Exercises: Content and Structure

In this lesson, our content gains structure!
Let's begin!

WE'LL COVER THE FOLLOWING

- Inline and block content
- Exercise 02-02: Adding inline and block elements to our html
- Exercise 02-03 – Changing default behavior using `display`

HTML defines a few dozen markup elements. Most of them define content, while other elements define **logical containers**. For example, the `<h1>` elements define the content and the markup that signs that the content within should be rendered as a first-level heading.

In contrast, the `<section>` element is a **logical container**. It tells the browser that the nested content defines a section of the document (e.g., a sidebar, a part of a chapter, etc.).

Inline and block content

The elements that provide content can be inline or block elements.

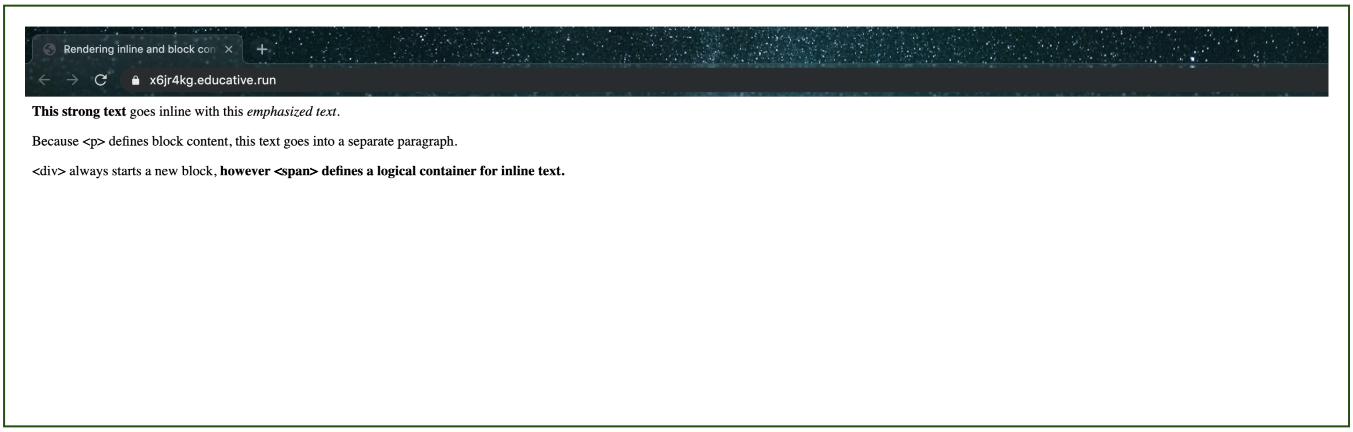
Inline elements render their content so that the content goes on the same "line" as the previous markup, while block elements start a new block from the beginning of a new line.

To make it easier to understand, Exercise 02-02 shows a simple example of inline and block elements:

Exercise 02-02: Adding inline and block elements to our html

```
<!DOCTYPE html>
<html>
<head>
    <title>Rendering inline and block content</title>
</head>
<body>
  <strong>This strong text</strong>
  goes inline with this
  <em>emphasized text.</em>
  <p>
    Because &lt;p&gt; defines block content,
    this text goes into a separate paragraph.
  </p>
  <div>
    &lt;div&gt; always starts a new block,
    <span style="font-weight: bold">
      however &lt;span&gt; defines a
      logical container for inline text.
    </span>
  </div>
</body>
</html>
```

When you check how this markup is rendered in a browser, you'll see the page displayed like in the image below:



This strong text goes inline with this *emphasized text.*

Because <p> defines block content, this text goes into a separate paragraph.

<div> always starts a new block, **however <span> defines a logical container for inline text.**

Rendering inline and block content

This page contains three blocks represented by the three paragraphs in the figure above. The first block is rendered according to the content of the three lines directly following the tag. It is assembled from three parts: the `<strong>` element, the plain text starting with "goes", and the `<em>` element. Because all of them are inline elements, they are rendered as a single line.

The `<p>` markup defines a paragraph that holds block content by definition as that's what is expected from a paragraph; that is why it is started in a new line, and does not go on the same line as the previous markup.

The third block is rendered as it is, because `<div>` defines a block, while

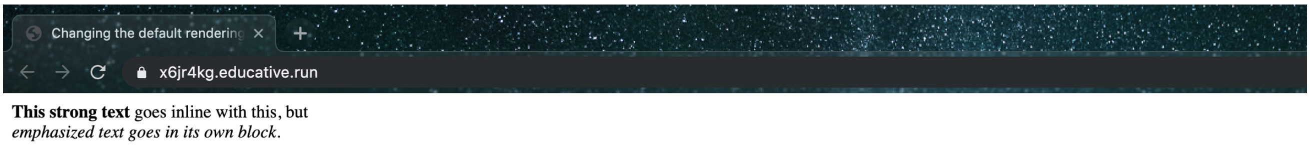
`<span>` defines inline content.

The default behavior can be changed through the display property of the style attribute. For example, in **Exercise 02-03**, adding the `“display: block”` property to the `<em>` element’s style forces starting a new block.

Exercise 02-03 – Changing default behavior using `display`

```
<!DOCTYPE html>
<html>
<head>
    <title>Changing the default rendering style</title>
</head>
<body>
  <strong>This strong text</strong>
  goes inline with this, but
  <em style="display: block">
    emphasized text goes in its own block.
  </em>
</body>
</html>
```

The result of this slight change is shown in the image below:



This strong text goes inline with this, but
emphasized text goes in its own block.

The rendering style of <em> has been changed to “block”

In the *next lesson*, we’ll render text and enrich our web page.