

Task 1



Name: Matthew Kamal Saad

Submitted to: Eng. Ahmed Saber

Date: 3/10/2025

Table of Contents

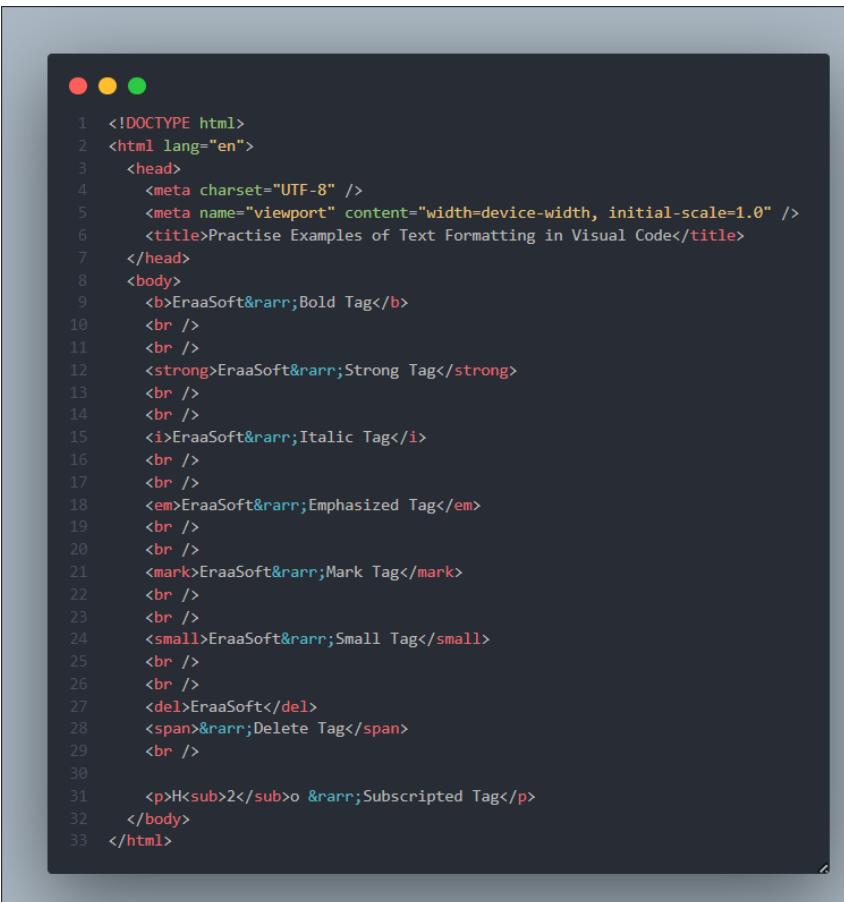
Text Formating in HTML.....	3
Special types of text.....	3
Practical Examples of Text Formatting Using Visual Studio Code	4
Media Tags in HTML.....	5
Iframes HTML.....	5
frame-Target for a link.....	6
Audio.....	7
Video	8
Forms	9
Forms HTML.....	9
Practical Examples of Forms using Visual Studio code	10
Semantic Html	11
References	12

HTML Formatting Elements

Formatting elements were designed to display special types of text:

Tags	Refer to
	Bold Text
	Important Text
<i>	Italic Text
	Emphasized Text
<mark>	Marked Text
<small>	Smaller Text
	Deleted Text
<sub>	Subscript Text
<sup>	Superscript Text

Practical Examples of Text Formatting Using Visual Studio Code



The screenshot shows a Visual Studio Code window displaying an HTML file titled "Practise Examples of Text Formatting in Visual Code". The code includes examples of bold, strong, italic, emphasized, mark, small, del, and subscripted text.

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <title>Practise Examples of Text Formatting in Visual Code</title>
7   </head>
8   <body>
9     <b>EraaSoft&rarr;Bold Tag</b>
10    <br />
11    <br />
12    <strong>EraaSoft&rarr;Strong Tag</strong>
13    <br />
14    <br />
15    <i>EraaSoft&rarr;Italic Tag</i>
16    <br />
17    <br />
18    <em>EraaSoft&rarr;Emphasized Tag</em>
19    <br />
20    <br />
21    <mark>EraaSoft&rarr;Mark Tag</mark>
22    <br />
23    <br />
24    <small>EraaSoft&rarr;Small Tag</small>
25    <br />
26    <br />
27    <del>EraaSoft</del>
28    <span>&rarr;Delete Tag</span>
29    <br />
30
31    <p>H<sub>2</sub>O &rarr;Subscripted Tag</p>
32  </body>
33 </html>
```

The browser tab title is "Practise Examples of Text Formatting in Visual Code". The browser address bar shows the URL <http://127.0.0.1:3000/Section 1/index.html?serverWindowId=44d0551e-971a-4bbd-b3b1-fc6ed2524565>.

EraaSoft→Bold Tag

EraaSoft→Strong Tag

EraaSoft→Italic Tag

EraaSoft→Emphasized Tag

EraaSoft→Mark Tag

EraaSoft→Small Tag

EraaSoft →Delete Tag

H₂O →Subscripted Tag

Media Tags in HTML

Iframes

Html iframes

The HTML <iframe> tag specifies an inline frame.

An inline frame is used to embed another document within the current HTML document.



Practical Examples of iFrames Using Visual Studio Code

Visual Studio Code interface showing an HTML file named "index.html". The code includes an iframe tag pointing to "https://codesandbox.io/". The right panel displays the rendered content of the iframe, which is the CodeSandbox landing page.

Visual Studio Code interface showing an HTML file named "index.html". The code includes an iframe tag pointing to "https://www.eraasoft.com/". The right panel displays the rendered content of the iframe, which is the EraaSoft website.

frame - Target for a Link

An iframe can be used as the target frame for a link.

The target attribute of the link must refer to the name attribute of the iframe:

Practical Examples of iFrames Using Visual Studio Code

The screenshot shows the Visual Studio Code interface. On the left is the Explorer sidebar with a project named 'ERASOFT FRONTEND DIPLOMA' containing files like index.html and index2.html. The main editor area shows the content of index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Practise Examples of Text Formatting in Visual Code</title>
</head>
<body>
    <iframe
        width="590"
        height="315"
        src="./index2.html"
        name="iframe-a"
        allowfullscreen>
    </iframe>
    <br />
    <p>
        <a href="https://www.eraasoft.com/" target="iframe-a">www.EraaSoft.com</a>
    </p>
</body>
</html>
```

To the right is a browser window titled 'Practise Examples of Text Formatting in Visual Code' showing the rendered HTML. A placeholder text 'The Anchor link will appear here' is visible in the iframe area. Below the browser window, the URL 'www.EraaSoft.com' is shown.

After Clicking on the Link

The screenshot shows the Visual Studio Code interface again. The browser window now displays the content from index2.html, which includes an iframe pointing back to index.html. An orange arrow points from the 'After Clicking on the Link' callout to the browser window. The browser title is 'Practise Examples of Text Formatting in Visual Code' and the URL is 'http://127.0.0.1:3000/Section 1/index.html?serverWindowId=6ba0ff30-014a-44e7-bda2-c9ca53045f'. The browser content shows the EraaSoft logo and Arabic text: 'تعلم الزجاجة الان' and 'لضمان مستقبل افضل'.

Audio

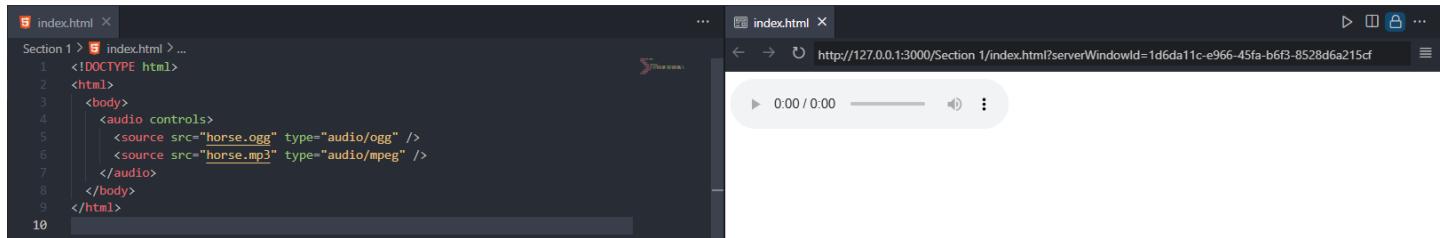
Html audio

To play an audio file in HTML, use the <audio> element

HTML Audio - How It Works

The controls attribute adds audio controls, like play, pause, and volume.

The <source> element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.



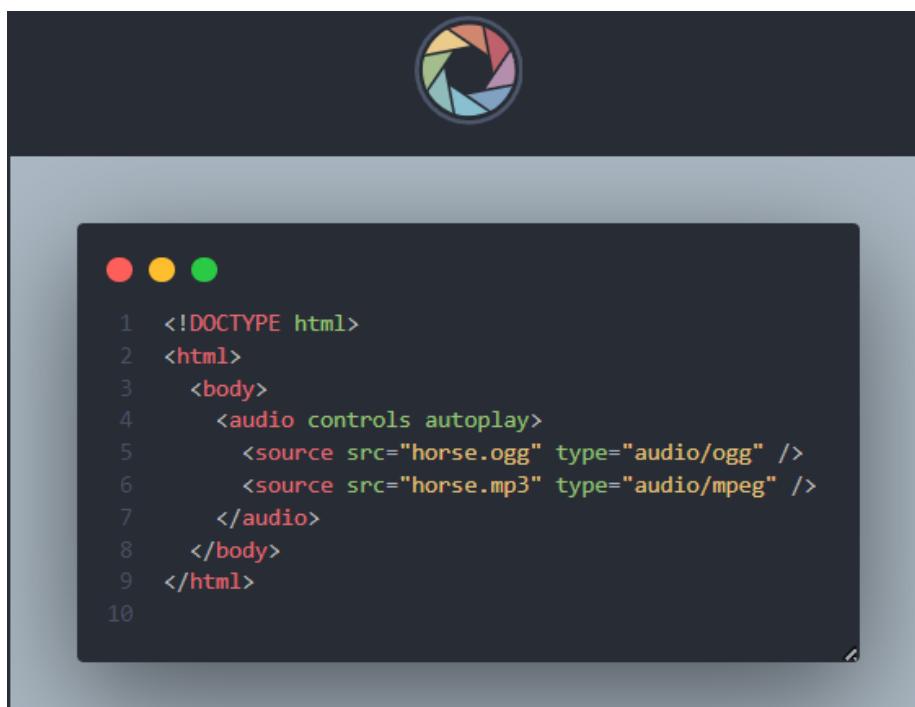
The image shows a split-screen view. On the left is a code editor window titled "index.html" containing the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <body>
4     <audio controls>
5       <source src="horse.ogg" type="audio/ogg" />
6       <source src="horse.mp3" type="audio/mpeg" />
7     </audio>
8   </body>
9 </html>
```

On the right is a browser window titled "index.html" showing the rendered audio player. The player has a play button, a progress bar showing "0:00 / 0:00", and volume control icons.

HTML <audio> Autoplay

To start an audio file automatically, use the autoplay attribute



The image shows a split-screen view. On the left is a code editor window titled "index.html" containing the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <body>
4     <audio controls autoplay>
5       <source src="horse.ogg" type="audio/ogg" />
6       <source src="horse.mp3" type="audio/mpeg" />
7     </audio>
8   </body>
9 </html>
```

On the right is a browser window showing the rendered audio player. The player has a play button, a progress bar, and volume control icons, indicating it is currently playing.

Video

Html video

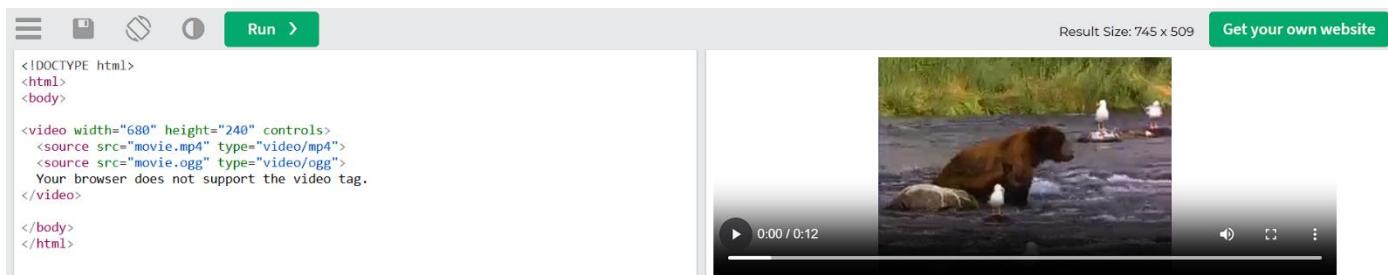
The HTML <video> element is used to show a video on a web page.

How it Works

The controls attribute adds video controls, like play, pause, and volume.

It is a good idea to always include width and height attributes. If height and width are not set, the page might flicker while the video loads.

The <source> element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized format.



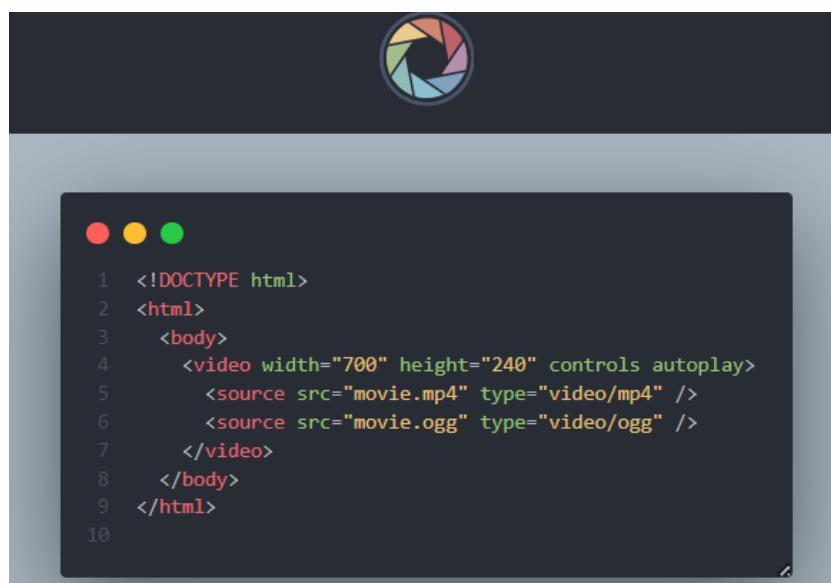
A screenshot of a code editor interface. On the left, the HTML code is displayed:

```
<!DOCTYPE html>
<html>
<body>
<video width="680" height="240" controls>
<source src="movie.mp4" type="video/mp4">
<source src="movie.ogg" type="video/ogg">
Your browser does not support the video tag.
</video>
</body>
</html>
```

On the right, there is a preview window showing a brown bear standing in shallow water, with several seagulls nearby. Below the video player, a progress bar shows "0:00 / 0:12". The top right corner of the preview window says "Result Size: 745 x 509" and "Get your own website".

HTML <video> Autoplay

To start an video file automatically, use the autoplay attribute



A screenshot of a code editor interface. The code is identical to the one in the previous screenshot, but the video player has started automatically, showing the same scene of a bear in water. The progress bar now shows "0:00 / 0:12".

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4      <video width="700" height="240" controls autoplay>
5          <source src="movie.mp4" type="video/mp4" />
6          <source src="movie.ogg" type="video/ogg" />
7      </video>
8  </body>
9 </html>
```

Forms in HTML

HTML forms are used to collect user input on web pages. They are defined using the `<form>` tag and can contain various interactive controls for users to input data.

Key Components of an HTML Form:

- **`<form>` element:**

This is the container for all form elements.

- **`<input>` element:**

This is a versatile element used for various input types, specified by the `type` attribute:

- `type="text"`: For single-line text input.
- `type="password"`: For password input (characters are masked).
- `type="email"`: For email address input.
- `type="number"`: For numerical input.
- `type="checkbox"`: For selecting multiple options.
- `type="radio"`: For selecting a single option from a group.
- `type="submit"`: For a button that submits the form.
- `type="button"`: For a generic clickable button.
- `type="hidden"`: For data that needs to be sent but not displayed to the user.

- **`<label>` element:**

Used to associate a descriptive label with a form control, improving accessibility. The `for` attribute of the `label` should match the `id` of the `input` it describes.

- **`<textarea>` element:**

For multi-line text input.

- **`<select>` element:**

Creates a dropdown list for selecting options.

- `<option>` elements define the individual choices within the dropdown.
- `<optgroup>` can be used to group related options.

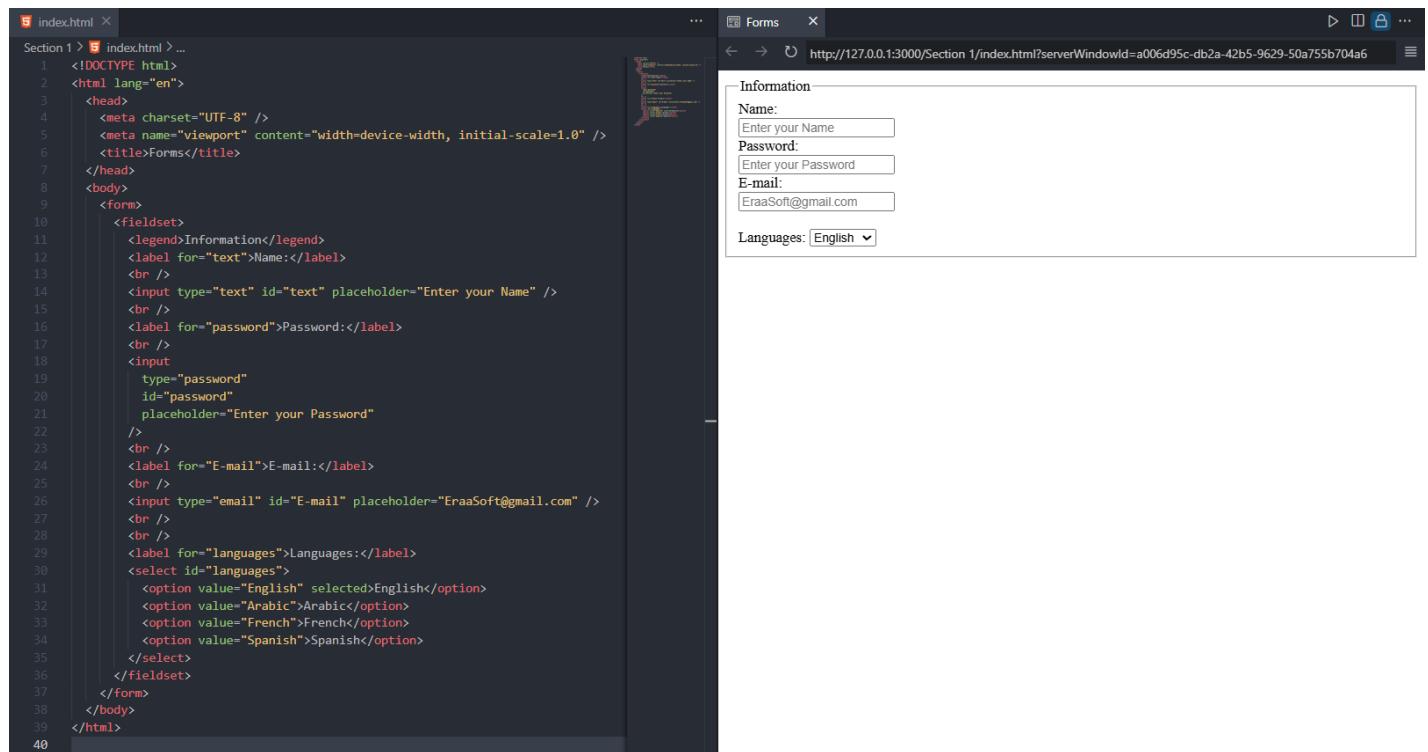
- **<button> element:**

Creates a clickable button. It can be used as a submit button or for other actions.

- **<fieldset> and <legend>:**

- <fieldset> groups related form elements, visually organizing the form.
- <legend> provides a caption for the <fieldset>.

Practical Examples of Forms Using Visual Studio Code



The screenshot shows the Visual Studio Code interface with two main panes. On the left, the code editor displays the content of an HTML file named 'index.html'. The code includes a form with a fieldset containing labels and input fields for Name, Password, E-mail, and Languages. On the right, a browser preview window shows the rendered HTML. The browser title is 'Forms' and the URL is 'http://127.0.0.1:3000/Section 1/index.html?serverWindowId=a006d95c-db2a-42b5-9629-50a755b704a6'. The preview shows a form titled 'Information' with fields for Name, Password, E-mail, and Languages, along with their respective input fields and placeholder text.

```
index.html
Section 1 > index.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Forms</title>
7  </head>
8  <body>
9      <form>
10         <fieldset>
11             <legend>Information</legend>
12             <label for="text">Name:</label>
13             <br />
14             <input type="text" id="text" placeholder="Enter your Name" />
15             <br />
16             <label for="password">Password:</label>
17             <br />
18             <input
19                 type="password"
20                 id="password"
21                 placeholder="Enter your Password"
22             />
23             <br />
24             <label for="E-mail">E-mail:</label>
25             <br />
26             <input type="email" id="E-mail" placeholder="EraaSoft@gmail.com" />
27             <br />
28             <br />
29             <label for="languages">Languages:</label>
30             <select id="languages">
31                 <option value="English" selected>English</option>
32                 <option value="Arabic">Arabic</option>
33                 <option value="French">French</option>
34                 <option value="Spanish">Spanish</option>
35             </select>
36         </fieldset>
37     </form>
38  </body>
39 </html>
40
```

Semantic HTML

What are Semantic Elements?

A semantic element clearly describes its meaning to both the browser and the developer.

Examples of **non-semantic** elements: <div> and - Tells nothing about its content.

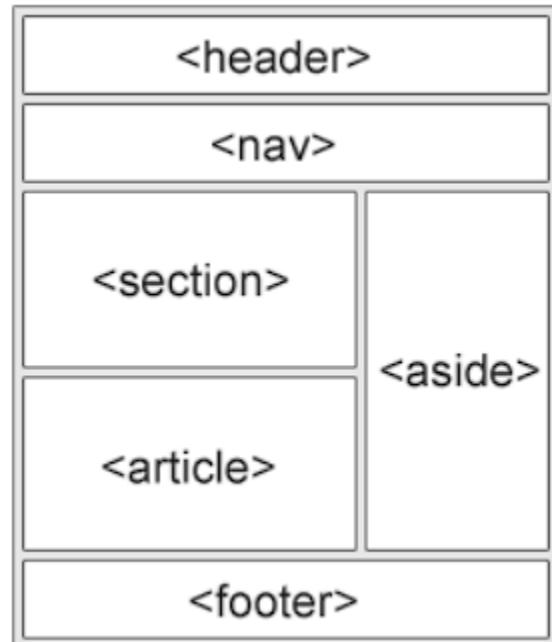
Examples of **semantic** elements: , <table>, and <article> - Clearly defines its content.

Semantic Elements in HTML

Many web sites contain HTML code like: <div id="nav"> <div class="header"> <div id="footer"> to indicate navigation, header, and footer.

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

- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>



References

- <https://www.w3schools.com/>
- <https://developer.mozilla.org/en-US/>