FUNDAMENTALS OF HTML

Comments in HTML

Generally, developers use comments to **explain their code to other developers** or mark something important that needs editing. Comments are **ignored** by the browser and **hence won't be seen on the webpage or other content.**

You can write a comment by placing the comment text between **<! > tags.**

For example:

```
<!-- This is a comment -->
```

Comments can't be nested, which means a comment can't be put inside another comment.

TAGS

Tags are used to represent HTML elements. These can be seen as keywords that define how a web browser will format and display the website content.

- Tags **define** all document elements, i.e. they give meaning to the plain HTML text.
- Two characters < and > surround HTML tags (They are called **angle brackets)**.
- The name of the tag can either begin with an **alphabet** or an **underscore (_)**.
- The element contents are displayed between the **start** and **end tags**
- Tags that have an **opening** and **closing** can have **any number of tags**

within them.

- The **<H1>** and **<h1>** tags in HTML have the **same meaning,** i.e. tags are **not case-sensitive.**
- The HTML tags are **usually available in pairs,** i.e. opening and closing (it's the same, with the tag name '/' at the beginning) tag.

Eg: <html> and </html> is a tag that comes in pairs and <hr> does not have a closing tag.

	Tag	**Description**
1	**<!DOCTYPE html>**	Specifies that HTML version 5 is used to create the web page
2	**<html> </html>**	Root container for all other HTML elements of the web page (including the head tag)
3	**<head> </head>**	The <head> element is a container for metadata (data about data) Metadata typically defines the document title, character set, styles, scripts, and other meta information.
4	**<title> </title>**	Provides the title of the document and is displayed in the tab in the browser
5	**<body></body>**	Contains all of the elements visible on the web page

NOTE*: There are also "self-closing" tags, whereby a br tag, for eg., will look like "****<br/>****" instead of simply "****<br>***

EXTRA:

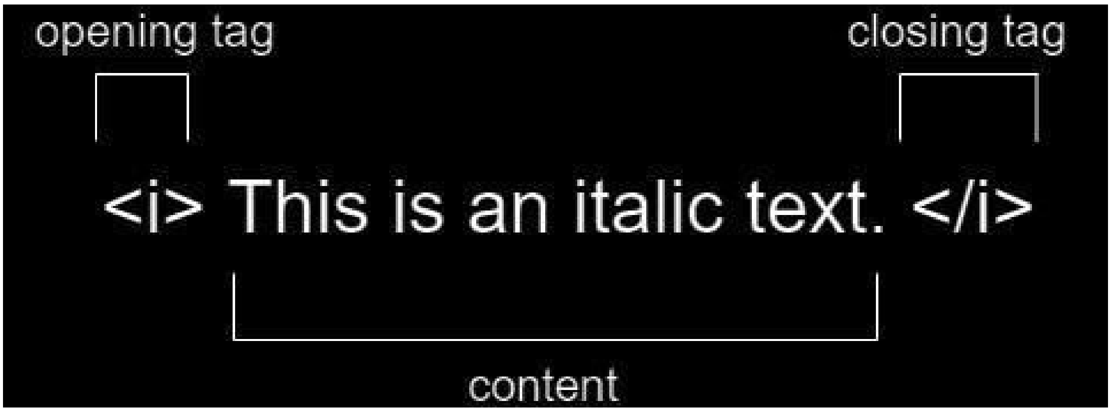
To get the list of all valid tags in HTML5, visit:

https://developer.mozilla.org/en-US/docs/Web/HTML/Element

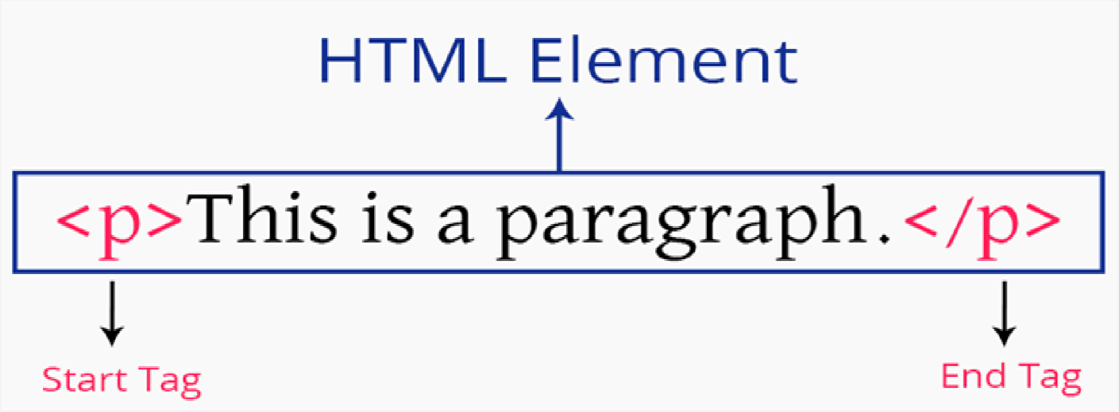
These can be explored whenever required while making a website.

HTML ELEMENTS

Elements are the things that make up the web page. Tags define the beginning and end of HTML elements. A web page can be seen as a collection of HTML elements.



```
<i> This is an italic text. </i>
```



```
<p>This is a paragraph.</p>
```

The basic elements used till now have been briefly described below

	HTML Element	**Description**
1	**<p> CONTENT </p>**	Paragraph tag
2	**<h1> CONTENT </h1>**	Heading tag
3	** **	Break tag - to enter into a new line
4	**<hr>**	Horizontal ruler

Paragraphs

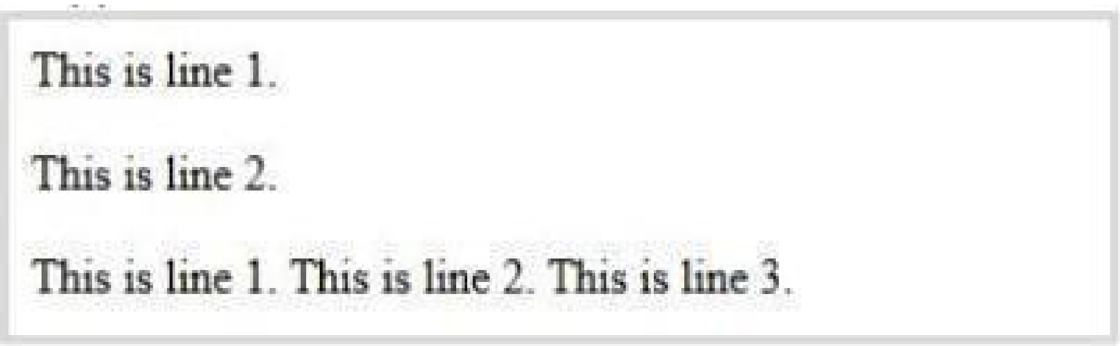
Paragraphs are **blocks of text** separated from each other by some space.

They are defined using the **<p>** and **</p>** tags. When the p element ends, the next element appears in the next line.

E.g.: here's a sample of code for <p> tag:

```
<!DOCTYPE html>
<html>
  <head>
    <title>p tag</title>
  </head>
  <body>
     <p>This is line 1.</p>
     <p>This is line 2.</p>
     <!-- trying to format the text wiathout using p-tag -->
     This is line 1. This is line 2. This is line 3.
  </body>
</html>
```

It appears on a web browser like this:



This is line 1.

This is line 2.

This is line 1. This is line 2. This is line 3.

NOTE: *When formatting without a p-tag, new lines are appended on the current line. This happens because the* ***spacing of text doesn't matter to the browser.***

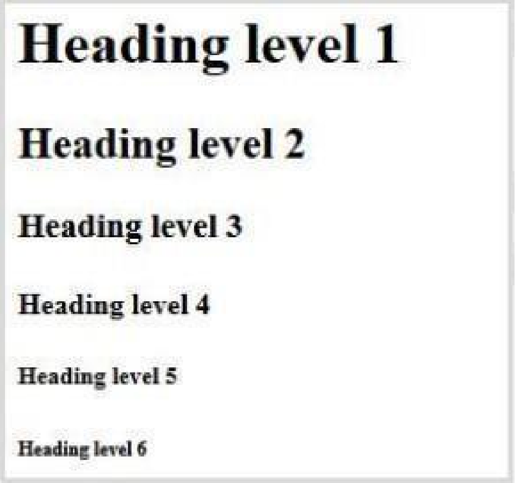
Headings

These are HTML tags that are used to indicate that some content should be treated as **headings**. The headings are divided into six levels: **h1, h2, h3, h4, h5**, and **h6.** Among them, **h1** is the **highest-level** heading and **h6** is the lowest-level heading.

E.g.: here's a sample of code for Heading tags

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Heading Levels</title>
  </head>
  <body>
    <h1>Heading level 1 </h1>
    <h2>Heading level 2 </h2>
    <h3>Heading level 3 </h3>
    <h4>Heading level 4 </h4>
    <h5>Heading level 5 </h5>
    <h6>Heading level 6 </h5>
  </body>
</html>
```

Output:



```
Heading level 1
Heading level 2
Heading level 3
Heading level 4
Heading level 5
Heading level 6
```

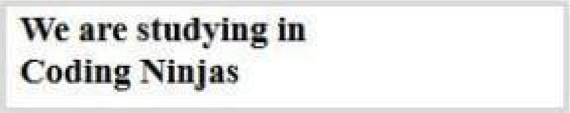
BR Tag

<br> tag can be used to make a **single line split** between the contents of the tab. This means that when this tag is used between a single line, the **contents after this tag will pass to the next line**. Do not use it to allow space between a block of elements (eg., paragraph and heading).

E.g.

```
<h3>We are studying in<br/>Coding Ninjas</h3>
```

The output is as follows:



We are studying in
Coding Ninjas

HR Tag

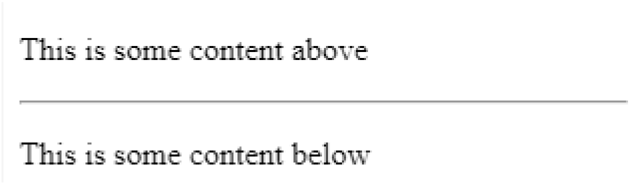
The <hr> tag in HTML, short for "horizontal rule," is a simple and effective element used to create a horizontal line or divider on a web page. This tag is self-closing, meaning it doesn't require a corresponding closing tag. Historically, the <hr> tag was employed to indicate a thematic break or a visual division between different sections of a document.

The basic syntax of the <hr> tag is uncomplicated: <hr>. When used in HTML, it produces a horizontal line that spans the width of its containing element. While it has attributes like align, width, and color, these are often considered deprecated in HTML5, and styling is usually accomplished through CSS for better control over appearance.

```
<!DOCTYPE html>
<html>
    <head>
      <title>Hello, World !< /title>
    </head>
    <body>
      <p>This is some content above</p>
      <hr>
```

```
        <p>This is some content below</p>
      </body>
</html>
```

Output: -



This is some content above

This is some content below

LISTS

Lists are used to **group different pieces of information** so that they are **easily linked** and **easy to read**.

Lists help construct a **well-structured**, more open, and **easy-to-maintain** document from a structural standpoint.

There are three types of lists to pick from: **ordered**, **unordered**, and **description lists**.

Unordered Lists

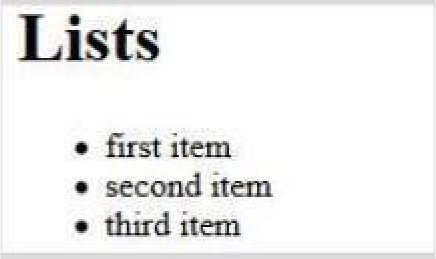
It's used to group a group of similar objects that aren't arranged in any specific order. Where the counting of objects isn't necessary, unordered lists are used.

Bullets are used by default to separate the items. They are defined using the <ul> tag.

Example:

```
<!DOCTYPE html>
<html>
 <head>
  <title>Unordered Lists</title>
 </head>
 <body>
  <h1>Lists</h1>
  <ul>
   <li>first item</li>
   <li>second item</li>
   <li>third item</li>
  </ul>
 </body>
</html>
```

The output is as follows:



Lists

- first item
- second item
- third item

HTML provides an interesting feature to change the style of the list item marker. There are 4 types of **styles in unordered lists:**

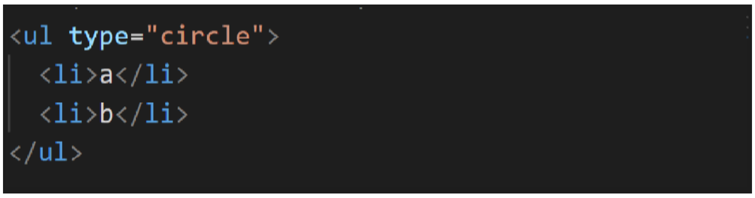
● ***type="disc"*** - Sets the list item marker to a bullet (default).

○ ***type="circle"*** - Sets the list item marker to a circle.

■ ***type="square"*** - Sets the list item marker to a square.

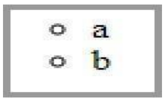
type="none" - The list items will not be marked.

For example, to create a list with ***type=circle:***



```
<ul type="circle">
  <li>a</li>
  <li>b</li>
</ul>
```

Output: -



```
○ a
○ b
```

***NOTE**: The above styles can be produced by using the '**type'** attribute.*

Ordered Lists

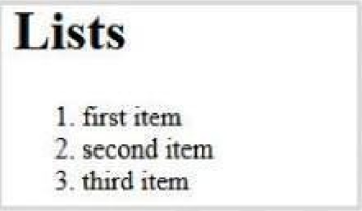
It is used in a certain order to group several related items.

When the n**umbering of items is necessary**, ordered lists are used. By default, **numerical numbers follow** the items.

They are defined using the **<ol>** tag. Eg:

```
<html>
  <head>
    <title>Ordered Lists</title>
  </head>
  <body>
    <h1>Lists</h1>
    <ol>
        <li>first item</li>
        <li>second item</li>
        <li>third item</li>
    </ol>
  </body>
</html>
```

The output is as follows:



```
Lists

   1. first item
   2. second item
   3. third item
```

Similarly, like the unordered lists, there are also different types of ways to number the ordered lists using the **'type'** attribute:

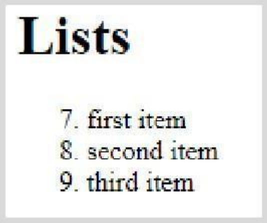
1. **type= "1"** - the numbering will contain **numbers** (default)
2. **type= "A"** - the numbering will contain **uppercase letters**
3. **type= "a"** - the numbering will contain **lowercase letters**

4. **type= "I"** - the numbering will contain **uppercase Roman numbers**
5. **type= "i"** - the numbering will contain **lowercase Roman numbers**

Now, what if you want to change the starting numbering of the lists?

HTML has got the solution for it: the **'start'** attribute. So, if we change <ol> to

<ol start="7">, you will now see the output as



Lists

7. first item
8. second item
9. third item

Description Lists

A list of definitions is **not the same** as a list of items. This is a **collection of items with an explanation.**

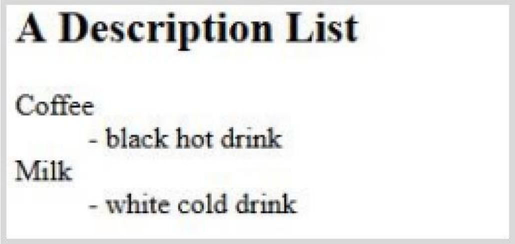
Tag	**Description**
<dl> tag	to **start** a definition list.
<dt> tag	to **begin** each definition - list term.
<dd> tag	to **begin each definition - list definition.**

In comparison to ordered and unordered lists, description lists are **very specific in their application** and thus are **rarely used**. However, whenever a structure such as a list of terms and their descriptions is required, description lists are ideal.

Example:

```
<!DOCTYPE html>
<html>
<head>
<title>Description Lists</title>
</head>
<body>
<h2>Description List</h2>
<dl>
<dt>Coffee</dt>
<dd>- black hot drink</dd>
<dt>Milk</dt>
<dd>- white cold drink</dd>
</dl>
</body>
</html>
```

The output is as follows:



A Description List

Coffee

    - black hot drink

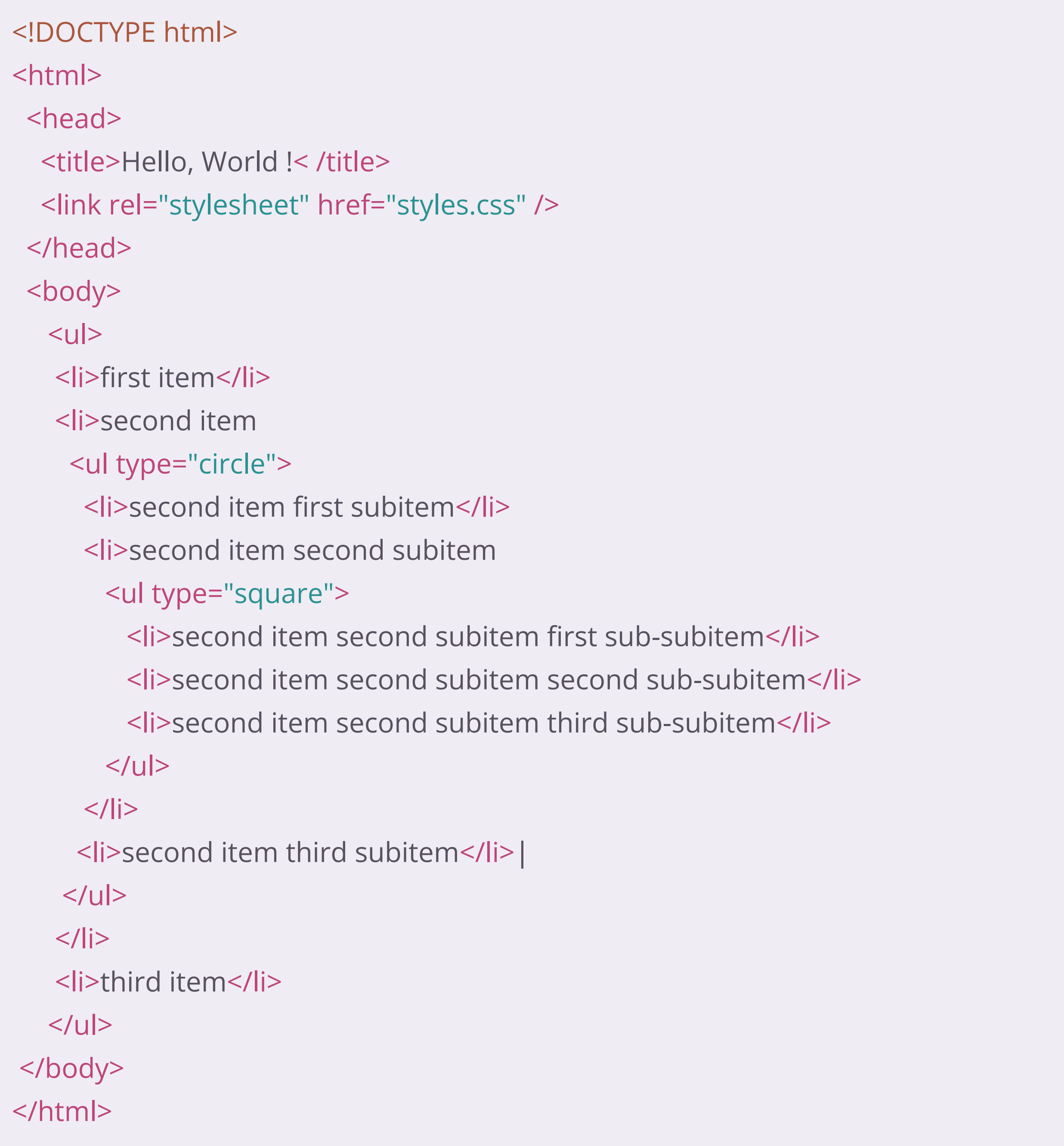
Milk

    - white cold drink

NESTING ELEMENTS

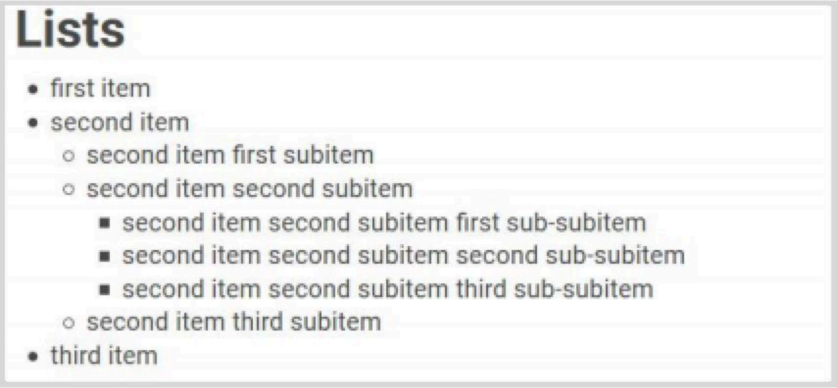
HTML elements can be nested i.e. **elements can contain elements.** All HTML documents consist of nested HTML elements

E.g.:



```
<!DOCTYPE html>
<html>
  <head>
    <title>Hello, World !< /title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <ul>
      <li>first item</li>
      <li>second item
        <ul type="circle">
          <li>second item first subitem</li>
          <li>second item second subitem
            <ul type="square">
              <li>second item second subitem first sub-subitem</li>
              <li>second item second subitem second sub-subitem</li>
              <li>second item second subitem third sub-subitem</li>
            </ul>
          </li>
          <li>second item third subitem</li>|
        </ul>
      </li>
      <li>third item</li>
    </ul>
  </body>
</html>
```

This will give the output as



Lists

- first item
- second item
    - second item first subitem
    - second item second subitem
        - second item second subitem first sub-subitem
        - second item second subitem second sub-subitem
        - second item second subitem third sub-subitem
    - second item third subitem
- third item

***NOTE**: There is no limitation to the depth of nested lists. Although it is true for all paired/container tags, we should be careful in nesting elements inside each other and should only do something meaningful.*

CONCLUSION:

In conclusion, the notes provide a foundational understanding of essential HTML concepts. The focus begins with the role of comments in HTML, serving as a means for developers to explain their code or highlight important elements for future editing. Notably, these comments remain invisible on web pages as browsers disregard them during rendering.

Moving on to tags, the notes articulate their significance as representations of HTML elements, dictating the formatting and display of content within a webpage. Tags, enclosed in angle brackets, offer a structured approach to organizing and presenting information. The case-insensitive nature of tag names and the necessity for paired opening and closing tags are emphasized.

The discussion extends to boilerplate tags, emphasizing the crucial role of **<DOCTYPE html>**, **<html>**, **<head>**, **<title>**, and **<body>** in establishing the foundational structure of HTML documents. The **<head>** section is highlighted for its role as a container for metadata, while **<title>** influences the document title displayed in the browser tab. The **<body>** tag encapsulates all elements visible on the webpage.

The discussion extends to boilerplate tags, emphasizing the crucial role of <DOCTYPE html>, <html>, <head>, <title>, and <body> in establishing the foundational structure of HTML documents. The <head> section is highlighted for its role as a container for metadata, while<title> influences the document title displayed in the browser tab. The <body>tag encapsulates all elements visible on the webpage.

The exploration of HTML elements delves into commonly used tags such as **<p>**, **<h1>**, **<br>**, and **<hr>**. These tags play key roles in defining paragraphs, headings, line breaks, and horizontal rules, respectively. The practical examples enhance the understanding of how these elements are employed in real-world scenarios.

Additionally, the notes comprehensively cover lists, differentiating between unordered (**<ul>**), ordered (**<ol>**), and description (**<dl>**) lists. Styling options for list items, including types and numbering, are elucidated. The notes also touch upon the concept of nesting elements, highlighting that HTML elements can be nested within each other, facilitating a structured and hierarchical organization of content.

Overall, this set of notes serves as a valuable introductory resource for individuals embarking on HTML-based web development. The detailed explanations, coupled with practical examples, contribute to a well-rounded comprehension of HTML fundamentals, laying a solid foundation for further exploration and learning in web development.

REFERENCES:

a. https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/HTML_text_fundamentals

b. https://developer.mozilla.org/en-US/docs/Web/HTML/Element