

Hypertext Markup Language
HTML

Nathan Warner



**Northern Illinois
University**

Computer Science
Northern Illinois University
September 19, 2023
United States

Contents

1	Boilerplate template	2
2	Tags	2
3	Line break	2
4	centering content	2
5	Horizontal lines	3
6	Preserve Formatting	3
7	Nonbreaking spaces	3
8	Elements	4
9	Attributes	4
	9.1 Core attributes	4
10	Setting background image for webpage	5
11	Formatting	5
12	Div and span	6
13	Meta tags	6
14	Page Redirection	6
15	Lists	7
	15.1 Unordered	7
	15.2 Ordered	7
	15.3 Description Lists	7
16	Links	8
	16.1 Attributes	8
17	Images	8
18	Select	8
19	Input	9
20	Forms	10
	20.1 Attributes	10
	20.2 Form Controls	10

1 Boilerplate template

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>This is document title</title>
6     <link rel="stylesheet" href="style.css">
7 </head>
8 <body>
9     <h1>This is a heading</h1>
10    <p>Document content goes here.....</p>
11 </body>
12 </html>
```

2 Tags

- <!DOCTYPE...> This tag defines the document type and HTML version.
- <html> This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags.
- <head> This tag represents the document's header which can keep other HTML tags like <title>, <link> etc.
- <title> The <title> tag is used inside the <head> tag to mention the document title.
- <body> This tag represents the document's body which keeps other HTML tags like <h1>, <div>, <p> etc.
- <h1> This tag represents the heading

3 Line break

```
1 <br />
```

4 centering content

```
1 <center>
2 Some text
3 </center>
```

5 Horizontal lines

```
1 <hr />
```

6 Preserve Formatting

```
1 <pre>
2 some
3     text
4 </pre>
```

7 Nonbreaking spaces

```
1 <p>An example of this technique appears in the movie "12&nbsp;Angry&nbsp;Men."</p>
```

8 Elements

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

9 Attributes

An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts: a name and a value:

```
1 <p align="left">This is left aligned</p>
```

9.1 Core attributes

The four core attributes that can be used on the majority of HTML elements (although not all) are:

- Id: The id attribute of an HTML tag can be used to uniquely identify any element within an HTML page.
- Title: The behavior of this attribute will depend upon the element that carries it, although it is often displayed as a tooltip when cursor comes over the element or while the element is loading.
- Class: The class attribute is used to associate an element with a style sheet, and specifies the class of element.
- Style: The style attribute allows you to specify Cascading Style Sheet (CSS) rules within the element.
- Dir: The dir attribute allows you to indicate to the browser about the direction in which the text should flow.
 - ltr: Left to right (the default value)
 - rtl: Right to left (for languages such as Hebrew or Arabic that are read right to left)
- align: Horizontally aligns tags
 - right
 - left
 - center
- background: Places a background image behind an element (Used only in body tag, and tables)
 - URL

10 Setting background image for webpage

To set the background image for the webpage, we can add the *background* attribute for the *body* tag:

```
1 <body background="image/linktoimage"> </body>
```

11 Formatting

- Bold: `
 </br>`
- Italic: `<i> </i>`
- Underline: `<u> </u>`
- Strike out: `<strike> </strike>`
- Monospaced font: `<tt> </tt>`
- Superscript text: ``
- Subscript text: ``
- Inserted text: `<ins> </ins>`
- Deleted text: ` `
- Larger text: `<big> </big>`
- Smaller Text: `<small> </small>`
- Emphasize text: ` `
- Marked Text: `<mark> </mark>`, this element will display text with yellow ink.
- Strong text: ` `
- Abbreviation: `<abbr title="Abhishek">Abhy</abbr>`
- Acronym: `<acronym>XHTML</acronym>`
- Quoting Text: `<blockquote>...</blockquote>`
- Short Quotations: `<q> </q>`
- Text Citations: `<cite> </cite>`
- Computer Code: `<code> </code>`
- Keyboard Text: `<kbd> </kbd>`
- Programming Variables: `<var> </var>`
- Address Text: `<address> </address>`

12 Div and span

The `<div>` and `` elements allow you to group together several elements to create sections or subsections of a page.

- Div: `<div>` `</div>`: Used for larger groupings... Perhaps multiple tags
- Span: The `` element, on the other hand, can be used to group inline elements only. So, if you have a part of a sentence or paragraph which you want to group together

Span:

```
1 <p> This is <span style="color:blue"> some </span> text </p>
```

13 Meta tags

You can add metadata to your web pages by placing `<meta>` tags inside the header of the document which is represented by `<head>` and `</head>` tags. A meta tag can have following attributes in addition to core attributes:

- charset: Specifies the character encoding for the HTML document
- name: Name for the property. Can be anything. Examples include, keywords, description, author, revised, generator etc.
- http-equiv: Used for http response message headers. For example, http-equiv can be used to refresh the page or to set a cookie. Values include content-type, expires, refresh and set-cookie.
- content: Specifies the property's value.

```
1 <meta http-equiv="refresh" content="30"> <!-- Refresh the page every 30 seconds ---!>
```

14 Page Redirection

You can use `<meta>` tag to redirect your page to any other webpage. You can also specify a duration if you want to redirect the page after a certain number of seconds.

```
1 <head>
2 <title>Meta Tags Example</title>
3 <meta name="keywords" content="HTML, Meta Tags, Metadata" />
4 <meta name="description" content="Learning about Meta Tags." />
5 <meta name="revised" content="Tutorialspoint, 3/7/2014" />
6 <meta http-equiv="refresh" content="5; url=http://www.tutorialspoint.com" />
7 </head>
```

15 Lists

15.1 Unordered

```
1 <ul>
2   <li>Coffee</li>
3   <li>Tea</li>
4   <li>Milk</li>
5 </ul>
```

15.2 Ordered

```
1 <ol>
2   <li>Coffee</li>
3   <li>Tea</li>
4   <li>Milk</li>
5 </ol>
```

15.3 Description Lists

```
1 <dl>
2   <dt>Coffee</dt>
3   <dd>- black hot drink</dd>
4   <dt>Milk</dt>
5   <dd>- white cold drink</dd>
6 </dl>
```

16 Links

```
1 <a href="url">link text</a>
```

16.1 Attributes

The target attribute can have one of the following values:

- `_self` - Default. Opens the document in the same window/tab as it was clicked
 - `_blank` - Opens the document in a new window or tab
 - `_parent` - Opens the document in the parent frame
 - `_top` - Opens the document in the full body of the window
-

```
1 <a href="https://www.w3schools.com/" target="_blank">Visit W3Schools!</a>
```

17 Images

```
1 
2 
```

18 Select

The select element is used to create drop-down lists

```
1 <label for="some name">Some text: </label>
2 <select name="somename" id="someid">
3     <option value="someidentifier">option1</option>
4     <option>option2</option>
5     <option>option3</option>
6     <option>option4</option>
7 </select>
```

19 Input

The `<input>` HTML element is used to create interactive controls for web-based forms in order to accept data from the user

```
1 <label for="mylabel"> Some label text </label>
2 <input type=""> </input>
```

Available **type** values:

- Text
- Button
- Checkbox
- Color
- Date
- Email
- File
- Hidden
- Image
- Month
- Number
- Password
- Range
- Reset
- Search
- Submit
- Tel
- Url

We can also provide the attributes **minlength** and **maxlength**.

20 Forms

the `<form>` element represents a section of a document that contains form controls, such as text fields, checkboxes, radio buttons, submit buttons, and more. The `<form>` element can be used to collect user input for further processing, such as submitting to a server for registration, data retrieval, or any other type of client-server interaction.

20.1 Attributes

- **action:** Specifies the URL to which the form data should be sent when the form is submitted.
- **method:** Defines the HTTP method to be used when sending the form data. Common methods are `GET` (appends form data to the URL in name/value pairs) and `POST` (sends the form data in the body of the request).
- **enctype:** Specifies how the form data should be encoded when submitting it to the server. It's used when the form contains `<input type="file">` elements. Common values include:
 - `application/x-www-form-urlencoded` (default): All characters are encoded before being sent.
 - `multipart/form-data`: No characters are encoded, necessary for file uploads.
 - `text/plain`: Spaces are converted to `+` symbols, but no other characters are encoded.
- **target:** Specifies where to display the response after submitting the form, e.g., `_blank` (a new tab/window), `_self` (the same tab/window, default), `_parent`, `_top`, or the name of a frame.
- **autocomplete:** Specifies whether the browser should enable autocomplete for the form. It can have values `"on"` or `"off"`.
- **novalidate:** This Boolean attribute indicates that the form shouldn't be validated when submitted. It's useful when you have custom validation scripts.

20.2 Form Controls

- Text fields (`<input type="text">`)
- Radio buttons (`<input type="radio">`)
- Checkboxes (`<input type="checkbox">`)
- Password fields (`<input type="password">`)
- Submit buttons (`<input type="submit">` or `<button type="submit">`)
- Dropdown lists (`<select>`)
- Textarea (`<textarea>`)
- File upload (`<input type="file">`)

Example:

```
1 <form action="/submit-data" method="POST">
2   <label for="username">Username:</label>
3   <input type="text" id="username" name="username" required>
4
5   <label for="password">Password:</label>
6   <input type="password" id="password" name="password" required>
7
8   <input type="submit" value="Login">
9 </form>
```
