Div/Span

The Key Differences Between `span` and `div` Tags

You can use both the `span` and `div` tags as a container if you want to make a particular part of the web page distinct and style it differently. But again, they don't serve the exact same purpose.

The HTML div Tag

The div tag is a generic block-level element used for associating and grouping together a larger chunk of a web page – usually a section such as a header, footer, the main content, and so on.

In the example below, I group the header of a web page together with the `div` tag and styled it using CSS.

```
<div class="header">
  <h2 class="logo">freeCodeCamp</h2>
  <ul class="nav">
    <li>
      <a href="">Home</a>
    </li>
    <li>
      <a href="">About</a>
    </li>
    <li>
      <a href="">Serices</a>
    </li>
    <li>
      <a href="">Contact</a>
    </li>
  </ul>
</div>
```

In the CSS below, I laid out the header and the navbar in it with CSS Flexbox. I also removed the default margin and padding assigned to elements by browsers.

```
* {
      margin: 0;
      padding: 0;
      box-sizing: border-box;
    }

.header {
      padding: 0 70px;
      display: flex;
      align-content: center;
      justify-content: space-between;
      margin-top: 20px;
      margin-bottom: 20px;
      color: crimson;
    }

.nav {
      display: flex;
      align-content: center;
      justify-content: center;
      gap: 60px;
      list-style-type: none;
    }

.nav li a {
      text-decoration: none;
      font-size: 1.2rem;
      color: crimson;
    }
```

In addition, you can use the div tag to group similar content together. This could be similar text, images, videos, and so on. So, you can always nest divs within divs, and attach unique classes or id attributes to them so you don't get confused.

The HTML `span` Tag

The `span` tag is an inline element that you use to make a smaller part of content stand out with CSS or JavaScript. You shouldn't nest span unless you thoroughly know what you're doing – but you can put multiple span tags within a block-level element.

In the example below, I made some particular words stand out by wrapping span tags around them and styling them differently.

```
<p>
      This is a <span class="crimson">crimson text</span> within black texts.
      This is an <span class="indigo">indigo text</span> within others, and this
      is an <span class="orange">orange text</span> within other texts.
</p>
```

```
body {
    display: flex;
    align-items: center;
    justify-content: center;
    max-width: 900px;
    margin: 0 auto;
    height: 100vh;
  }

p {
    font-size: 2.5rem;
  }

.font-style {
    font-style: italic;
  }

.crimson {
    color: crimson;
  }

.indigo {
    color: indigo;
  }

.orange {
    color: orange;
  }
```

You can see the most important differences between the `span` and `div` tags in the table below:

`SPAN` **TAG**	`DIV` **TAG**
Inline-level element	Block-level element
Used for grouping small chunks of text	Used for grouping large chunks of texts together
Must not be nested to avoid confusion	Usually nested

When should you use `span` or `div`?

You should use `span` when you want to style a particular part of your content differently or manipulate it with JavaScript. You can also use it as a container for inline elements.

You should use the `div` tag, on the other hand, if you want to group large chunks of content together, and when you want to layout elements on the web page.