

CSS and Bootstrap

Block and Inline Elements

- **Block elements** start on a new line and take up full width of parent
- **Inline elements** don't start on a new line and only take up width of content

```
<h2>Headings are block elements</h2>
```

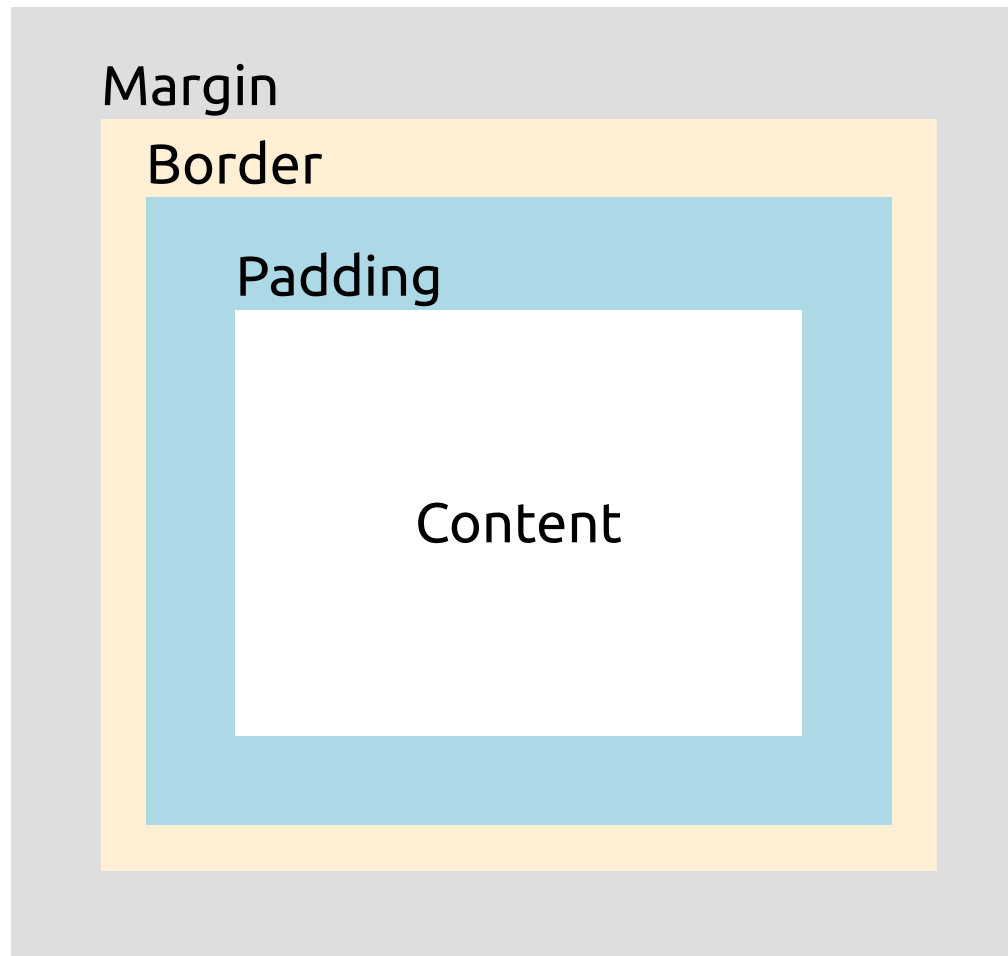
```
<p>So are paragraphs. <em>These</em> are <strong>inline</strong></p>
```

Headings are block elements

So are paragraphs. *These* are **inline**

Box Model

- Every HTML element is a box



Box Properties

- **Content** - The text, image, or media content
- **Padding** - The space between the box's content and border
- **Border** - Line that goes around content and padding
- **Margin** - The space between the box and other boxes

Padding

```
div {  
  padding-top: 20px;  
  padding-bottom: 20px;  
  padding-left: 40px;  
  padding-right: 40px;  
}
```

Content

Shorthand Property

```
// Vertical Horizontal
```

```
padding: 20px 40px;
```

```
// Top Right Bottom Left
```

```
padding: 20px 40px 20px 40px;
```

Content

Border

```
div {  
  // Width Style Color  
  border: 8px solid lightblue  
}
```

Content

Margin

```
div {  
  // Vertical Horizontal  
  margin: 20px 40px;  
  
  // Top Right Bottom Left  
  margin: 20px 40px 20px 40px;  
}
```

Box 1

Box 2

Vertical Margins Collapse

```
div {  
  // Vertical Horizontal  
  margin: 20px 40px;  
}
```

Box 1

Box 2

Margin vs Padding

- Margins are always transparent
- Vertical margins collapse
- Can't click on margin
- Inline elements ignore top and bottom margin

Aligning Boxes

```
div {  
  margin: auto; // center  
  margin-left: auto; // right-align  
  margin-right: auto; // left-align  
}
```

Box 1

Box 2

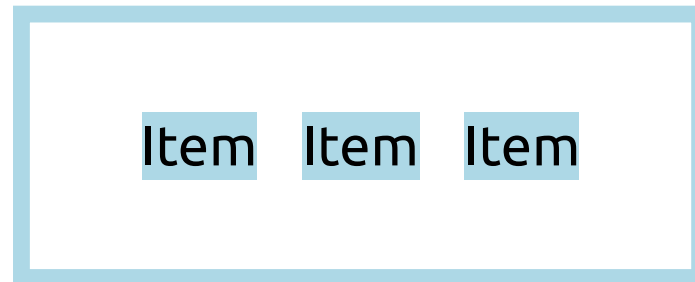
Box 3

Flexbox

Two types of boxes:

- **Flex containers** - parent element
- **Flex items** - children elements

Flex Container



Flex Container

```
div {  
  display: flex;  
}
```

Flex Container Properties

- flex-direction - direction of items
- justify-content - alignment of items along main axis
- flex-wrap - wrap items to new line

```
div {  
  display: flex;  
  // row | column | row-reverse | column-reverse  
  flex-direction: column;  
  flex-direction: column-reverse;  
}
```

Item 1

Item 2

Item 3

Item 3

Item 2

Item 1

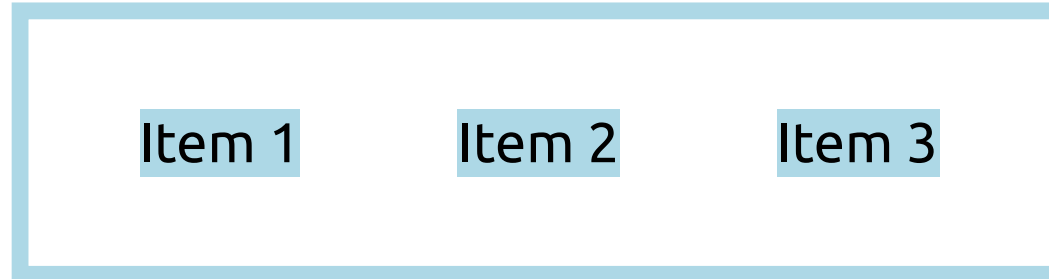

```
div {  
  display: flex;  
  flex-direction: row;  
  // flex-start | flex-end | center |  
  // space-between | space-around | space-evenly  
  justify-content: flex-end;  
}
```



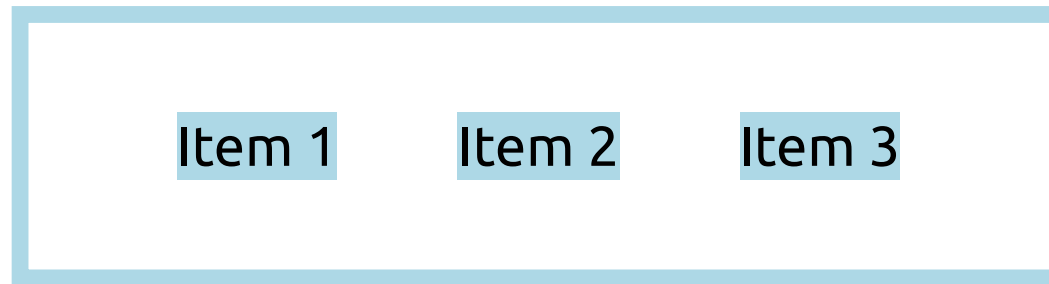
A diagram illustrating the CSS flexbox layout. It shows a horizontal container with a light blue border. Inside the container, three items labeled "Item 1", "Item 2", and "Item 3" are displayed. Each item is enclosed in a light blue rectangular box. The items are aligned to the right side of the container, demonstrating the effect of the `justify-content: flex-end;` property.

Item 1 Item 2 Item 3

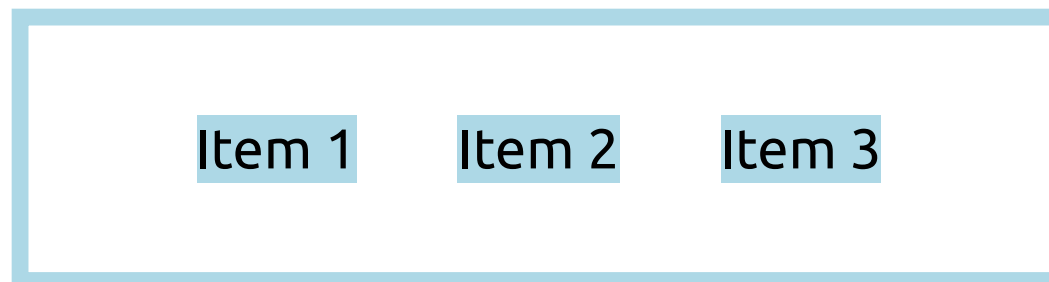
Space Between



Space Around



Space Evenly



```
div {  
  display: flex;  
  width: 400px;  
  flex-wrap: wrap;  
}
```

Item 1

Item 2

Item 3

Item 4

Flex Item Properties

- flex-basis - default size of item
- flex-grow - grow factor of item
- flex-shrink - allows item to shrink

Bootstrap

What is Bootstrap

- CSS Framework
- Easy to style and build website

Exercise