

Activity 1 - Block versus inline

We've provided you with a [starter project file](#). (The code is included below, as well). It can be opened with the [Intel XDK](#).

This simple file has two block level elements (the paragraphs) and two spans, which are inline display. Additionally, there are several classes defined for you.

Try the following:

- Notice immediately each paragraph gets its own line.
- Notice the inline span that follows the paragraph (`</p>`) starts on its own line. But the second inline span directly follows its predecessor - it does not get a new line, it continues on the same line.
- Try applying the `brext` and `prext` classes to the elements. (`viaclass="brext"`). When applied to the paragraphs you can see that the width of the paragraph stretches to the edge of the window. It is the same as its parent width. Note that its height is no more than its content. But the background colors and borders are tight to the spans.
- Try applying the `w` class to each of the elements. This class sets the `width` property. It works on both paragraphs, but it has no effect on the two spans.
- Try applying the `h` class to each element. This class sets the `height` property. Again, the paragraphs are affected, but the spans are not.
- Apply the `pad` class to each element. This class sets the `padding` property. The paragraphs are clearly padded. The spans are also padded, but the extra padding does not space them out. So if the background colors are still being applied, the padded background of one inline element may overlap the other. Can you think of a solution that would prevent the overlap?
- Apply the `marg-ver t` class to each element. This class sets the top and bottom margins. Note that it works on the block level paragraphs well enough, but has no effect on the inline spans.
- Apply the `marg-horiz` class to each element. This class sets the left and right margins. It works on everything.

Play some more with the elements, trying to appreciate how block and inline elements differ.

- Try changing the HTML so that the inline elements are inside the block level ones.
- Put a `margin:0` on the body .
- Have fun!

NOTE: This activity is best conducted in the XDK (or in CodePen), **not in the discussion forum** below. The discussion forum includes its own stylesheet which may override the styles you want to experiment with.

```
<!DOCTYPE html>
<html>
<head>
  <title>Block vs Inline</title>
  <style>
    .u { text-decoration: underline; }
    .brect {
10.     background-color:lightblue;
        border:1px dotted dimgray;
    }
    .prect {
        background-color:lightpink;
        border:1px dotted dimgray;
    }
    .w {
20.     width:300px;
    }
    .h {
        height:100px;
    }
    .pad {
        padding:20px;
    }
    .marg-vert {
        margin-top:30px;
        margin-bottom:30px;
30.    }
    .marg-horiz {
        margin-left:30px;
        margin-right:30px;
    }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
40. <p class="">He was a squyer of lowe degre</p>
```

```
<p class="">Will Ladislaw's mind was now wholly bent on  
seeing Dorothea again, and forthwith quitting  
Middlemarch. </p>
```

```
<span class="">Ladislaw felt the awkwardness of asking  
for more last words.</span>
```

```
<span class="u">It is certainly trying to a man's  
dignity to reappear when he is not expected to do so: a  
first farewell has pathos in it, but to come back for a  
second lends an opening to comedy</span>
```

```
</body>
```

```
</html>
```