

# Navigation and Links

HTML5



# Anchors (Links)

To add a hyperlink to a web page, use the [a](#) (anchor) element. The content of the element will be the text visible to the user, and the “[href](#)” attribute determines where the user will end up if they click that visible text.

```
<p>  
  To learn more about the anchor element, click  
  <a href="https://developer.mozilla.org/en-US/docs/Web/HTML/Element/a">  
    here!  
  </a>  
</p>
```

To learn more about the anchor element, click [here!](#)

That example used an absolute path (an exact path directly to the resource from a domain’s root.) This is often the approach used when linking to other websites.

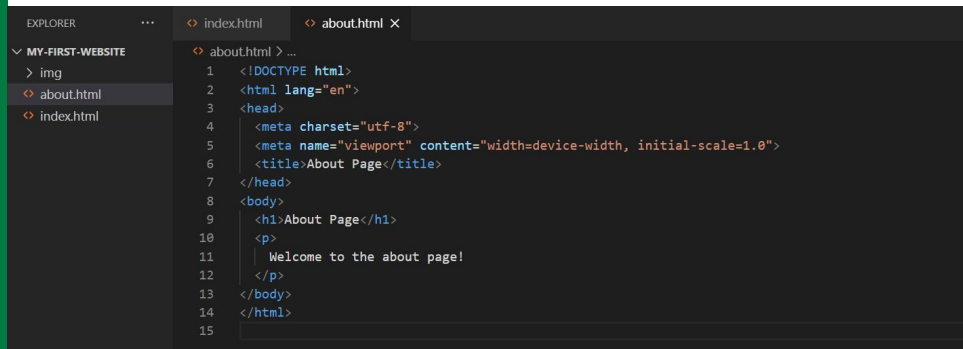
Anchors can also use relative paths, which is preferred when travelling to other pages within your website.

Let’s observe the use of a relative path in an anchor.

# Linking Your Pages

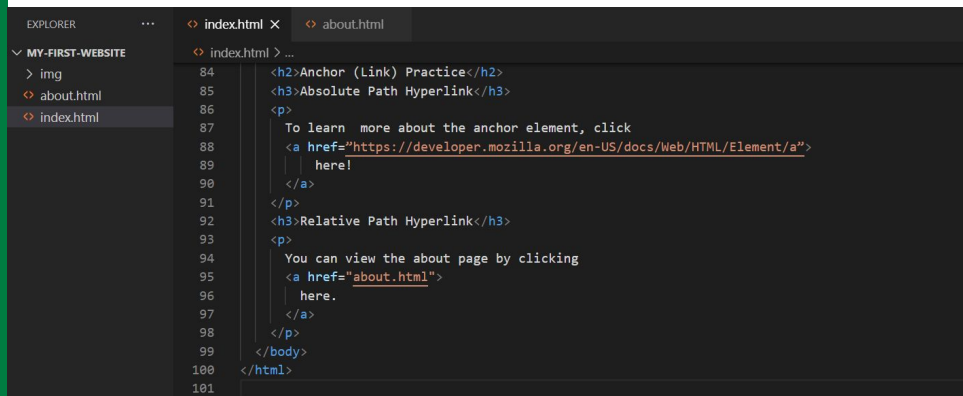
We can use anchors to have hyperlinks bring our users from page to page within our website.

Add an about.html page, so we have somewhere to link to!

A screenshot of the Visual Studio Code editor. The Explorer sidebar on the left shows a project named 'MY-FIRST-WEBSITE' with files 'img', 'about.html', and 'index.html'. The 'about.html' file is selected and its content is displayed in the main editor. The code is a basic HTML document with a DOCTYPE declaration, language attribute, UTF-8 charset, viewport meta tag, and a title 'About Page'. The body contains an h1 heading 'About Page' and a paragraph 'Welcome to the about page!'.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>About Page</title>
7 </head>
8 <body>
9   <h1>About Page</h1>
10  <p>
11    Welcome to the about page!
12  </p>
13 </body>
14 </html>
15
```

Add an anchor, linking to the new about.html page.

A screenshot of the Visual Studio Code editor. The Explorer sidebar on the left shows the same project with 'index.html' selected. The main editor shows the content of 'index.html' with two new paragraphs added. The first paragraph is titled 'Anchor (Link) Practice' and contains a link to a Mozilla developer page. The second paragraph is titled 'Relative Path Hyperlink' and contains a link to 'about.html'.

```
84 <h2>Anchor (Link) Practice</h2>
85 <h3>Absolute Path Hyperlink</h3>
86 <p>
87   To learn more about the anchor element, click
88   <a href="https://developer.mozilla.org/en-US/docs/Web/HTML/Element/a">
89     here!
90   </a>
91 </p>
92 <h3>Relative Path Hyperlink</h3>
93 <p>
94   You can view the about page by clicking
95   <a href="about.html">
96     here.
97   </a>
98 </p>
99 </body>
100 </html>
101
```

# Other Uses for Anchors

Many users have an e-mail client (like [Outlook](#) or [Thunderbird](#)) on their computer, or on their phone. With this in mind, it can be very convenient to click or tap an e-mail address to begin composing an e-mail addressed to that e-mail address.

You can use an anchor to accomplish this by prepending the href attribute value with “mailto:”, like so...

**<a href="mailto:billy@example.com">**

Click here to e-mail billy@example.com!

**</a>**

The same can be said for phone numbers. Users with digital phone call software on their computers, or users on a mobile device, expect to be able to click or tap on a phone number that appears in a web page to begin a call with that person or business.

For telephone numbers, we prepend the href attribute value instead with “tel:”, like so...

**<a href="tel:17801234567">**

Click here to call +1 (780) 123-4567!

**</a>**

# Linking to a Location on the Current Page

An important attribute afforded to any HTML element is the [id](#) (identifier) attribute. The value for an id must be unique within the page it appears upon (no duplicate id values are allowed.)

Assigning ids gives us the opportunity to have an anchor scroll to the element with that id on the page when clicked, but their use doesn't cease there.

Using ids is also common, and sometimes entirely necessary, when writing CSS and JS code as well. It gives those languages ways to target and manipulate specific elements in your web pages.

To specify which element id to scroll to, precede the href attribute value with an octothorpe (#), specifying the id.

A common use case for linking to locations inside the page is a table of contents that links to important headings lower in the article...

```
<a href="#intro">
```

Click here to scroll to the introduction.

```
</a>
```

Might scroll you to...

```
<h2 id="intro">
```

Introduction

```
</h2>
```

# Navigation Element

When compiling a collection of links, it is expected you bundle them together inside of a [nav](#) (navigation) element.

It is not uncommon for anchors / links to be placed into an ordered or unordered list to be more granular in your description of the nav's contents. The previous example we made, for instance, might feature a nav that looks something like so...

```
<nav>
  <h2>Practice Navigation</h2>
  <ul>
    <li>
      <a href="index.html">
        Home
      </a>
    </li>
    <li>
      <a href="about.html">
        About
      </a>
    </li>
  </ul>
</nav>
```