

CSS Grid Intro

- description : CSS Responsive Web Design- Images
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-07-02 Fri

The source of this article

[CSS Grid Intro](#)

CSS Grid Layout Module



Grid Layout

CSS Grid Layout Module은 행(rows)과 열(columns)이 있는 그리드 기반 레이아웃 시스템을 제공하여, floats와 positioning을 사용하지 않고도 웹 페이지를 더 쉽게 디자인할 수 있습니다.

Grid Elements

그리드 레이아웃은 하나 이상의 자식 요소가 있는 부모 요소로 구성됩니다.

Example

```
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
  background-color: #2196F3;
```

```
padding: 10px;
}
.grid-item {
background-color: rgba(255, 255, 255, 0.8);
border: 1px solid rgba(0, 0, 0, 0.8);
padding: 20px;
font-size: 30px;
text-align: center;
}
</style>
</head>
<body>

<h1>Grid Elements</h1>

<p>
  A Grid Layout must have a parent element with the display
property
  set to grid or inline-grid.</p>

<p>
  Direct child element(s) of the grid container automatically becomes
grid items.
</p>

<div class="grid-container">
  <div class="grid-item">1</div>
  <div class="grid-item">2</div>
  <div class="grid-item">3</div>
  <div class="grid-item">4</div>
  <div class="grid-item">5</div>
  <div class="grid-item">6</div>
  <div class="grid-item">7</div>
  <div class="grid-item">8</div>
  <div class="grid-item">9</div>
</div>
</body>
```



Display Property

HTML 요소는 `display` 속성이 `grid` 또는 `inline-grid`로 설정되면 그리드 컨테이너가 됩니다.

Example

```

<style>
  .grid-container {
    display: grid;
    grid-template-columns: auto auto auto; /* 3개의 컬럼(세로 열)을 만들고, 자동
으로 3개의 로우(가로 행)가 생성 */
    background-color: #2196F3;
    padding: 10px;
  }

  .grid-item {
    background-color: rgba(255, 255, 255, 0.8);
    border: 1px solid rgba(0, 0, 0, 0.8);
    padding: 20px;
    font-size: 30px;
    text-align: center;
  }
</style>
</head>
<body>

  <h1>The display Property:</h1>
  <div class="grid-container">
    <div class="grid-item">1</div>
    <div class="grid-item">2</div>
    <div class="grid-item">3</div>
    <div class="grid-item">4</div>
    <div class="grid-item">5</div>
    <div class="grid-item">6</div>
    <div class="grid-item">7</div>
    <div class="grid-item">8</div>
    <div class="grid-item">9</div>
  </div>

  <p>
    Set the <em>display</em> property to <em>grid</em> to make a block-
    level grid container.
  </p>
</body>

```



Example

```

<style>
  .grid-container {
    display: inline-grid;

```

```
grid-template-columns: auto auto auto; /* 3개의 컬럼(세로 열)을 만들고, 자동  
으로 3개의 로우(가로 행)가 생성 */  
background-color: #2196F3;  
padding: 10px;  
}  
  
.grid-item {  
background-color: rgba(255, 255, 255, 0.8);  
border: 1px solid rgba(0, 0, 0, 0.8);  
padding: 20px;  
font-size: 30px;  
text-align: center;  
}  
</style>  
</head>  
<body>  
  
<h1>The display Property:</h1>  
<div class="grid-container">  
  <!-- .grid-item$*9 엔터 클릭하면 아래와 같이 -->  
  <div class="grid-item">1</div>  
  <div class="grid-item">2</div>  
  <div class="grid-item">3</div>  
  <div class="grid-item">4</div>  
  <div class="grid-item">5</div>  
  <div class="grid-item">6</div>  
  <div class="grid-item">7</div>  
  <div class="grid-item">8</div>  
  <div class="grid-item">9</div>  
</div>  
  
<p>  
  Set the <em>display</em> property to <em>inline-grid</em>  
  to make a block-level grid container.  
</p>  
</body>
```



Grid Columns

그리드 아이템의 수직선(공간)은 컬럼(columns)라고 합니다.



ref : The source of this png file is [CSS Grid Intro](#)

Grid Rows

그리드 아이템의 수평선(공간) 로우(rows)라고 합니다.



ref : The source of this png file is [CSS Grid Intro](#)

Grid Gaps

각각의 컬럼/로우 사이의 공간을 갭(gaps)이라고 합니다.



아래의 속성들을 사용하여 갭 사이즈를 조정할 수 있습니다.

- grid-column-gap
- grid-row-gap
- grid-gap

Example

grid-column-gap 속성은 컬럼(세로 열) 사이의 간격을 설정합니다.

```
<style>
  .grid-container {
    display: grid;
    grid-column-gap: 50px;
    grid-template-columns: auto auto auto;
    background-color: #2196F3;
    padding: 10px;
  }

  .grid-item {
    background-color: rgba(255, 255, 255, 0.8);
    border: 1px solid rgba(0, 0, 0, 0.8);
    padding: 20px;
    font-size: 30px;
    text-align: center;
  }
</style>
</head>
<body>
```

```
<h1>The grid-column-gap Property:</h1>
```

```
<div class="grid-container">
  <div class="grid-item">1</div>
  <div class="grid-item">2</div>
  <div class="grid-item">3</div>
  <div class="grid-item">4</div>
  <div class="grid-item">5</div>
  <div class="grid-item">6</div>
  <div class="grid-item">7</div>
  <div class="grid-item">8</div>
  <div class="grid-item">9</div>
```

```
</div>
```

```
<p>Use the <em>grid-column-gap</em> property to adjust the space between
the columns.</p>
</body>
```



Example

grid-row-gap 속성은 로우(rows, 가로 행) 사이의 간격을 설정합니다.

```
<style>
  .grid-container {
    display: grid;
    grid-template-columns: auto auto auto;
    background-color: #2196f3;
    padding: 10px;
    grid-row-gap: 50px;
  }

  .grid-item {
    background-color: rgba(255, 255, 255, 0.8);
    border: 1px solid rgba(0, 0, 0, 0.8);
    padding: 20px;
    font-size: 30px;
    text-align: center;
  }
</style>
```

```
</head>
```

```
<body>
```

```
<h1>The grid-row-gap Property:</h1>
```

```
<div class="grid-container">
  <div class="grid-item">1</div>
  <div class="grid-item">2</div>
  <div class="grid-item">3</div>
  <div class="grid-item">4</div>
  <div class="grid-item">5</div>
  <div class="grid-item">6</div>
  <div class="grid-item">7</div>
  <div class="grid-item">8</div>
  <div class="grid-item">9</div>
</div>
```

<p>Use the grid-row-gap property to adjust the **space** between the rows.</p>

</body>



Example

grid-gap 속성은 grid-row-gap 및 grid-column-gap 속성에 대한 속기 속성(shorthand property)입니다.

```
<style>
  .grid-container {
    display: grid;
    grid-gap: 50px 100px; /* rows and columns in order */
    grid-template-columns: auto auto auto;
    background-color: #2196F3;
    padding: 10px;
  }
  .grid-item {
    background-color: rgba(255, 255, 255, 0.8);
    border: 1px solid rgba(0, 0, 0, 0.8);
    padding: 20px;
    font-size: 30px;
    text-align: center;
  }
</style>
</head>
<body>

  <h1>The grid-gap Property:</h1>

  <div class="grid-container">
    <div class="grid-item">1</div>
    <div class="grid-item">2</div>
    <div class="grid-item">3</div>
    <div class="grid-item">4</div>
```

```
<div class="grid-item">5</div>
<div class="grid-item">6</div>
<div class="grid-item">7</div>
<div class="grid-item">8</div>
<div class="grid-item">9</div>
</div>
```

<p>Use the grid-gap property to adjust the **space** between the columns andthe rows.</p>

```
</body>
```



Example

grid-gap 속성을 사용하여 행 간격과 열 간격을 하나의 값으로 설정할 수도 있습니다:

```
<style>
.grid-container {
  display: grid;
  grid-gap: 50px; /* one value for both rows and columns */
  grid-template-columns: auto auto auto;
  background-color: #2196F3;
  padding: 10px;
}
.grid-item {
  background-color: rgba(255, 255, 255, 0.8);
  border: 1px solid rgba(0, 0, 0, 0.8);
  padding: 20px;
  font-size: 30px;
  text-align: center;
}
</style>
</head>
<body>

<h1>The grid-gap Property:</h1>

<div class="grid-container">
  <div class="grid-item">1</div>
  <div class="grid-item">2</div>
  <div class="grid-item">3</div>
  <div class="grid-item">4</div>
  <div class="grid-item">5</div>
  <div class="grid-item">6</div>
  <div class="grid-item">7</div>
```



```
<div class="grid-item">8</div>
<div class="grid-item">9</div>
</div>
```

<p>Use the `grid-gap` property to adjust the **space** between the columns and the rows.</p>

```
</body>
```



Grid Lines

컬럼 사이의 라인들을 컬럼 라인(column lines)이라고 합니다.



그리드 컨테이너 내부에 그리드 아이템을 위치시킬 때 라인 넘버를 참조합니다.

Example

컬럼 라인 1에 그리드 아이템을 위치시키고, 컬럼 라인 3에서 종료하게 합니다.

```
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
  grid-gap: 10px;
  background-color: #2196F3;
  padding: 10px;
}

.grid-container > div {
  background-color: rgba(255, 255, 255, 0.8);
  text-align: center;
  padding: 20px 0;
  font-size: 30px;
}

.item1 {
  grid-column-start: 1;
  grid-column-end: 3;
}
</style>
</head>
<body>

<h1>Grid Lines</h1>
```

```
<div class="grid-container">
  <div class="item1">1</div>
  <div class="item2">2</div>
  <div class="item3">3</div>
  <div class="item4">4</div>
  <div class="item5">5</div>
  <div class="item6">6</div>
  <div class="item7">7</div>
  <div class="item8">8</div>
</div>

<p>You can refer to line numbers when placing grid items.</p>

</body>
```



Example

로우 라인 1에 그리드 아이템을 위치시키고, 로우 라인 3에서 종료하게 합니다.

```
<style>
  .grid-container {
    display: grid;
    grid-template-columns: auto auto auto;
    grid-gap: 10px;
    background-color: #2196F3;
    padding: 10px;
  }

  .grid-container > div {
    background-color: rgba(255, 255, 255, 0.8);
    text-align: center;
    padding: 20px 0;
    font-size: 30px;
  }

  .item1 {
    grid-row-start: 1;
    grid-row-end: 3;
  }
</style>
</head>
<body>

  <h1>Grid Lines</h1>

  <div class="grid-container">
```

```
<div class="item1">1</div>
<div class="item2">2</div>
<div class="item3">3</div>
<div class="item4">4</div>
<div class="item5">5</div>
<div class="item6">6</div>
<div class="item7">7</div>
<div class="item8">8</div>
</div>
```

<p>You can refer to line numbers when placing grid items.</p>
</body>



All CSS Grid Properties

[오션](#), [CSS](#), [Grid](#), [Layout](#), [Module](#)

From:

<http://rwiki.repia.com/> - 문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12

Permanent link:

http://rwiki.repia.com/doku.php?id=wiki:css:css_note:grid_intro&rev=1625208934

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

