CSS Flexbox - Flex Items

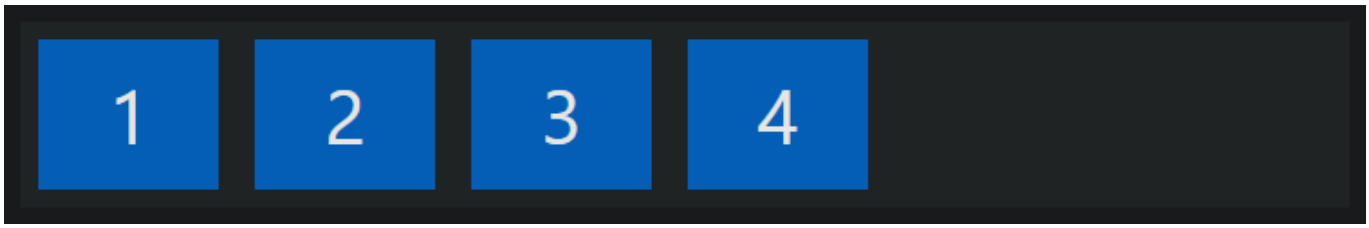
- description : CSS Lists,
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-07-16

The source of this article

CSS Flexbox - Flex Items

Child Elements (Items)

flex container의 직계 자식 요소들은 자동으로 flexible (flex) items가 됩니다.



위의 요소는 그레이 컬러의 flex container 내부에 4개의 블루 컬러의 flex items를 나타냅니다.

```
<style>
  .flex-container {
    display: flex;
    background-color: #f1f1f1;
  }

  .flex-container > div {
    background-color: Dodgerblue;
    color: #fff;
    width: 100px;
    margin: 10px;
    text-align: center;
    line-height: 75px;
    font-size: 30px;
  }
```

```
</style>
</head>
<body>

  <h1>Flexible Items</h1>

  <div class="flex-container">
    <div>1</div>
    <div>2</div>
    <div>3</div>
    <div>4</div>
  </div>

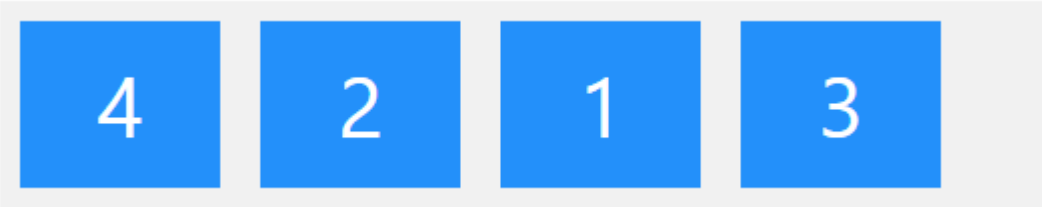
  <p>All direct children of a flexible container becomes flexible items.</p>
</body>
</html>
```

flex item 속성들은 하기와 같습니다.

- order
- flex-grow
- flex-shrink
- flex-basis
- flex
- align-self

The order Property

order 속성은 flex items의 순서를 지정합니다.



코드 상의 첫 번째 flex item은 레이아웃에서 첫 번째 아이템으로 나타날 필요는 없습니다.
order값은 숫자이어야 하고, 기본 값은 0 입니다.

Example

/order/ 속성은 flex items의 순서를 변경할 수 있습니다.

```
  <style>
    .flex-container {
```

```
      display: flex;
      align-items: stretch;
      background-color: #f1f1f1;
    }

    .flex-container>div {
      background-color: Dodgerblue;
      color: white;
      width: 100px;
      margin: 10px;
      text-align: center;
      line-height: 75px;
      font-size: 30px;
    }
  </style>
</head>

<body>

  <h1>The order Property</h1>

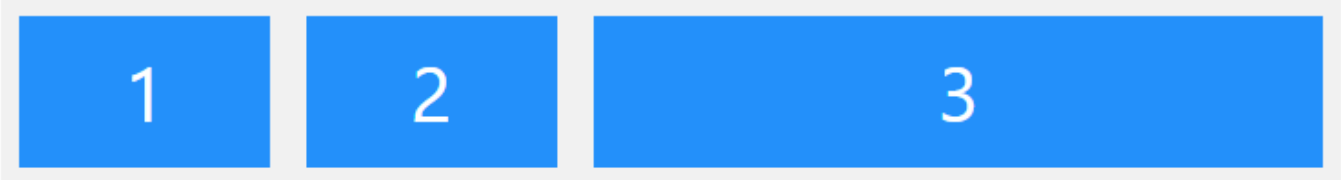
  <p>Use the order property to sort the flex items as you like:</p>

  <div class="flex-container">
    <div style="order: 3">1</div> <!-- 1번 박스를 세 번째로 -->
    <div style="order: 2">2</div> <!-- 2번 박스를 두 번째로 -->
    <div style="order: 4">3</div> <!-- 3번 박스를 네 번째로 -->
    <div style="order: 1">4</div> <!-- 4번 박스를 첫 번째로 -->
  </div>

</body>
```

The flex-grow Property

`flex-grow` 속성은 어떤 하나의 flex item이 나머지 flex items에 비례하여 얼마나 많이 확장될 수 있는지를 지정합니다.



값은 숫자이어야 하며, 기본 값은 0입니다.

```
  <style>
    .flex-container {
      display: flex;
      align-items: stretch;
```

```
      background-color: #f1f1f1;
    }

    .flex-container>div {
      background-color: dodgerblue;
      color: white;
      margin: 10px;
      /* 박스의 너비가 지정되지 않았습니다. */
      text-align: center;
      line-height: 75px;
      font-size: 30px;
    }
  </style>

</head>

<body>
  <h1>The flex-grow Property</h1>

  <p>Make the third flex item grow eight times faster than the other flex items:</p>

  <div class="flex-container">
    <div style="flex-grow: 1">1</div>
    <div style="flex-grow: 1">2</div>
    <div style="flex-grow: 8">3</div>
    <!-- 3번 박스는 브라우저 창의 너비와 상관없이 항상 1번, 2번 박스에 비해 8배 너비를 유지합니다. -->
  </div>
</body>
```

The flex-shrink Property

`flex-shrink` 속성은 어떤 하나의 flex item이 나머지 flex item에 비례하여 얼마나 많이 축소될 수 있는 지를 지정합니다.



값은 숫자이어야 하며, 기본 값은 1 입니다.

Example

세 번째 flex item이 나머지 다른 flex items만큼 축소되지 않게 하는 예제입니다

```
  <style>
    .flex-container {
      display: flex;
      align-items: stretch;
      background-color: #f1f1f1;
    }

    .flex-container>div {
      background-color: dodgerblue;
      color: white;
      width: 100px;
      margin: 10px;
      text-align: center;
      line-height: 75px;
      font-size: 30px;
    }
  </style>
</head>

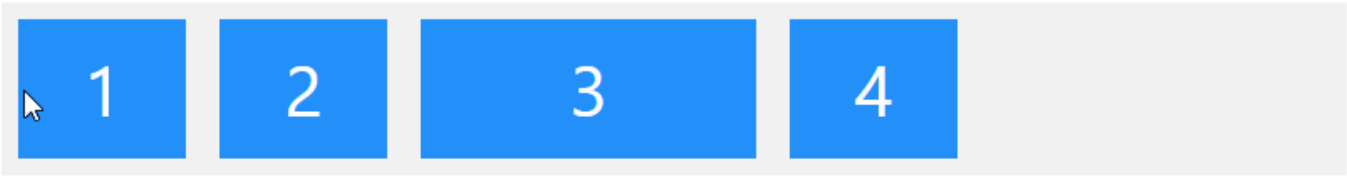
<body>
  <h1>The flex-shrink Property</h1>

  <p>Do not let the third flex item shrink as much as the other flex
items:</p>

  <div class="flex-container">
    <div>1</div>
    <div>2</div>
    <div style="flex-shrink: 0">3</div>
    <div>4</div>
    <div>5</div>
    <div>6</div>
    <div>7</div>
    <div>8</div>
    <div>9</div>
    <div>10</div>
  </div>
</body>
```

The flex-basis Property

`flex-basis` 속성은 어떤 하나의 flex item에 대한 초기 길이 값을 지정합니다.



Example

세 번째 flex item의 초기 길이 값을 200px로 설정합니다.

```
  <style>
    .flex-container {
      display: flex;
      align-items: stretch;
      background-color: #f1f1f1;
    }

    .flex-container>div {
      background-color: dodgerblue;
      color: #fff;
      width: 100px;
      margin: 10px;
      text-align: center;
      line-height: 75px;
      font-size: 30px;
    }
  </style>
</head>

<body>
  <h1>The flex-basis Property</h1>

  <p>Set the initial length of the third flex item to 200 pixels:</p>

  <div class="flex-container">
    <div>1</div>
    <div>2</div>
    <div style="flex-basis: 200px">3</div>
    <div>4</div>
  </div>
</body>
```

The flex Property

flex 속성은 flex-grow, flex-shrink, 그리고 flex-basis 속성들에 대한 약칭 속성입니다.

Example

하기 예제는 세 번째 flex item이 확장되지도, 축소되지도 않게 만들고, 초기 길이 값 200pixel을 가지게 합니다.

```
  <style>
    .flex-container {
      display: flex;
      align-items: stretch;
      background-color: #f1f1f1;
    }

    .flex-container>div {
      background-color: dodgerblue;
      color: #fff;
      width: 100px;
      margin: 10px;
      text-align: center;
      line-height: 75px;
      font-size: 30px;
    }
  </style>
</head>

<body>
  <h1>The flex Property</h1>

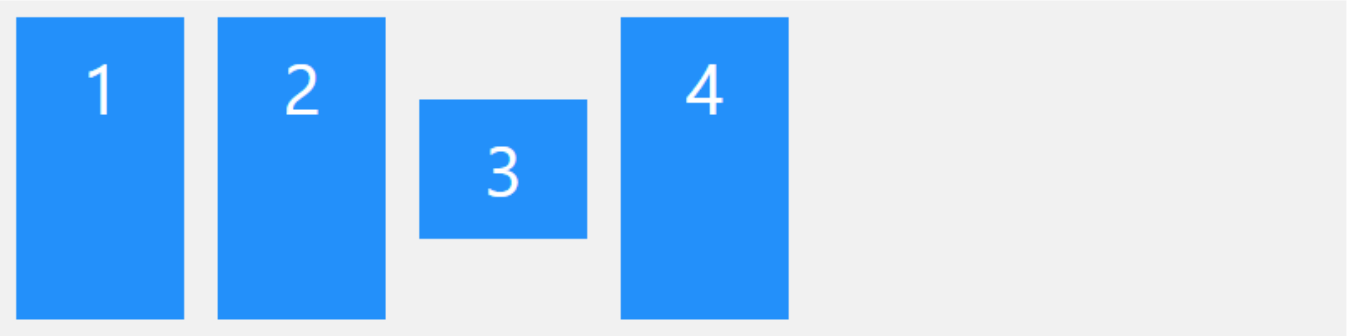
  <p>
    Make the third flex item not growable (0), not shrinkable (0), and with
an initial length 0f 200 pixels:
  </p>

  <div class="flex-container">
    <div>1</div>
    <div>2</div>
    <div style="flex: 0 0 200px">3</div>
    <div>4</div>
  </div>
</body>
```

The align-self Property

`align-self` 속성은 플렉서블 컨테이너 내부에서 선택한 플렉스 아이템에 대한 정렬 방식을 지정합니다.

`align-self` 속성은 플렉서블 컨테이너의 `align-items` 속성에 의해 설정된 기본 정렬 방식보다도 더 우선시됩니다.



아래의 2개 예제에서는 200픽셀 높이의 플렉스 컨테이너를 사용하여 `align-self` 속성을 더 잘 나타냅니다.

Example

컨테이너의 중간에 있는 세 번째 플렉스 아이템을 정렬합니다.

```
  <style>
    .flex-container {
      display: flex;
      height: 200px;
      background-color: #f1f1f1;
    }

    .flex-container>div {
      background-color: dodgerblue;
      color: #fff;
      width: 100px;
      margin: 10px;
      text-align: center;
      line-height: 75px;
      font-size: 30px;
    }
  </style>
</head>

<body>

  <h1>The align-self Property</h1>

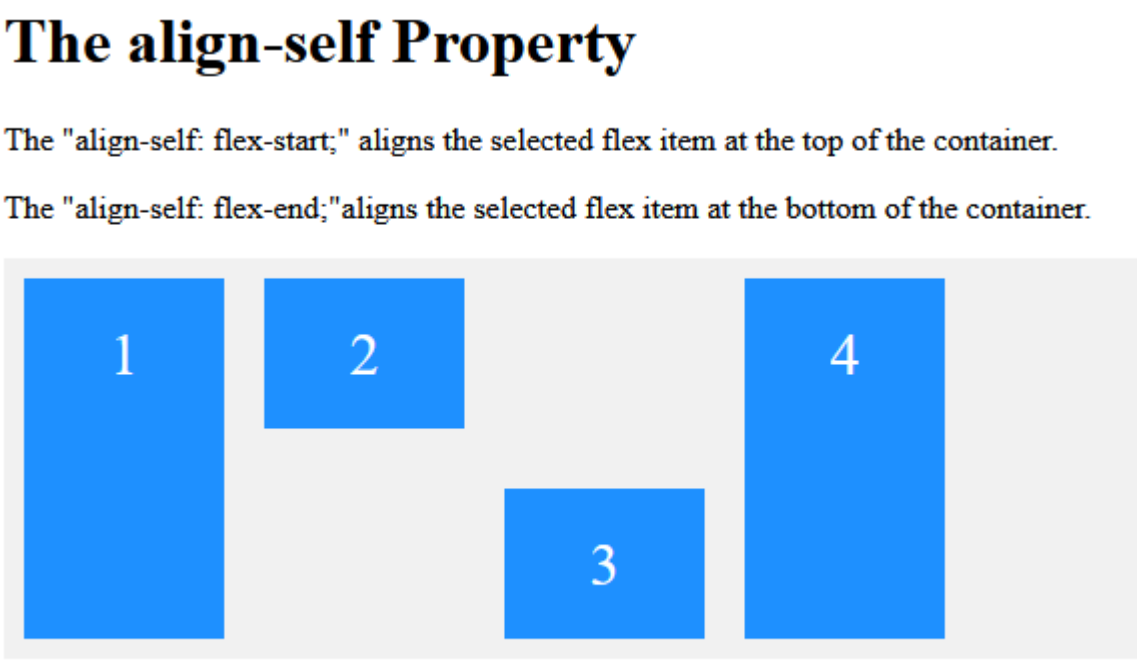
  <p>The "align-self: center;" aligns the selected flex item in the middle
of the container:</p>

  <div class="flex-container">
    <div>1</div>
    <div>2</div>
    <div style="align-self: center">3</div>
```

```
    <div>4</div>
  </div>
</body>
```

Example

컨테이너의 상단에 두 번째 플레스 아이템을 정렬하고, 컨테이너의 하단에 플렉스 아이템을 정렬합니다.




```
  <style>
    .flex-container {
      display: flex;
      height: 200px;
      background-color: #f1f1f1;
    }

    .flex-container>div {
      background-color: dodgerblue;
      color: #fff;
      width: 100px;
      margin: 10px;
      text-align: center;
      line-height: 75px;
      font-size: 30px;
    }
  </style>
</head>

<body>
```

```
  <h1>The align-self Property</h1>

  <p>The "align-self: flex-start;" aligns the selected flex item at the top
of the container.</p>
  <p>The "align-self: flex-end;"aligns the selected flex item at the bottom
of the container.</p>

  <div class="flex-container">
    <div>1</div>
    <div style="align-self: flex-start">2</div>
    <div style="align-self: flex-end">3</div>
    <div>4</div>
  </div>

  <p>The align-self property overrides the align-items property of the
container.</p>
</body>
```

The CSS Flexbox Items Properties

다음 표에는 모든 CSS Flexbox 항목 속성이 나열되어 있습니다.

Property	Description
align-self	플렉스 아이템의 정렬을 지정합니다(플렉스 컨테이너의 align-items 속성보다 우선시됩니다).
flex	flex-grow, flex-shrink 및 flex-basis 속성에 대한 약식 속성
flex-basis	플렉스 항목의 초기 길이를 지정합니다.
flex-grow	동일한 컨테이너 내부의 나머지 플렉스 아이템에 비례하여 플렉스 항목이 얼마나 확장될 것인지를 지정합니다.
flex-shrink	동일한 컨테이너 내부의 나머지 플렉스 아이템들에 비례하여 플렉스 아이템이 얼마나 축소될 것인지를 지정합니다.
order	동일한 컨테이너 내부에서 플렉스 아이템들의 순서를 지정합니다.

오션,, CSS, Flexbox, Flex, items

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:css:css_note:css_flex_items&rev=1626425426

Last update: **2022/03/10 19:52**