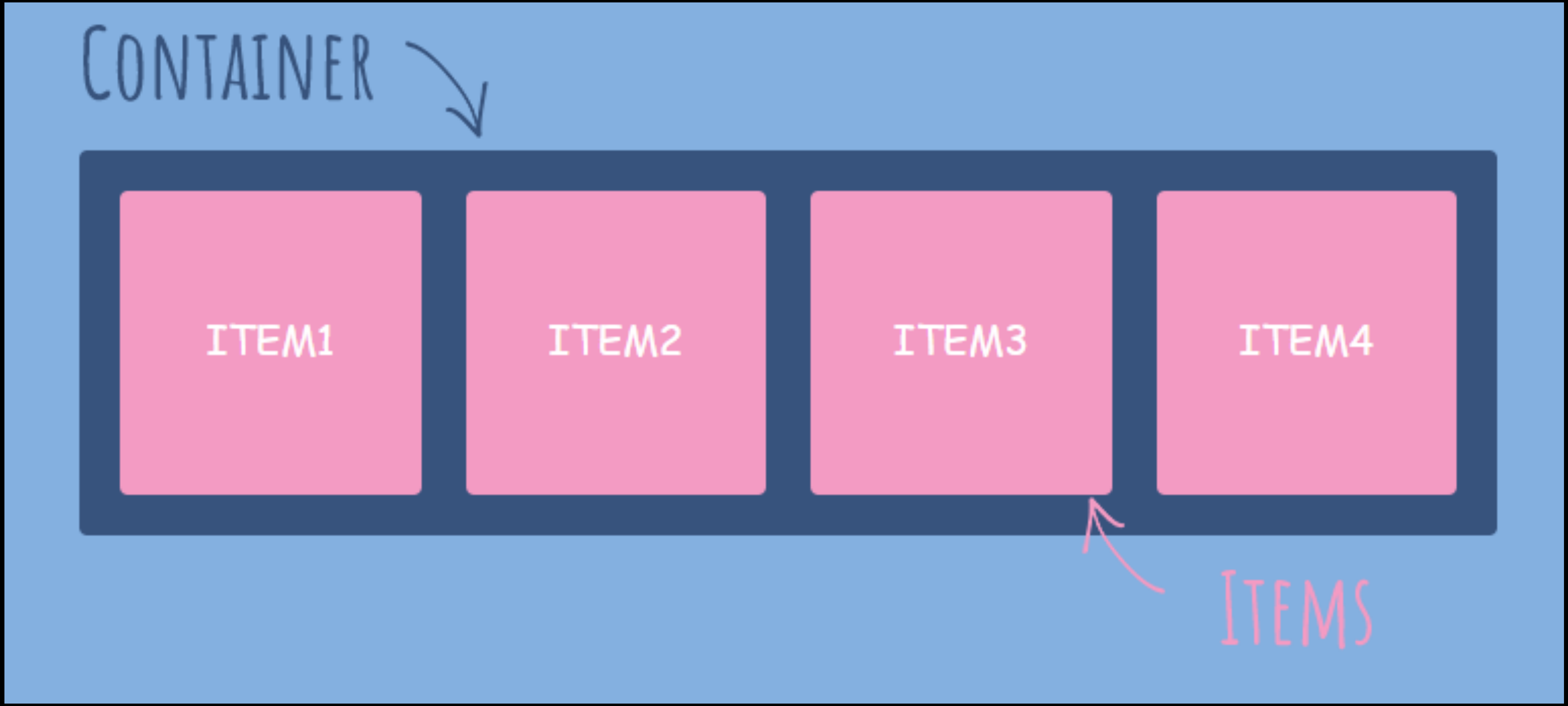
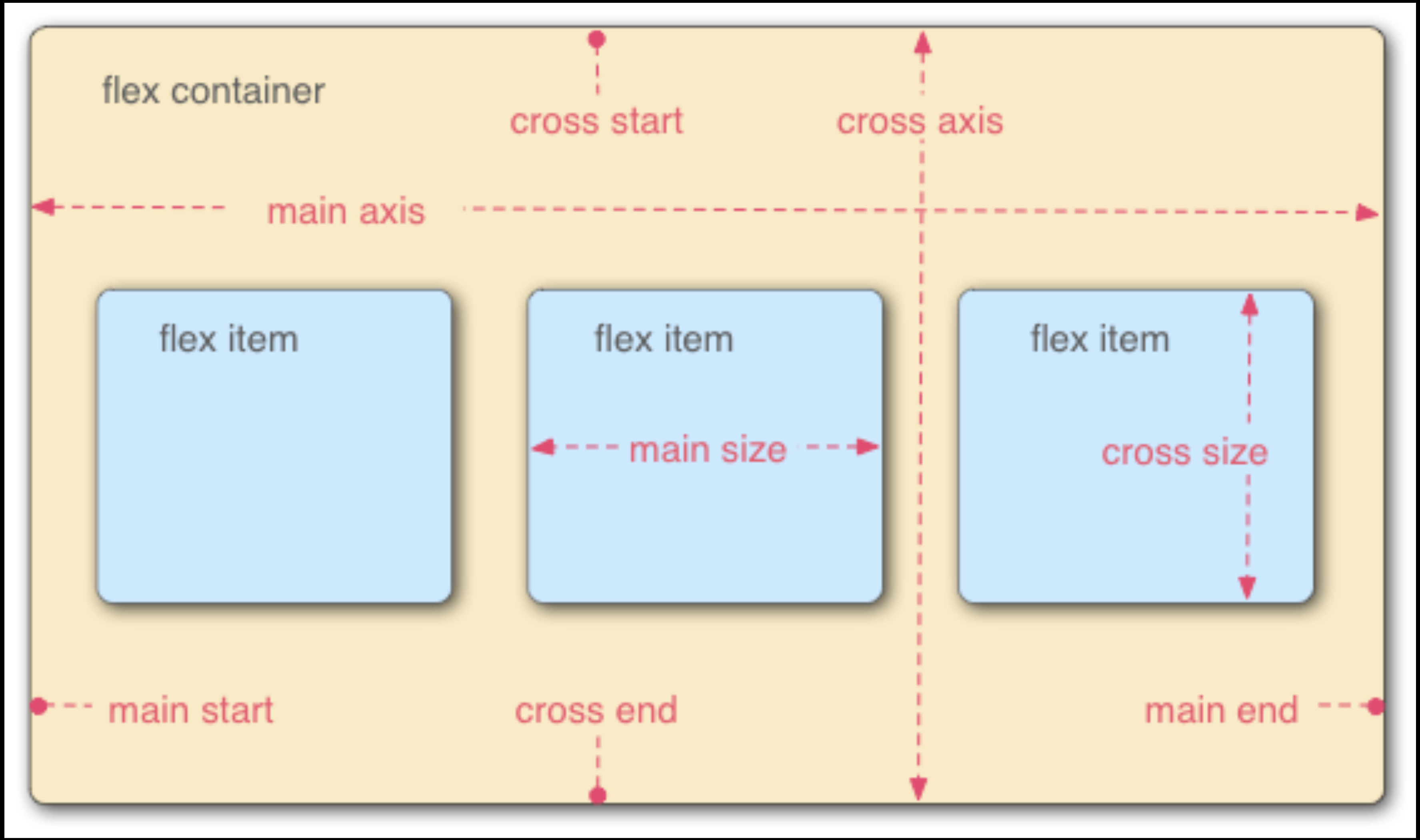
What is Flexbox?

Flexbox is a one-dimensional layout method for arranging items in rows or columns. Items *flex* (expand) to fill additional space or shrink to fit into smaller spaces.



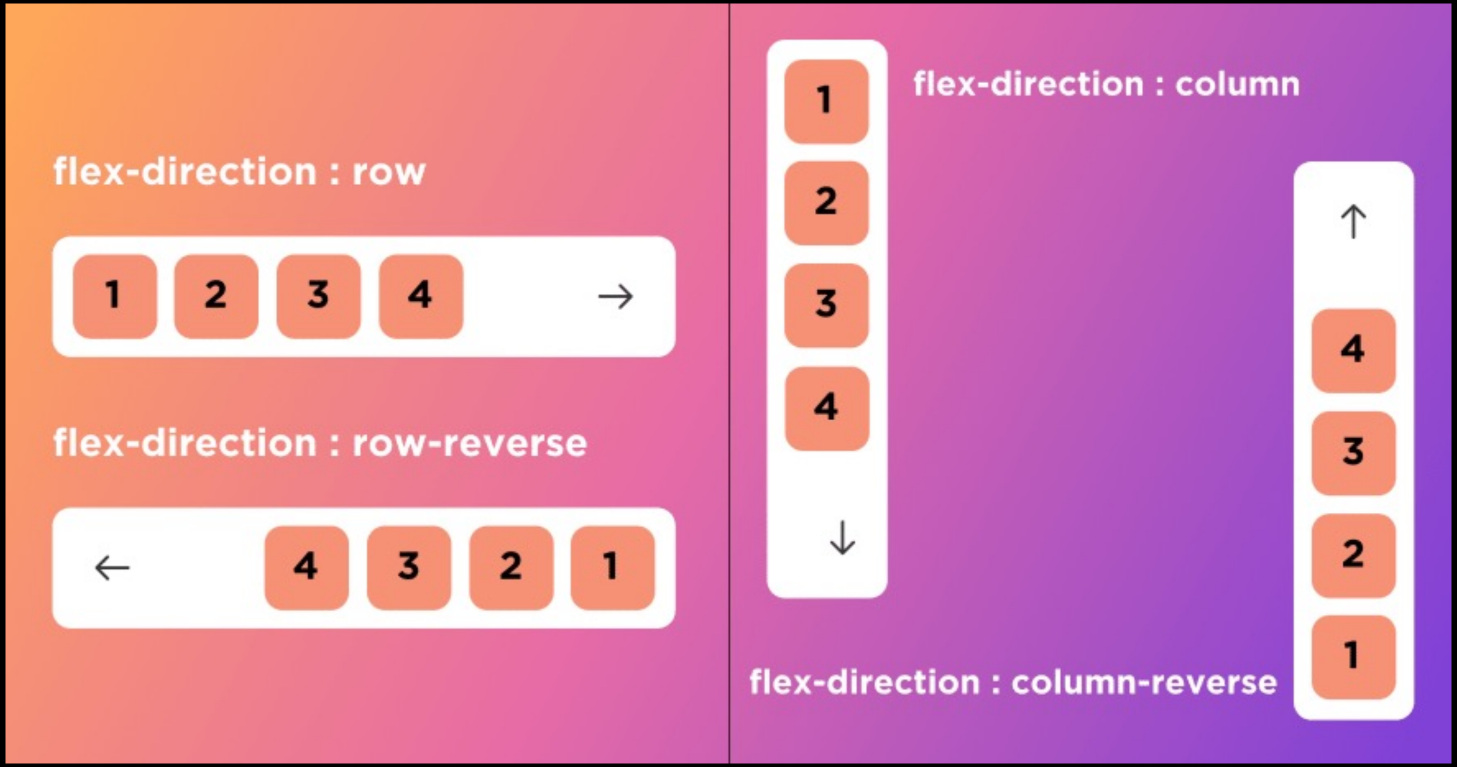
Flex Model



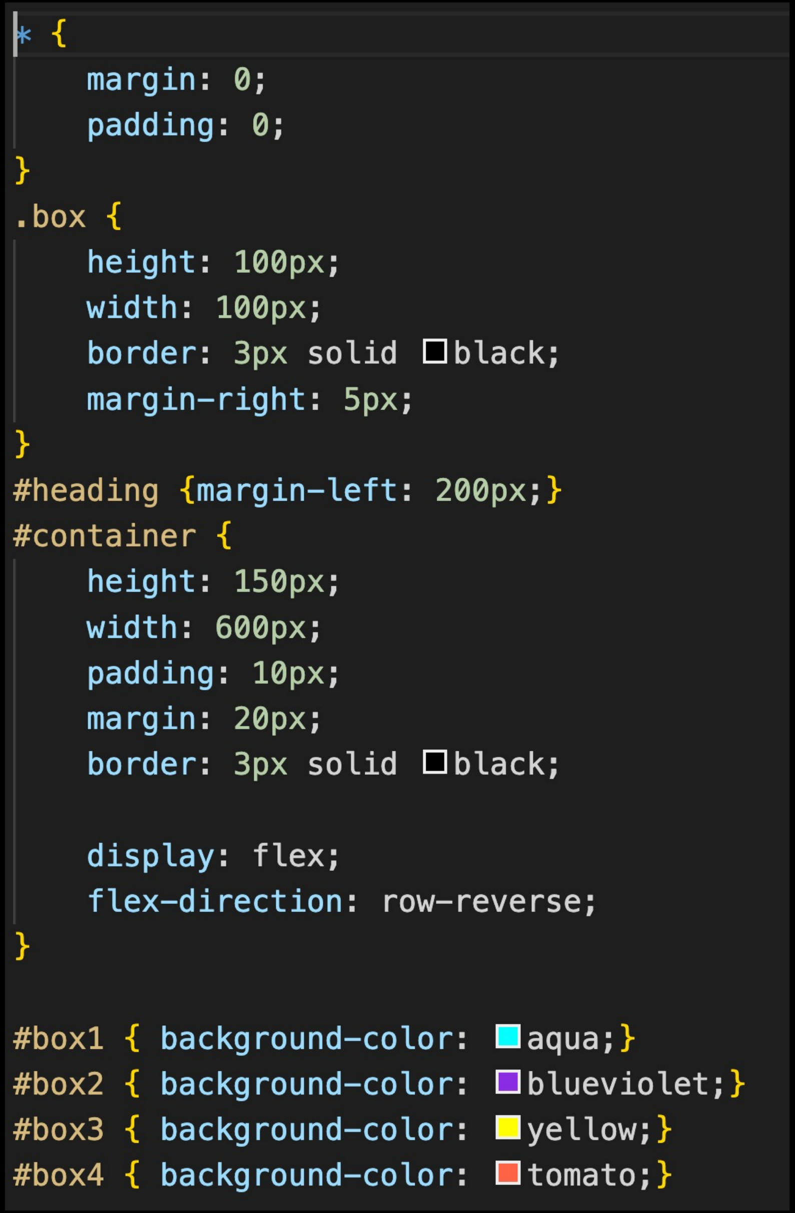
display: flex

Flexbox Direction

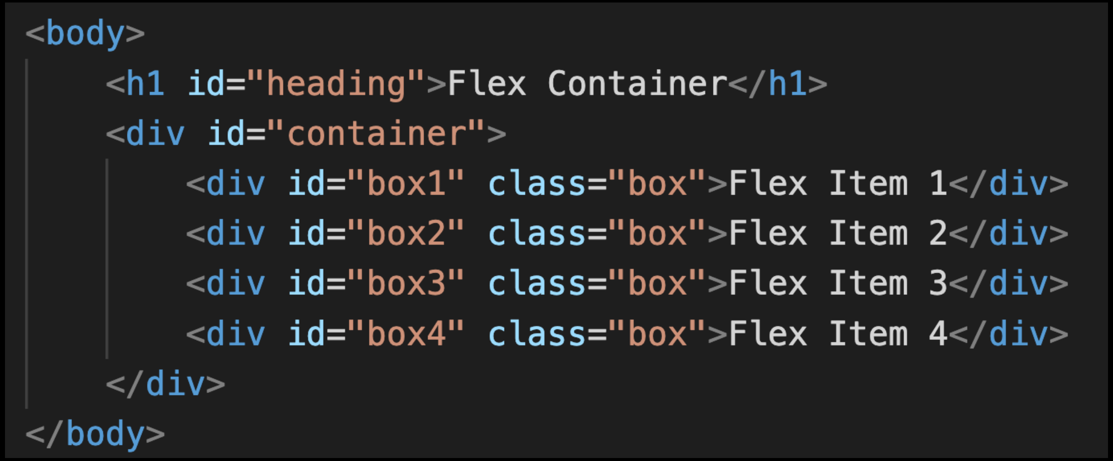
- **Property Name:** flex-direction is the property used to define the direction in a flex container.
- **Row Layout:** row value aligns the flex items horizontally, in a left-to-right fashion.
- **Column Layout:** column value stacks the flex items vertically, from top to bottom.
- **Reverse Direction:** Adding -reverse to row or column (as in row-reverse or column-reverse) reverses the order of the items.



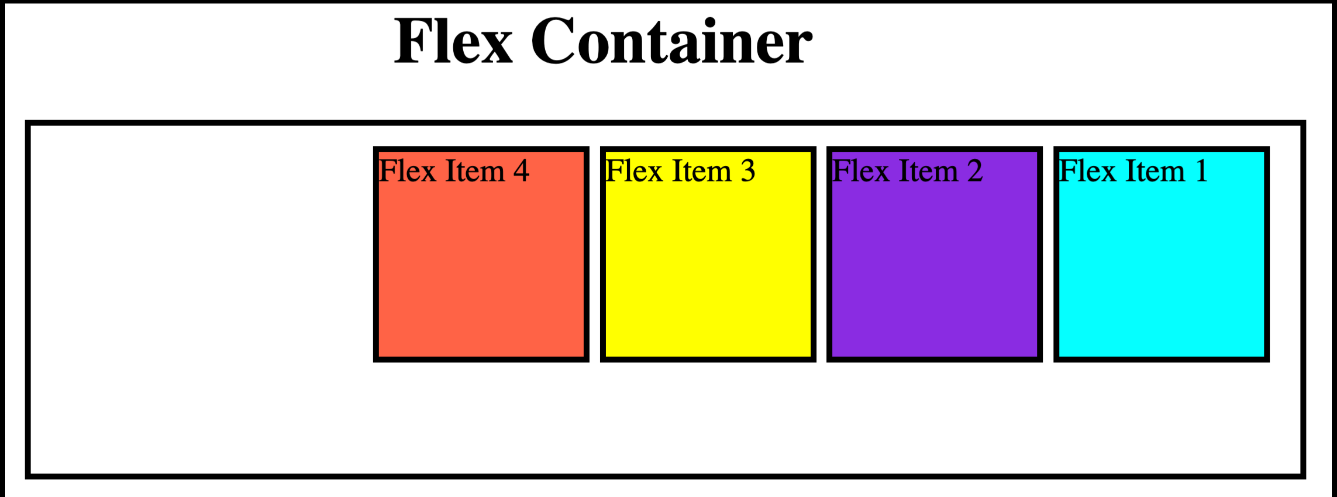
Flexbox Direction



```css
* {
    margin: 0;
    padding: 0;
}
.box {
    height: 100px;
    width: 100px;
    border: 3px solid ■black;
    margin-right: 5px;
}
#heading {margin-left: 200px;}
#container {
    height: 150px;
    width: 600px;
    padding: 10px;
    margin: 20px;
    border: 3px solid ■black;

    display: flex;
    flex-direction: row-reverse;
}

#box1 { background-color: 🟦aqua;}
#box2 { background-color: 🟪blueviolet;}
#box3 { background-color: 🟨yellow;}
#box4 { background-color: 🟥tomato;}
```



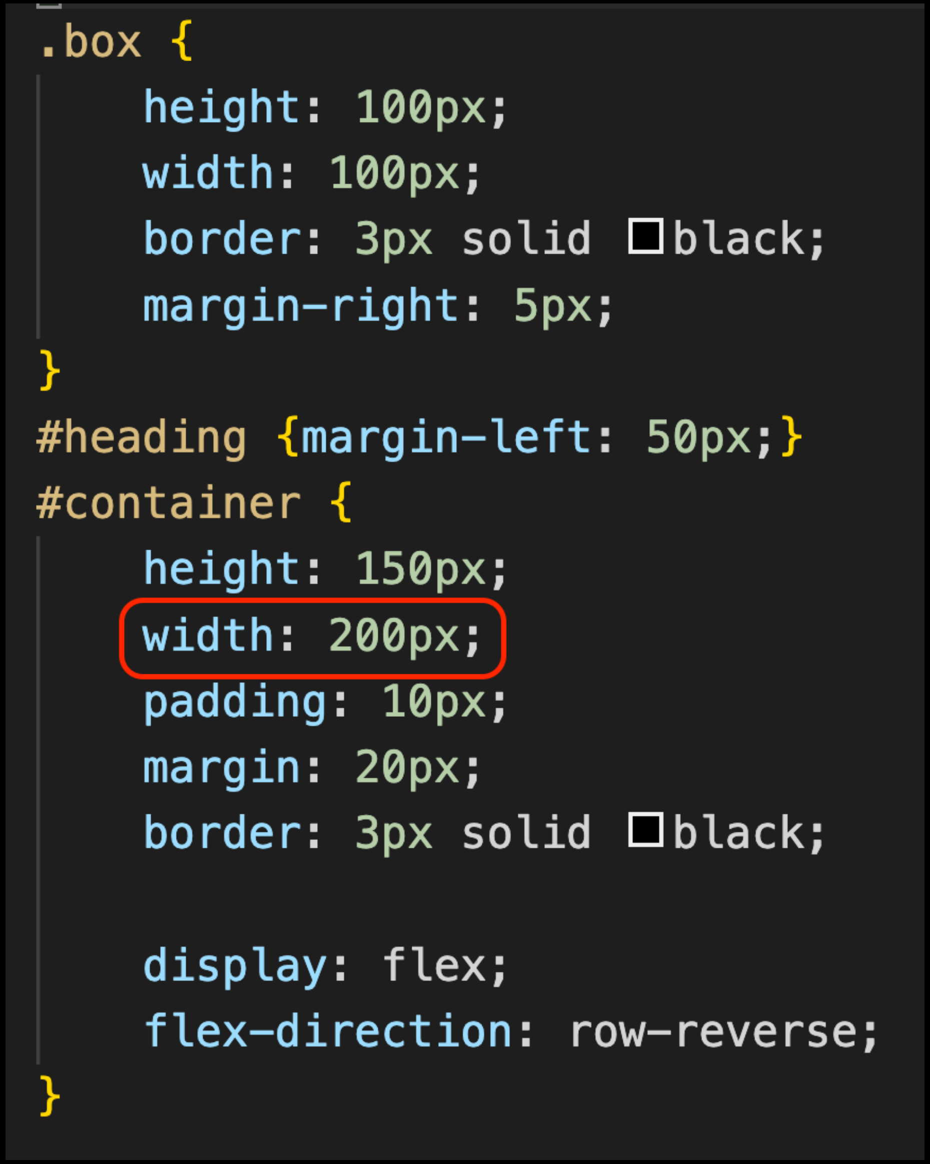
```html
<body>
    <h1 id="heading">Flex Container</h1>
    <div id="container">
        <div id="box1" class="box">Flex Item 1</div>
        <div id="box2" class="box">Flex Item 2</div>
        <div id="box3" class="box">Flex Item 3</div>
        <div id="box4" class="box">Flex Item 4</div>
    </div>
</body>
```



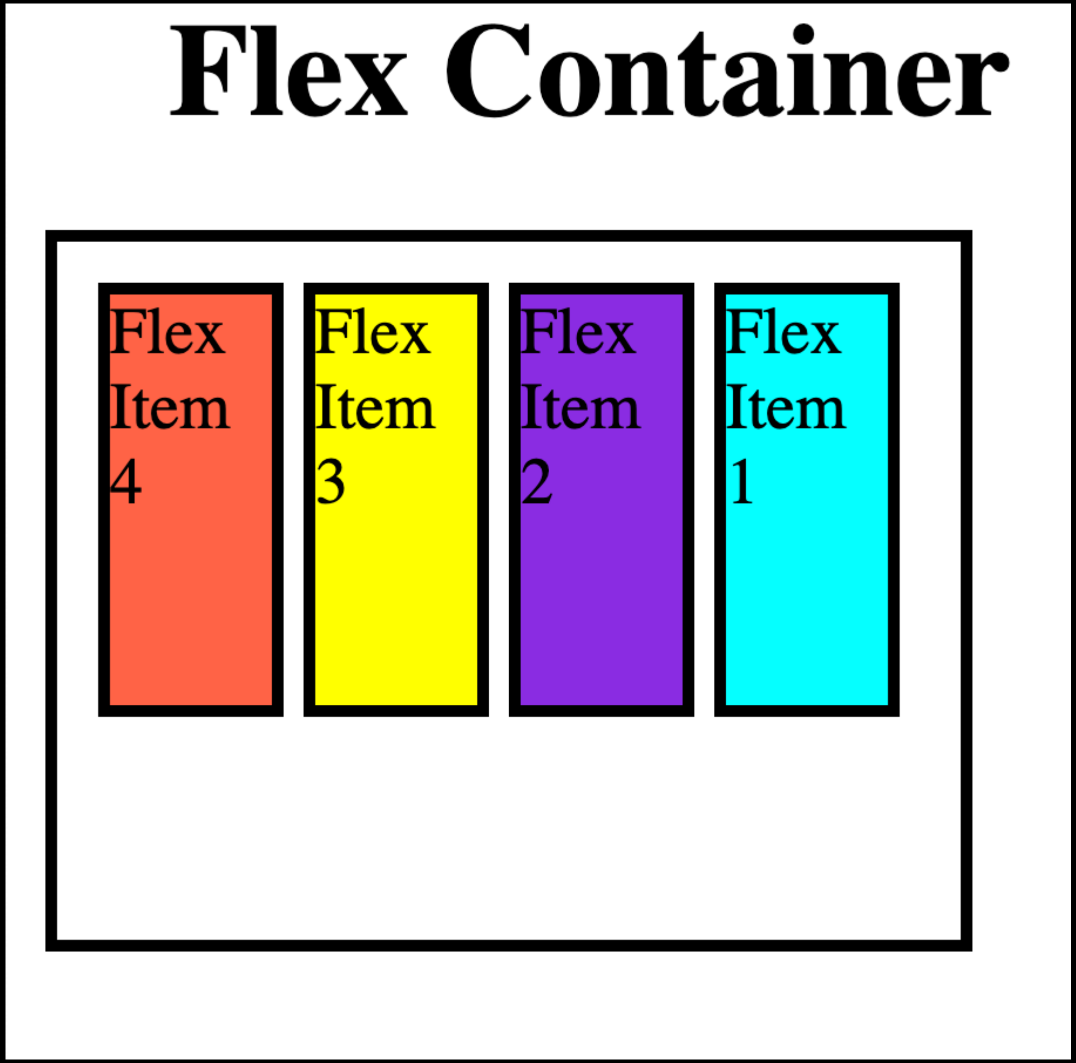
Flex Container

Flex Item 4 Flex Item 3 Flex Item 2 Flex Item 1

Flexbox Direction



```
.box {
    height: 100px;
    width: 100px;
    border: 3px solid □black;
    margin-right: 5px;
}
#heading {margin-left: 50px;}
#container {
    height: 150px;
    width: 200px;
    padding: 10px;
    margin: 20px;
    border: 3px solid □black;

    display: flex;
    flex-direction: row-reverse;
}
```



Flex Container

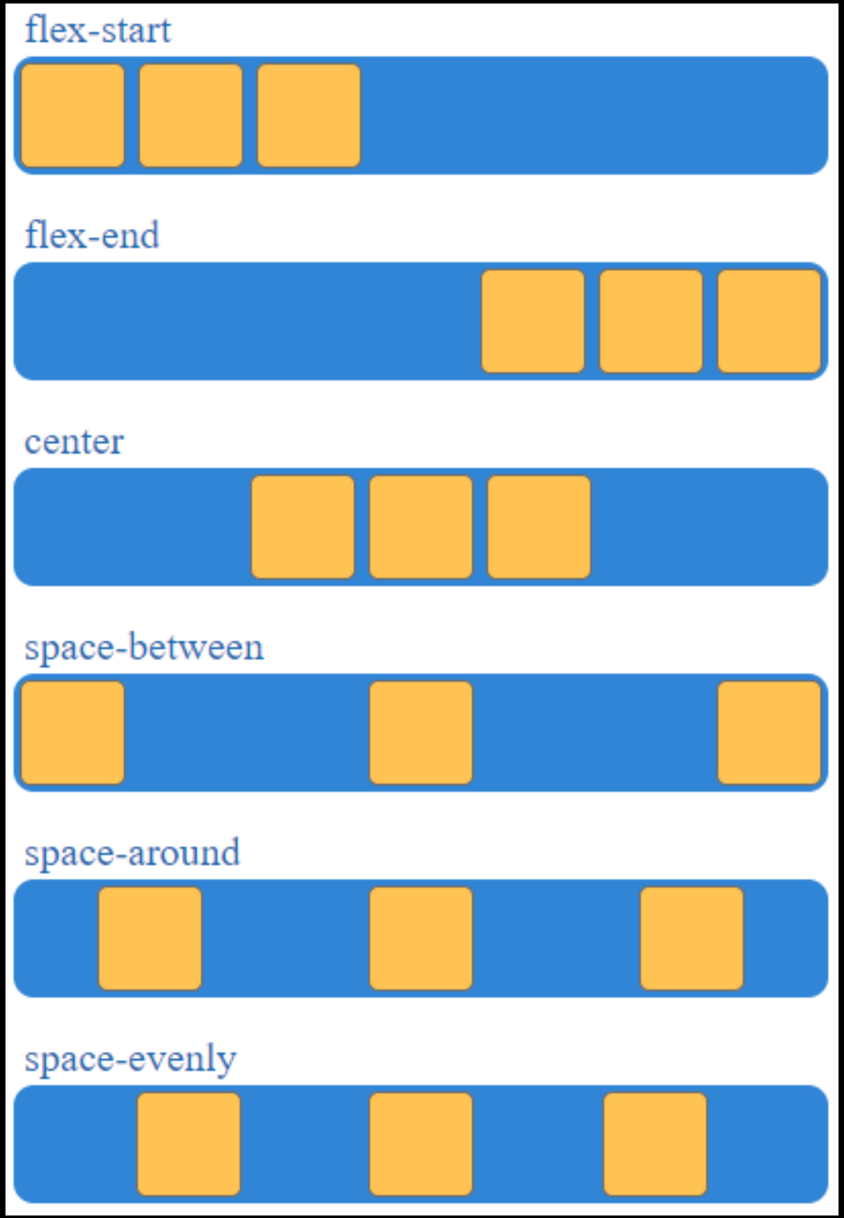
Flex Item 4

Flex Item 3

Flex Item 2

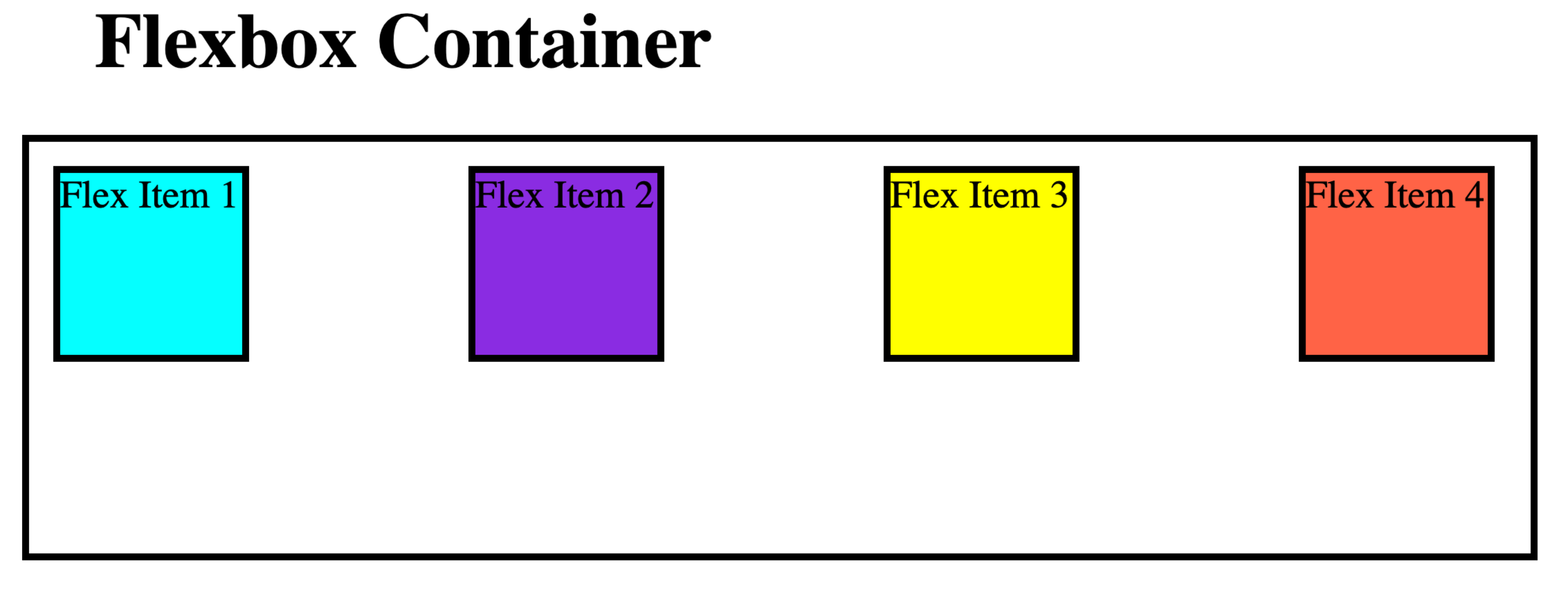
Flex Item 1

Properties: Flexbox container (Justify Content)

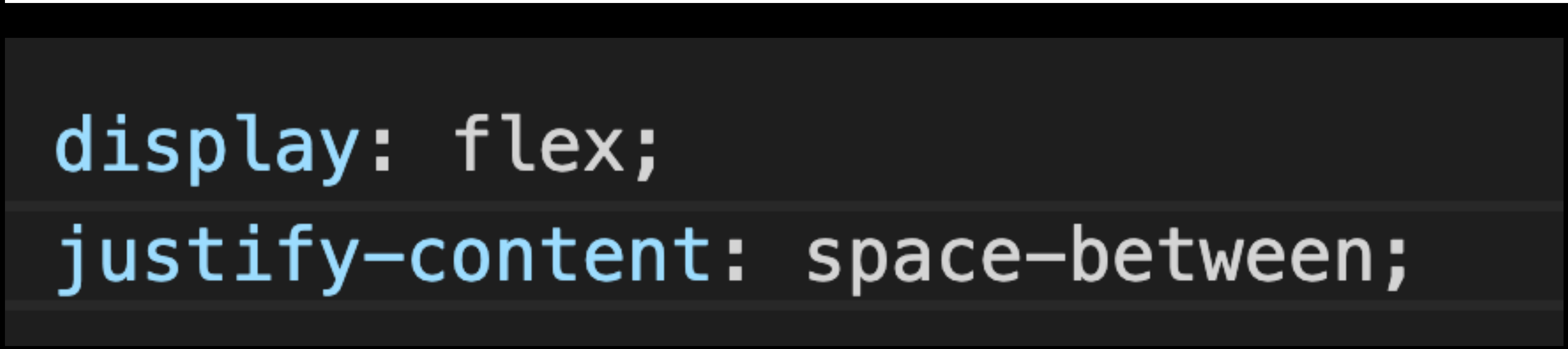


- **Alignment**: Aligns flex items along the main axis.
- **flex-start**: Items align to the start of the flex container.
- **flex-end**: Items align to the end of the flex container.
- **Center**: Items are cantered within the flex container.
- **space-between/space-around/space-evenly**: Distributes space between items evenly.

Properties: Flexbox container (Justify Content)

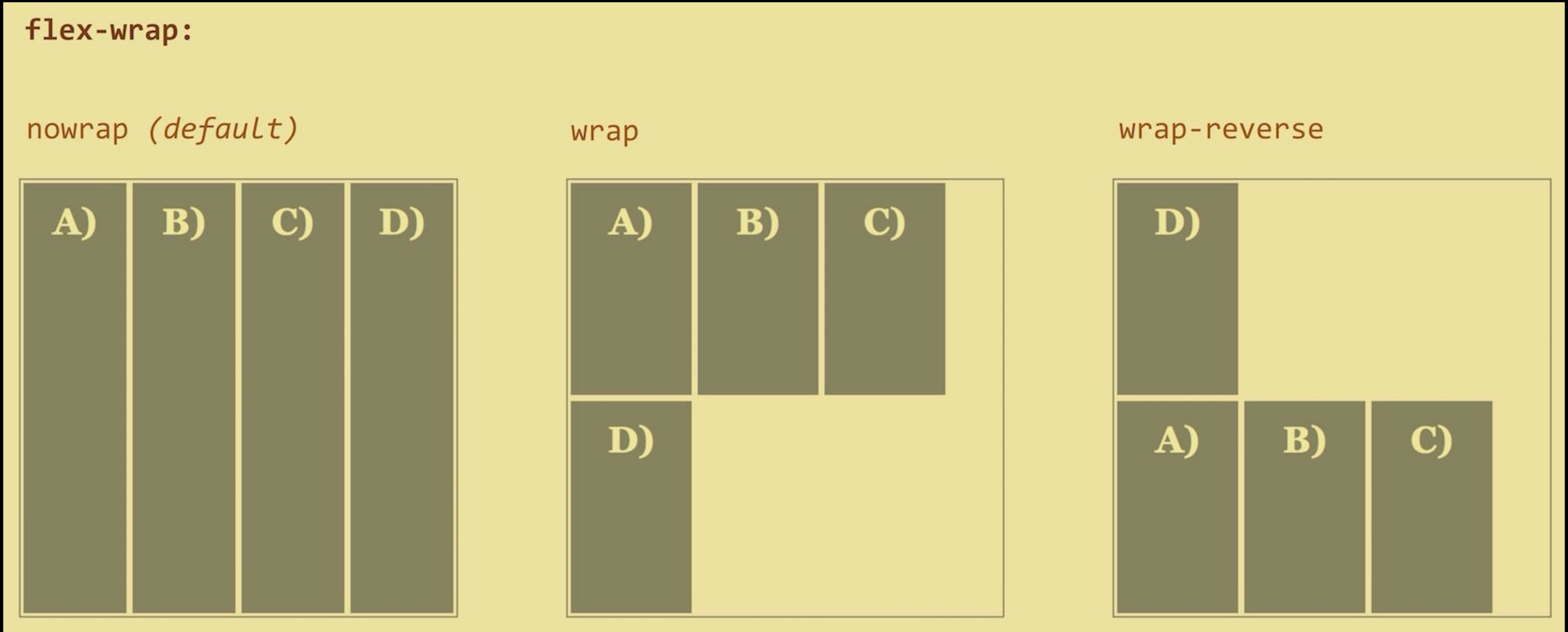


Flexbox Container

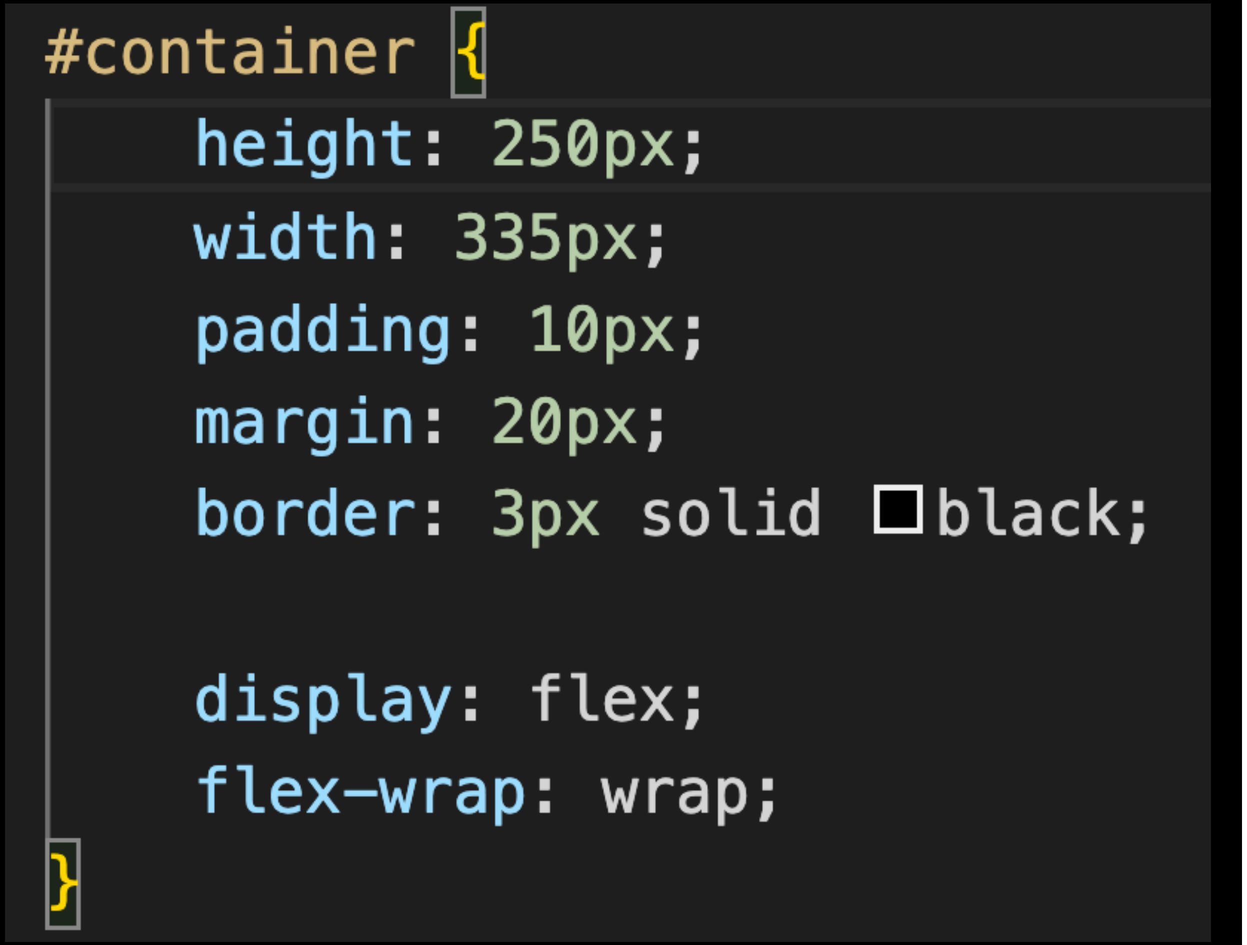


```
display: flex;
justify-content: space-between;
```

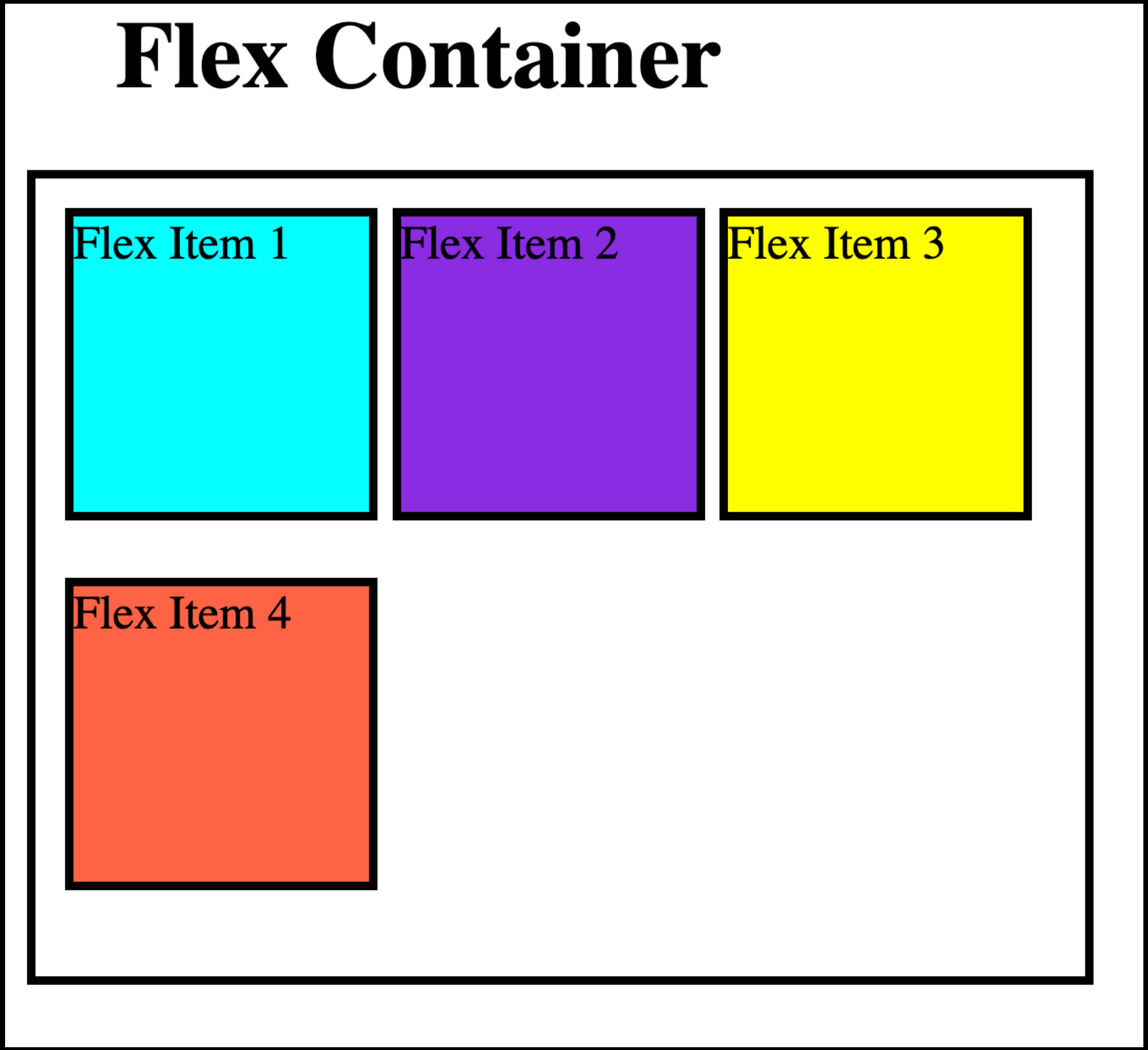
Properties: Flexbox container (Flex Wrap)



Properties: Flexbox container (Flex Wrap)



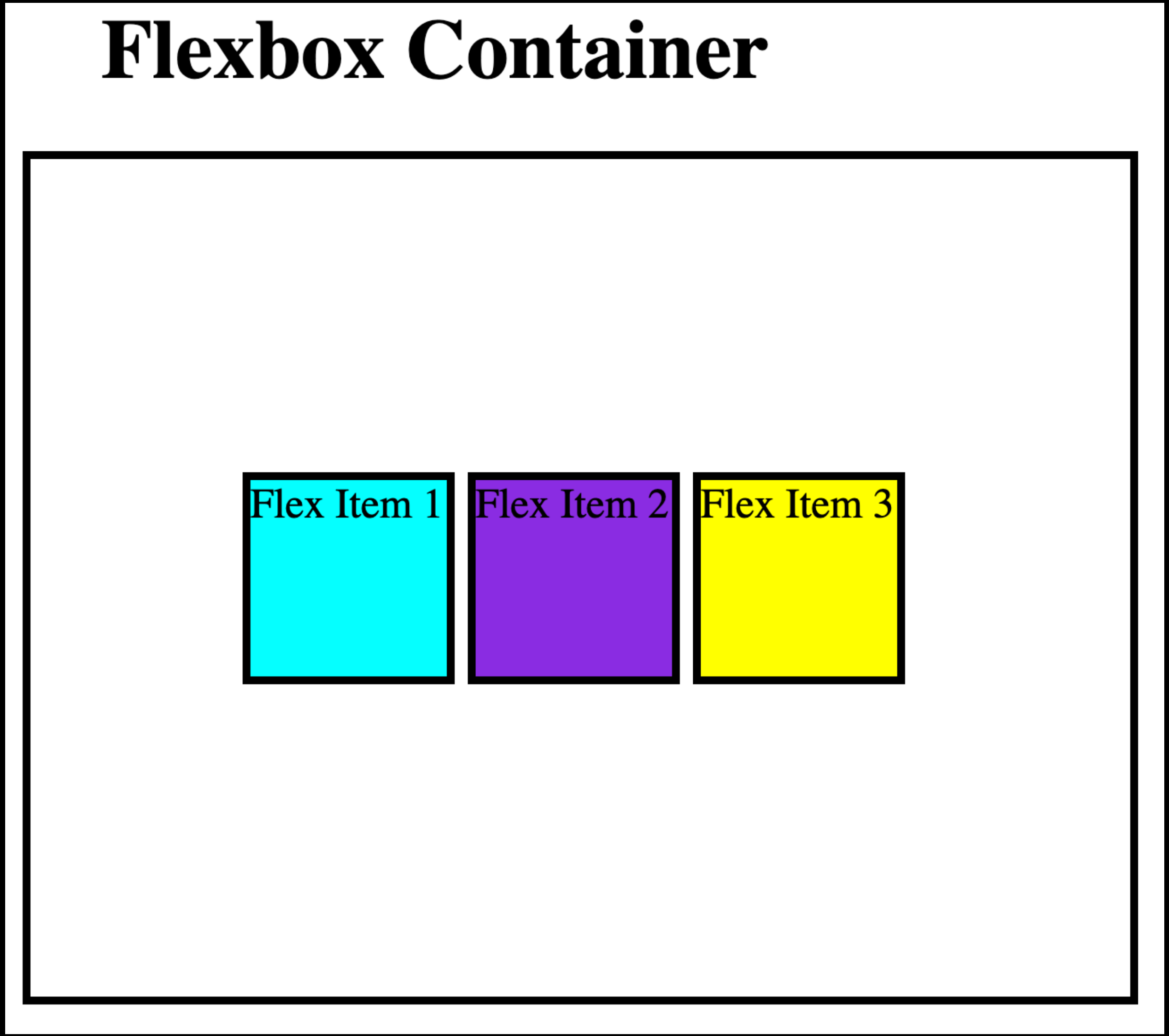
```css
#container {
    height: 250px;
    width: 335px;
    padding: 10px;
    margin: 20px;
    border: 3px solid ■black;

    display: flex;
    flex-wrap: wrap;
}
```



Properties: Flexbox container (Align Items)

This property is used to align the flex container's items along the cross-axis, which is perpendicular to the main axis.

```
display: flex;
flex-direction: row;
justify-content: center;
align-items: center;
```



Flexbox Container

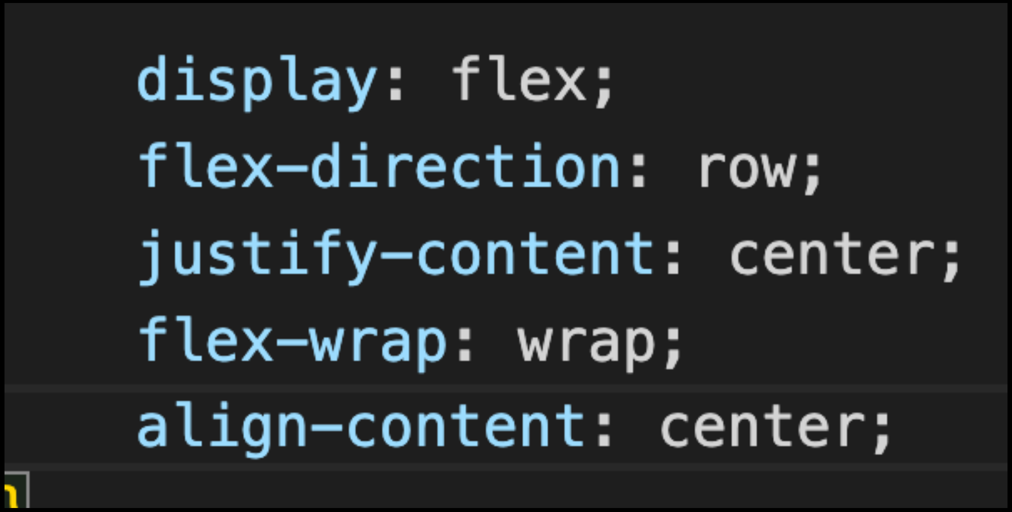
Flex Item 1

Flex Item 2

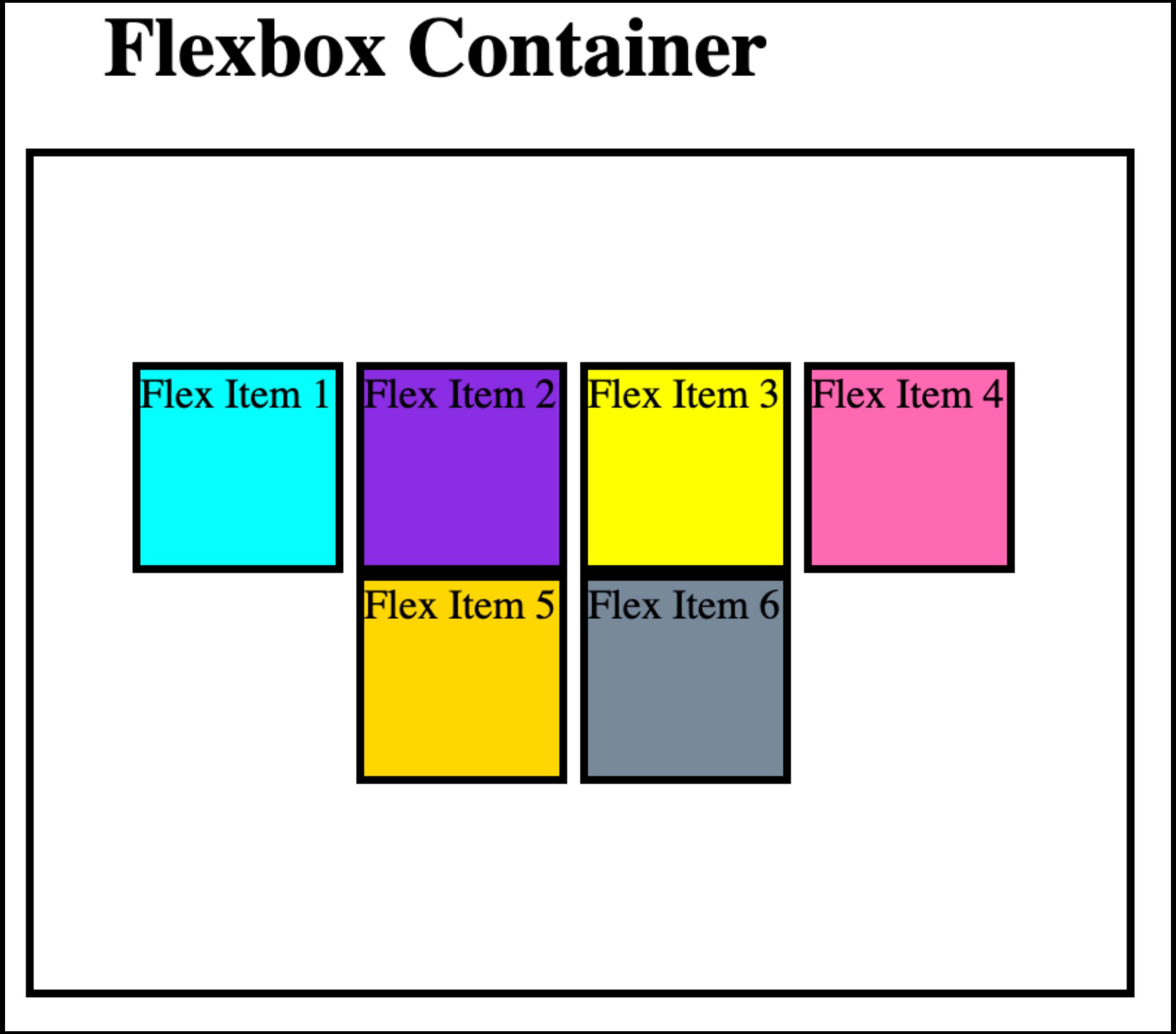
Flex Item 3

Properties: Flexbox container (Align Content)

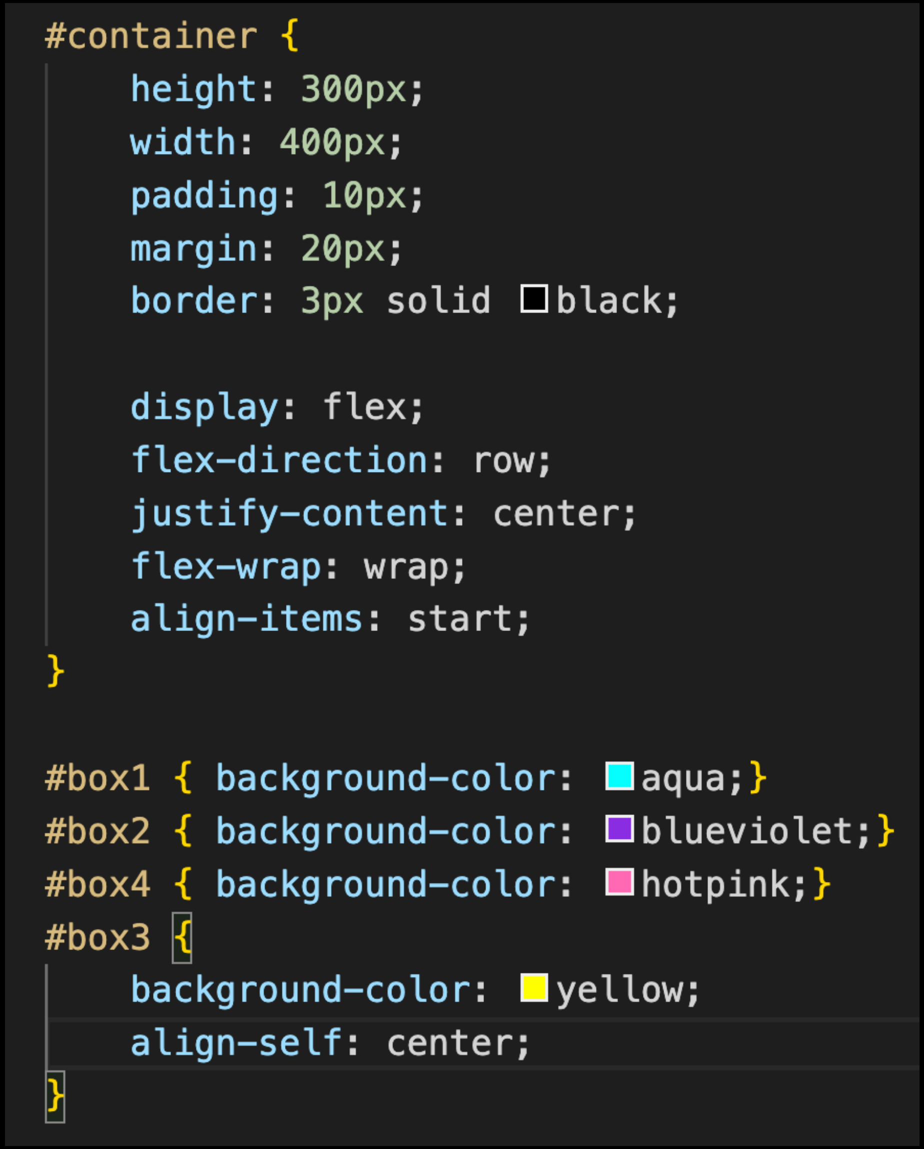
It is utilized to adjust the spacing between flex lines within a flex container, particularly when there is extra space along the cross-axis.



```
    display: flex;
    flex-direction: row;
    justify-content: center;
    flex-wrap: wrap;
    align-content: center;
```

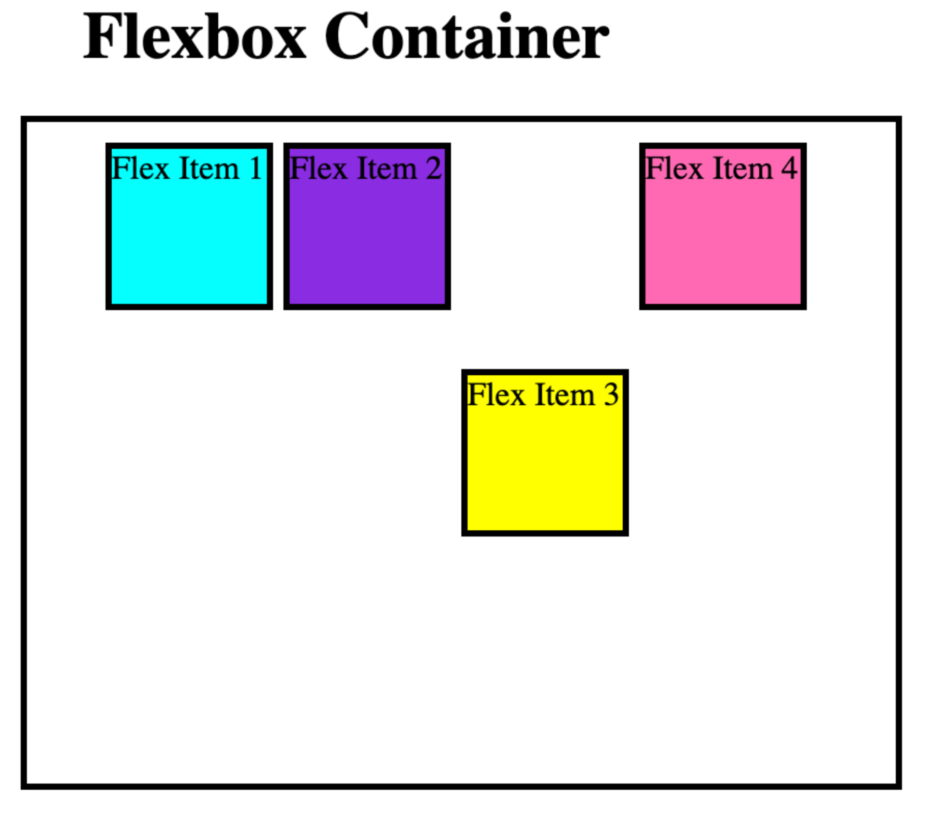


Properties: Flex Items (Align Self)



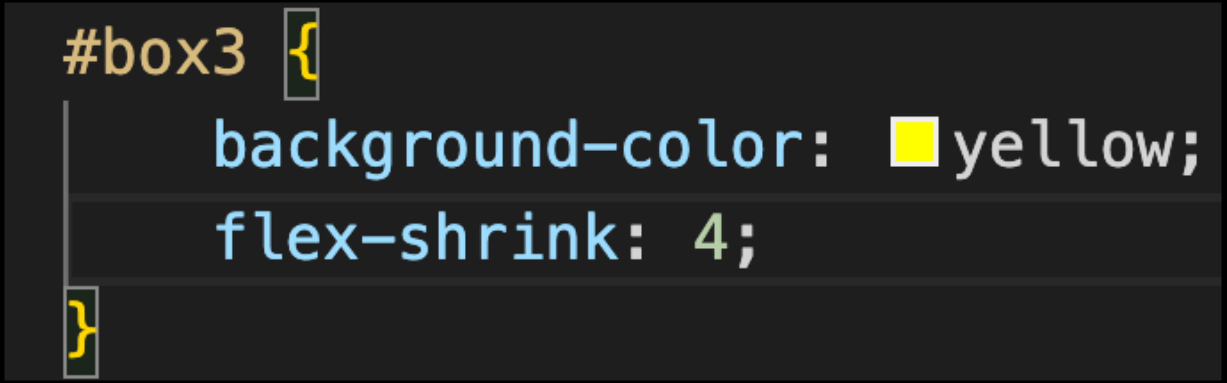
```
#container {
    height: 300px;
    width: 400px;
    padding: 10px;
    margin: 20px;
    border: 3px solid black;

    display: flex;
    flex-direction: row;
    justify-content: center;
    flex-wrap: wrap;
    align-items: start;
}

#box1 { background-color: aqua;}
#box2 { background-color: blueviolet;}
#box4 { background-color: hotpink;}
#box3 {
    background-color: yellow;
    align-self: center;
}
```

Allows individual flex items to override the container's align-items property, aligning them differently along the cross-axis.



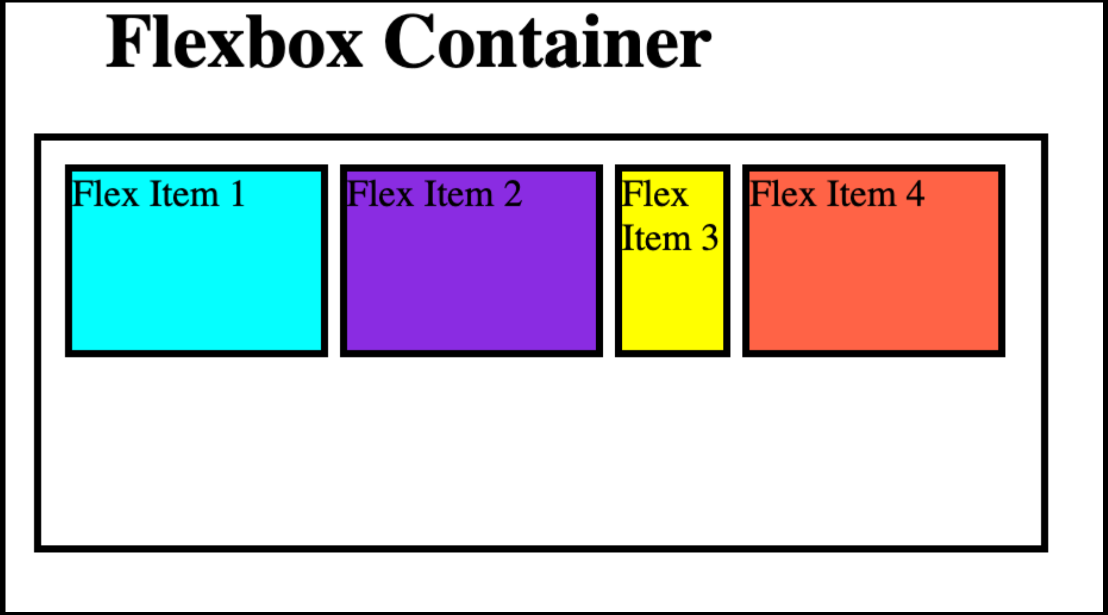
Flexbox Container

Properties: Flex Items (Flex Shrink)

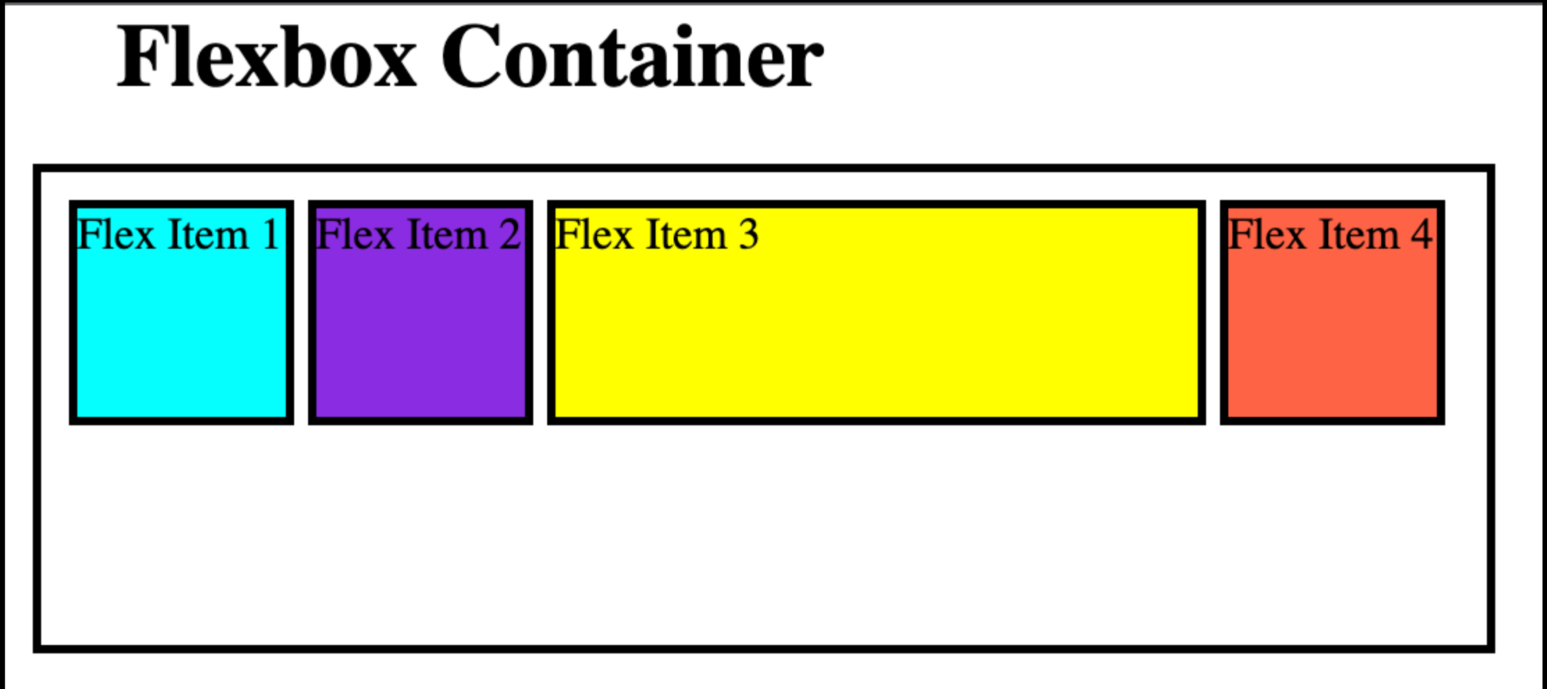


```
#box3 {
    background-color: yellow;
    flex-shrink: 4;
}
```

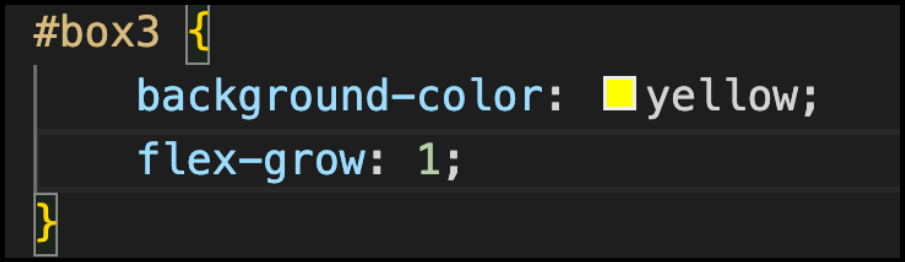
The "flex-shrink" property in CSS determines how much a flex item will shrink relative to other items in the flex container if there is insufficient space.



Properties: Flex Items (Flex Grow)



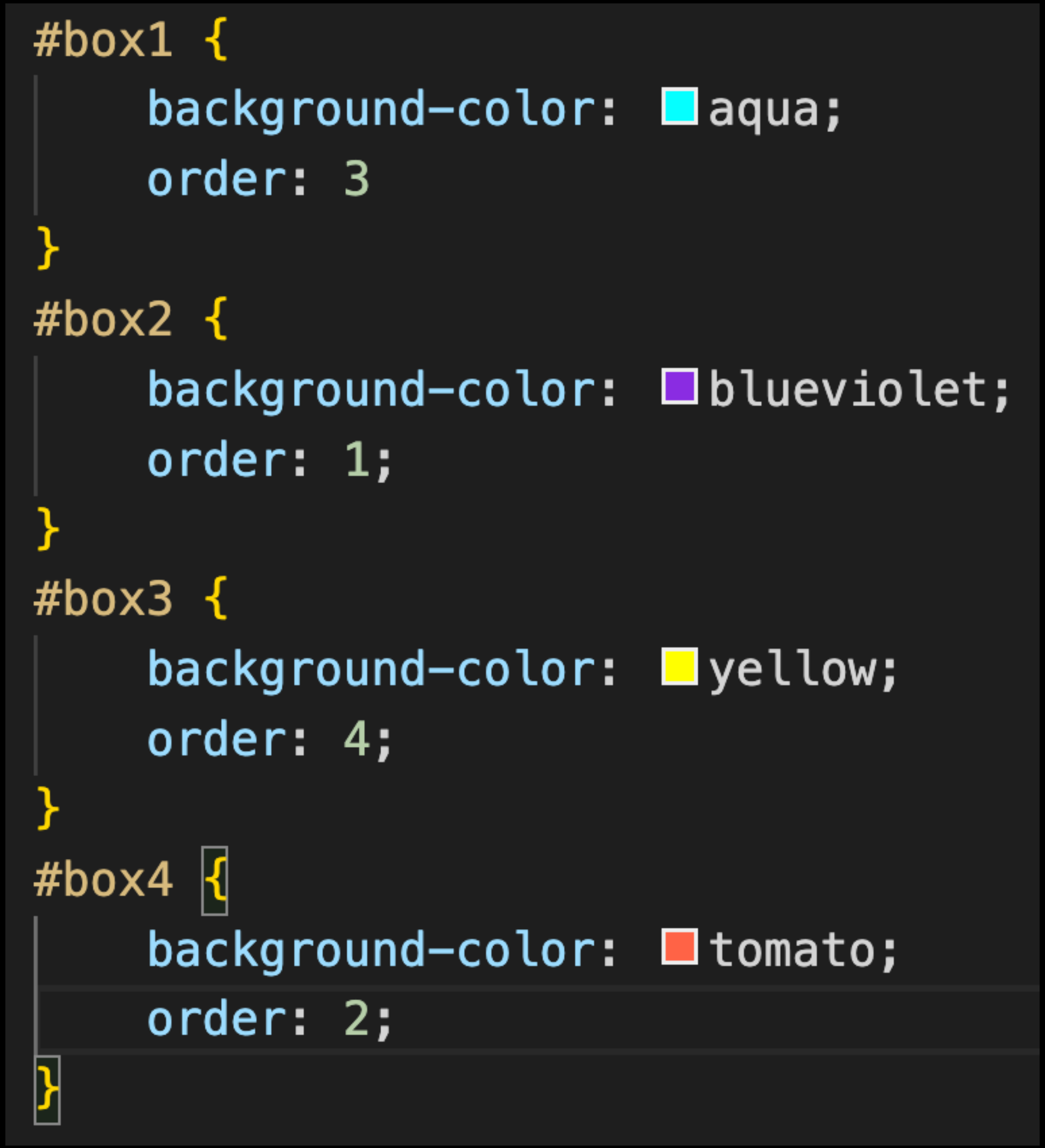
Flexbox Container



```
#box3 {
    background-color: yellow;
    flex-grow: 1;
}
```

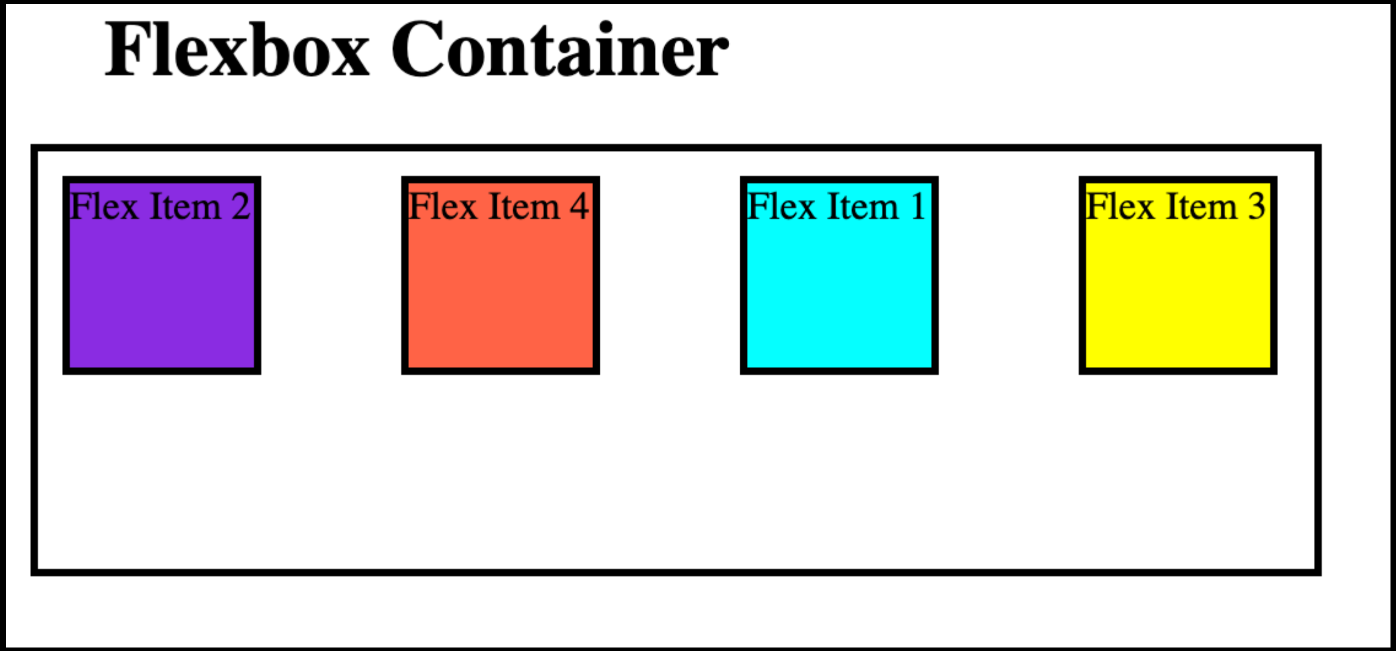
The "flex-grow" property in CSS specifies how much a flex item will grow relative to other items in the flex container when additional space is available.

Properties: Flex Items (Order)



```
#box1 {
    background-color: aqua;
    order: 3
}
#box2 {
    background-color: blueviolet;
    order: 1;
}
#box3 {
    background-color: yellow;
    order: 4;
}
#box4 {
    background-color: tomato;
    order: 2;
}
```

The "order" property in CSS allows you to define the sequence in which flex items appear within the flex container, overriding their original order in the HTML.



Practice Set

Flex Box

- Create the same nav bar created in the last Practice Set but do the spacing using flexbox.
- Use flexbox to centre an image inside a div.
- Create a div with three boxes where two boxes have fixed size, but the third box grows and shrinks with container size.