

HTML Basics

HTML stands for Hyper Text Markup Language. It is used by Internet browsers to display web content in a structured manner. HTML is a tag-based language and uses tags to identify its elements. Authoring an HTML document is simply a matter of putting content within the relevant tags. For example, if we have to create a paragraph for some text, we can use the `<p>` tag to mark the start of the paragraph add the text, and then use the `</p>` to end the paragraph. In fact, the entire HTML document itself sits within two tags -- it begins with `<html>` tag and once the entire document is done, it ends with the `</html>` tag.

Like the above examples of `<p>` and `<html>`, majority of HTML elements are identified as a pair of tags: the first tag (the opening tag) marks the beginning of an element and the second tag (the closing tag) marks the ends of that element. However, there are a handful of HTML elements that work just fine even with one single tag. These elements do not enclose any content and hence, all they need is the opening tag. Since they do not enclose any content, they are also known as empty elements, void elements, or singleton elements.

In terms of structure, an HTML document has two prominent tags: head and body. The head section is identified by the `<head>` and `</head>` tag-pair and contains meta-information of the HTML page (like say, the title of the page). The body section is identified by the `<body>` and `</body>` tag-pair and encloses the actual content of the HTML page. However, these head and body tags are optional. Thus, it is possible to have a valid HTML page that does not contain any these four tags: `<head>`, `</head>`, `<body>`, and `</body>`!

Hello World

Now that we are familiar with a little bit of HTML basics, let us dive into it and write our very first HTML document! We provide the document below.

```
<!doctype html>  
  
<html>  
  
<head>  
  
  <title> Hello World! </title>  
  
</head>  
  
<body>  
  
  <!-- This is a comment -->  
  
  Hello World!  
  
</body>  
  
</html>
```

Let us now describe various elements in the above example.

The example begins with the doctype declaration -- this should be the first thing in an HTML document. The doctype declaration is used to indicate the markup language and the version of the page to the Internet browsers. Without this declaration, some of the browsers could use their own default mode. Strictly speaking, the "`<!doctype html>`" is used for HTML5 documents.

After the doctype declaration, the example has the `html` element that starts with the `<html>` tag and its in the end, closes with the `</html>` tag. Thus, with the exception of the doctype, the rest of the page content sits inside the `html` element. Next, the `html` element itself contains two child elements: `head` and `body`. The `head` element (identified by `<head>` and `</head>`) does not contain much except the title element. We should mention that as per W3C, it is mandatory to have a title element for each HTML page. Next, the `body` element (identified by `<body>` and `</body>`) has two child elements of its own: a comment element and a text.

Let us load this page in a browser. To do that, we would need to double click the file with the extension `.html` . Once loaded, we would notice two things. First, the browser displays the "Hello World!" in the main window - this is the text present in the `body` element. Second, we would also see the browser displaying "Hello World!" along with "Mozilla Firefox" in the title bar -- this is the text added in the title element (we use Mozilla Firefox for our example). In case you are wondering, it is common for browsers to append the HTML title with their name.

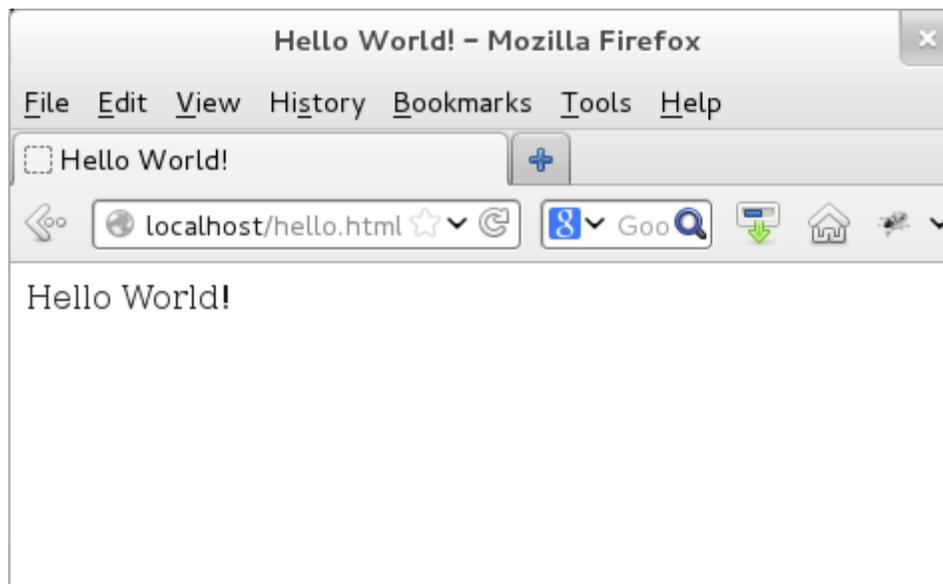


Figure: Hello World

The Structure, the Behavior, and the Style

Having tags allows HTML to do many things. We can divide HTML functionality into three broad areas: structure content, dynamic (interactive) behavior, and style. When we write an HTML document, these three areas are often the three main areas of focus. HTML elements (like `<head>`, `<p>`, etc) provide the structure for the document, HTML script elements (e.g.

JavaScript, jQuery) provide the behavior for the document, and the Cascading Style Sheets (CSS) elements provide the design aspect for the document.

Keeping these three areas separate allows us to develop each of them independently. The World Wide Web Consortium (W3C), which is the main body responsible for standardization of Web technologies, recommends the same. The common way to achieve this separation is to keep the script and the style content in different files and link them to the HTML page.