

# HTML - Elements

## HTML Elements Overview

**HTML elements** are the basic building blocks to create a web page; they are created with the help of an opening tag, content, and ending tag. HTML documents consist of a tree of these elements, and they specify how HTML documents should be built, and what kind of content should be placed within various parts of an HTML document.

## What is an HTML element?

An **HTML element** is a fundamental component of an HTML document that can contain data to display on the webpage, such as text, image, link, or sometimes nothing. An **HTML element** includes an opening tag, content, and a closing tag, where the opening tag may also include attributes.

An HTML element includes:

- **Opening Tag:** An opening tag specifies the beginning of the element and may include multiple **attributes**.
- **Content:** The content part includes the information to be displayed or processed within an element.
- **Closing Tag:** A closing tag marks the end of the element (A closing tag may be optional for some elements).

An HTML document consists of a tree of **HTML elements**, and they define the content and layout of a webpage, like how and what content should display in the different sections of a webpage.

## Example

Some of the examples of HTML elements are as follows:

```
<p>This is paragraph content.</p>
<h1>This is heading content.</h1>
<div>This is division content.</div>
```

## Explanation of Parts

The following table displays the different parts (opening tag, content, and closing tag) of the HTML elements used in the above example:

Opening Tag	Content	Closing Tag
<p>	This is paragraph content.	</p>

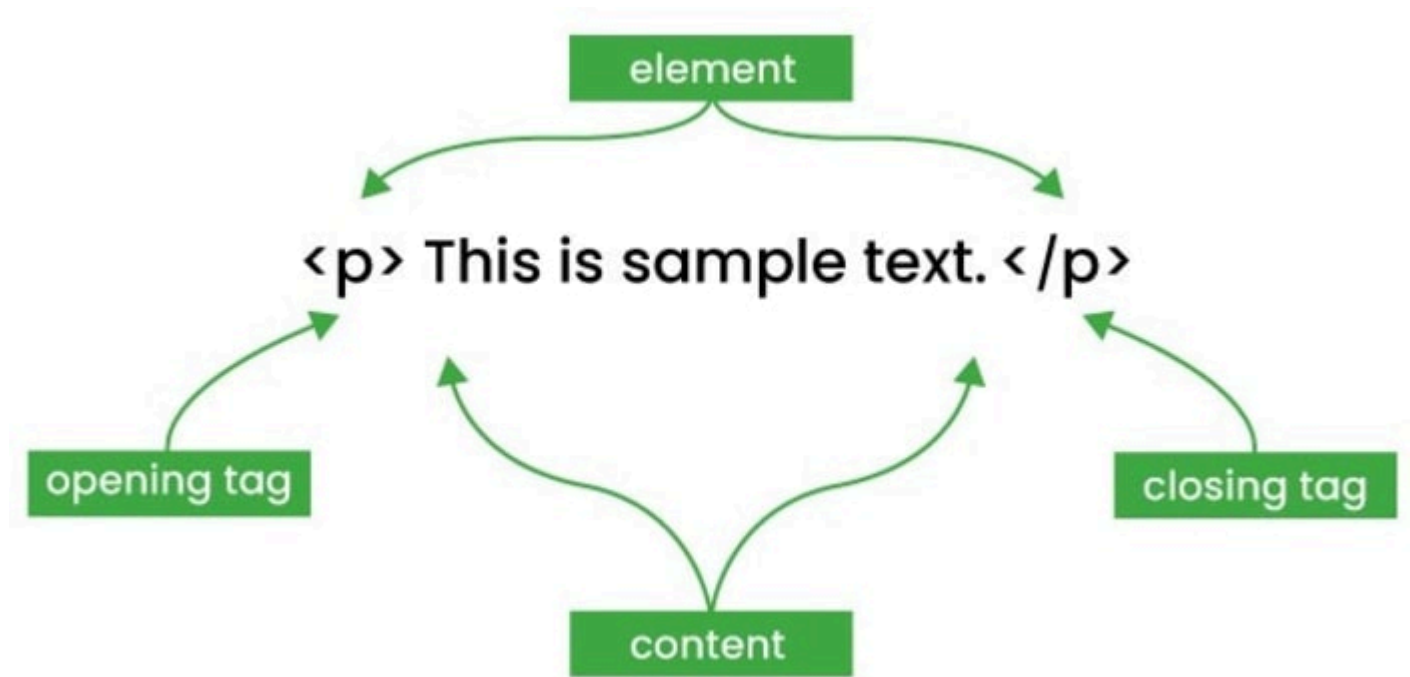
<h1>	This is heading content.	</h1>
<div>	This is division content.	</div>

So here **<p>...</p>** is an HTML element, **<h1>...</h1>** is another HTML element.

Explore our **latest online courses** and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

## HTML Element Structure

The following image demonstrates the structure of an element, like how an HTML element is written with the opening and closing tags:



## HTML Elements Vs. Tags

**HTML elements** are created using the HTML tags. An HTML element is defined by a pair of starting and closing tags having content between them, which defines the content and structure of the webpage. Whereas, **HTML tags** are like keywords and part of HTML elements enclosed in angel brackets (<>).

For example, **<p>** is the starting tag of a paragraph, and **</p>** is the closing tag of the same paragraph, but **<p>This is paragraph</p>** is a paragraph element.

## Nested HTML Elements

HTML allows nesting of elements. The nested elements are created by placing one or more HTML elements inside an HTML element. Where the container element can be considered as a parent element and others are as child elements. The child elements are nested inside the parent element. We can have multiple levels of nesting; however, it requires some guidelines to follow –

- Every opening tag must have a corresponding closing tag.
- The closing tag of the parent element must come after the closing tag of the child element.

- The nested elements must be valid HTML elements.

## Example

In the following example, we are nesting the italicized element (**<i>...</i>**) within the **h1** element and underline (**<u>...</u>**) element within the paragraph element.

</>

Open Compiler

```
<!DOCTYPE html>
<html>
<head>
  <title>Nested Elements Example</title>
</head>
<body>
  <h1>This is <i>italic</i> heading</h1>
  <p>This is <u>underlined</u> paragraph</p>
</body>
</html>
```

On executing the above example, we can observe the two sentences where we have nested one HTML within another.

**Note:** In the above example, the **<html>**, **<head>**, and **<body>** tags are also nested elements as they have one or more elements inside them.

## HTML Void Elements

HTML void elements are those elements that don't require closing tags. These tags don't have any content model and even don't allow nesting of elements. The void elements are also known as empty or self-closing elements.

Some of the void elements are such as **<img />**, **<hr />**, and **<br />** elements. The below table shows a list of void elements –

S.No	Elements & Description
1	<b>&lt;img /&gt;</b> Used for inserting images within HTML documents.
2	<b>&lt;hr /&gt;</b> It displays a horizontal rule.
3	<b>&lt;br /&gt;</b> It is used to provide a line break.
4	<b>&lt;source /&gt;</b>

It is used for embedding media resources like audio and video.

## Example

The following example demonstrates the example of HTML void elements –

</>

Open Compiler

```
<!DOCTYPE html>
<html>
<body>
  <p>This line contains a line break tag, <br> hence content is visible in two
  separate lines.</p>
  <p>This line is <hr>separated by a horizontal rule.</p>
</body>
</html>
```

On executing, the above code will produce two paragraphs, one with a line break and the other with a horizontal rule.

## Attributes With HTML Elements

The **attributes** can be placed with an opening tag by using the pairs of attribute name and value. Multiple attributes can be separated with a space.

The following statement demonstrates the use of two attributes **src** and **alt** with an HTML element **img**:

```

```

## Mandatory Closing HTML Elements

The HTML elements that are opened must be closed. Only the void elements like **<img />**, **<hr />**, **<br />**, etc. do not require closing tags; other elements such as **<p>** and **<h1>** require closing tags after the content part.

If any HTML element does not include a closing tag, it may not cause an error, and some content may still display properly. However, never miss the closing tag, as it may lead to unexpected and inconsistent results.

## Example

In this example, we are removing the closing tags from the paragraph tag. Still, it will show the correct result.

```
<!DOCTYPE html>
<html>
<body>
  <p>This line contains a line break tag, <br /> hence content is visible in two
  separate lines.
  <p>This line is <hr /> separated by a horizontal rule.
</body>
</html>
```

The above HTML code will produce two paragraphs, one with a line break and the other with a horizontal rule.

## Are HTML Elements Case-sensitive?

**No, HTML elements are not case-sensitive**, which means we can write HTML elements either in uppercase or lowercase. However, it is not a good practice, as W3C recommends and demands lowercase. Hence, always try to use lowercase for the tag names.

### Example

In the following example, we are writing HTML elements (tag names) in uppercase and mixed case (uppercase and lowercase); see the output; there is no error and HTML code runs fine –

```
<!DOCTYPE html>
<HTMl>
<BODY>
  <P>HTML is not case sensitive i.e we can use both upper or, lower case letters in
  the tags.</p>
</BOdy>
</html>
```