the input tag with the attribute (**type="search"**)
➜ This filed similar to the text type, but this field has a clear icon that allows users to clear the value of the field.

Date Field

➜ Use the input tag with the attribute (**type="date"**)
➜ It makes users to choose (day, month and year)
➜ To choose only month and year, use the **type = "month"**
➜ You can also use the **type = "time"** to choose only a time

After we press the submit button, the form values will be sent to the server side, and the page will be loaded again with no previous value
To make the request happen in another page so that the original page values don't be cleared
➜ Use the attribute (**target = "_blank"**) in the tag **form**

Specifies a list of pre-defined options for an <input> element.

```
<input list = "listName" name = "inputList" />
<datalist id = "listName">
  <option value = "CPP">
  <option value = "C#">
  <option value = "Java">
  <option value = "Python">
</datalist>
```

➔ The datalist **(id)** attribute value must be the same of the **(list)** attribute value of the input tag.
➔ You can write text that doesn't exist in the pre-options

In HTML, the spaces and newlines are ignored, but if we don't want to ignore it, put the text in the tag:

```
<pre></pre>
```

Code Section

To enter a code in HTML, put it between the tag:

```
<code></code>
```