Boxing Inline Elements

In this lesson, we will learn how to set box size for inline elements. Let's begin!

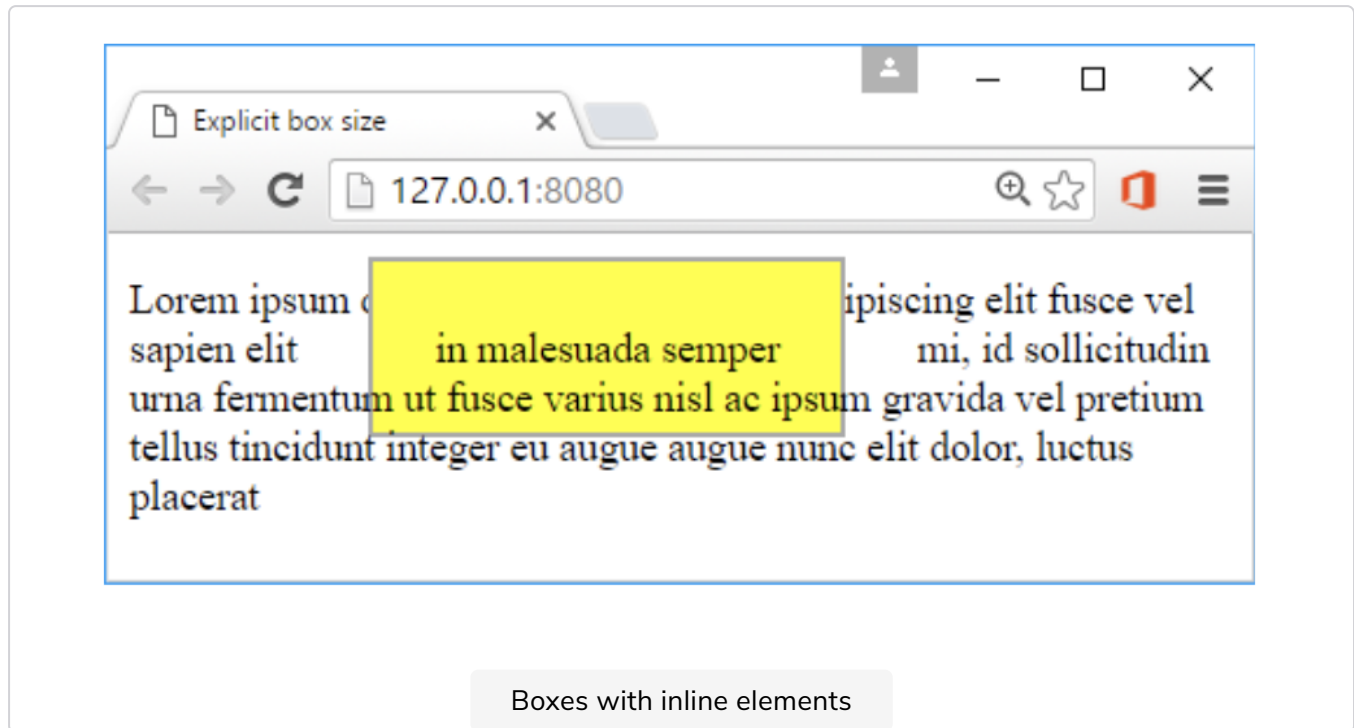Inline elements are also handled in a way other than block elements, and this is why you cannot set their size explicitly—or if you set, those size properties are ignored. However, you can still apply the box model properties (margin, border, and padding), but they generally provide a different result than you expect. Take a look at Listing 10-11, which defines box model properties for an inline `<span>` tag.

Listing 10-11: Explicit box size

```
<!DOCTYPE html>
<html>
<head>
  <title>Explicit box size</title>
  <style>
    .highligted {
      margin: 24px;
      border: 2px solid darkgray;
      background-color: yellow;
      padding: 24px;
    }
  </style>
</head>
<body>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipiscing
    elit fusce vel sapien elit
    <span class="highligted">in malesuada semper</span>
    mi, id sollicitudin urna fermentum ut fusce varius nisl
    ac ipsum gravida vel pretium tellus tincidunt integer
    eu augue augue nunc elit dolor, luctus placerat
  </p>
</body>
</html>
```

The image below shows how this markup is rendered:

Boxes with inline elements

This unexpected behavior is a direct result of the rendering algorithm used by the browser.

Inline elements are rendered line by line and lines are calculated according to the width of the browser window. When calculating the height of an inline element, the margin, border, and spacing properties are ignored, so the browser does not consider the box properties of the "in malesuada semper" `<span>` tag. Nonetheless, the calculation of width takes these properties into account, and also, the box with the border and padding area is drawn when the `<span>` is displayed.

The first line is rendered before the `<span>` tag, so it is covered by the box that is drawn together with the second line. However, the third line is drawn after the second line, so it is stacked upon the box already drawn.

In the *next lesson*, we'll learn how to use floating elements in CSS.