Sure, here's an example of how you can use the CSS box model to create a multi-column layout using flexbox:

HTML:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Multi-column Layout</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <div class="container">
        <div class="column">
            <h2>Column 1</h2>
            <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis mollis euismod felis, vel dapibus lectus ultrices non. Nam eget mauris ac velit tempus suscipit.</p>
        </div>
        <div class="column">
            <h2>Column 2</h2>
            <p>Integer vitae ipsum in quam tristique venenatis. Maecenas gravida, eros a viverra congue, urna dui convallis odio, non congue tortor lectus id mi.</p>
        </div>
        <div class="column">
            <h2>Column 3</h2>
            <p>Quisque bibendum, lacus sit amet rutrum pharetra, eros nisl faucibus turpis, id vestibulum est nisi a felis. Ut nec dapibus justo, id fermentum ex.</p>
        </div>
    </div>
</body>
</html>
```

CSS (styles.css):

```css
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
}

.container {
    display: flex;
    justify-content: space-between;
    padding: 20px;
}

.column {
    flex: 0 1 30%; /* Flex-basis: 30% to allow for 3 columns */
    background-color: #f0f0f0;
    padding: 20px;
    margin-right: 20px; /* Adds margin between columns */
    box-sizing: border-box; /* Ensures padding is included in the width */
}

.column:last-child {
    margin-right: 0; /* Remove margin from last column */
}
```

```
h2 {
    margin-top: 0;
}
```

This code will create a multi-column layout with three columns, each with some padding and margin between them. The flex property in the .column class ensures that each column takes up an equal amount of space. The box-sizing: border-box; property ensures that padding is included in the width calculation.