

|  |
|--|
| Professional Skills  |
| Agile Fundamentals   |
| Jira   |
| Git  |
| Databases Introduction   |
| Java Beginner  |
| Maven  |
| Testing (Foundation)   |
| Java Intermediate  |
| HTML <ul style="list-style-type: none"><li>Introduction to Web Development</li><li>Hypertext Markup Language</li><li>Tags</li><li>Structural elements</li><li>Metadata</li><li>Running a Web Server with VSC Live Server</li><li>Headings and paragraphs</li><li>Text formatting</li><li>Attributes</li><li>Images</li><li>Hyperlinks</li><li>Forms</li><li>Lists</li><li>Tables</li><li>Iframes</li></ul> |
| CSS  |
| Javascript   |
| Spring Boot  |
| Selenium   |
| Sonarqube  |
| Advanced Testing (Theory)  |
| Cucumber   |
|  |

## Tags

### Contents

- [Overview](#)
- [Tutorial](#)
  - [Paired and self-closing tags](#)
  - [Required and optional attributes](#)
  - [Parsing](#)
- [Exercises](#)

### Overview

When working with HTML, we use tags around our code to create different functionality. Tags tend to be come in pairs - one when opened (e.g. `<p>`), and one when closed (e.g. `</p>`). There are exceptions where some tags are self-closing, however.

### Tutorial

#### Paired and self-closing tags

```
<h1>This is a big heading</h1>
<h6>This is a small heading</h6>
<p>This is a paragraph</p>
```

HTML self closing tag:

```

<br/>
<hr/>
```

It is important to ensure that tags are closed in the order in which they are opened.

The below example is incorrect as the `</b>` tag is closed after the `</p>` tag:

```
<p>This is some text and this is <b>important in bold</p> </b>
```

This is the correct way to open and close tags:

```
<p>This is some text and this is <b>important</b>in bold</p>
```

The reasoning for this is the same as closing brackets in order in other programming languages - the scope of the program (or, in this case, the HTML DOM) will simply not understand what you're trying to do.

#### Required and optional attributes

There are some tags in HTML that will also contain attributes some are **required** and some are **optional**:

```
<h1 id="myHeading"></h1>
<!-- This has an optional id attribute -->
```

```

<!-- This has a mandatory src attribute -->
```

#### Parsing

|                      |
|----------------------|
| MongoDB              |
| Express              |
| NodeJS               |
| React                |
| Express-Testing      |
| Networking           |
| Security             |
| Cloud Fundamentals   |
| AWS Foundations      |
| AWS Intermediate     |
| Linux                |
| DevOps               |
| Jenkins Introduction |
| Jenkins Pipeline     |
| Markdown             |
| IDE Cheatsheet       |

Why does HTML use tags in the first place, and not brackets?

The simple answer is *HTML parsing*.

When we save a file with the `.html` extension, we signal to the browser engine to interpret the file as an HTML document.

The browser parses the document using a process of *tokenisation*, where every starting and ending HTML tag in the file is accounted for.

Each HTML tag corresponds to a *token* which the browser knows as part of its HTML parsing rule set.

The browser understands each string in angle brackets (e.g., `<html>`, `<p>`), and pattern-matches them to the set of rules that apply to each of them. (For example, a token that represents an anchor tag will have different properties from one that represents a paragraph.)

After the tokenization is done, the tokens are then converted into *nodes* - different chunks of data, containing the tag, its rules, and the content it contains - and are built into a tree-like hierarchy: the **Document Object Model (DOM)**.

[We speak more about the DOM in the Document Object Model module.](#)

## Exercises

1. Create a html document with the following paragraphs:

- This
- is
- a
- paragraph

► Solution