

display: grid

Thuộc tính này sẽ định nghĩa một phần tử là một lưới. Có 2 giá trị chính: **grid** và **inline-grid**.

- **grid**: Phần tử sẽ hiển thị dưới dạng lưới.
- **inline-grid**: Phần tử sẽ hiển thị dưới dạng lưới nhưng nó sẽ không chiếm toàn bộ chiều rộng của trang.

Ngay sau khi được thiết lập, phần tử có thuộc tính này sẽ trở thành dạng "lưới"

```
.container {  
  display: grid;  
}
```

grid-template-columns và grid-template-rows (viết trong grid container)

Đây là thuộc tính quan trọng nhất trong CSS Grid, nó sẽ xác định số lượng và kích thước của các cột và hàng trong lưới.

Nó sẽ nhận giá trị là một chuỗi các giá trị, mỗi giá trị sẽ tương ứng với một cột hoặc hàng.

```
<style>  
  .container {  
    display: grid;  
    grid-template-columns: 100px 100px 100px; /* Ba cột mỗi cột 100px */  
    grid-template-rows: 150px 150px; /* Hai hàng mỗi hàng 150px */  
    gap: 10px; /* Khoảng cách giữa các cột và hàng */  
    background-color: #f0f0f0;  
  }  
  
  .item {  
    background-color: #4caf50;  
    color: white;  
    display: flex;  
    justify-content: center;  
    align-items: center;  
  }  
</style>  
  
<div class="container">  
  <div class="item">1</div>  
  <div class="item">2</div>  
  <div class="item">3</div>  
  <div class="item">4</div>  
  <div class="item">5</div>
```

```
<div class="item">6</div>
</div>
```

`grid-template-columns` hoặc `grid-template-rows` nhận giá trị là các đơn vị đo, ví dụ như `px`, `em`, `%`, v.v. Và được viết tách biệt bởi dấu cách, tương ứng với số lượng hàng hoặc cột được quy định.

Đơn vị fr

Đây là một đơn vị đặc biệt trong CSS Grid, nó sẽ **chia đều không gian còn lại** của lưới cho các cột hoặc hàng.

Ví dụ:

```
.container {
  display: grid;
  grid-template-columns: 150px 2fr 1fr; /* Cột đầu tiên cố định 150px,
  cột thứ hai chiếm 2 phần, cột thứ ba chiếm 1 phần */
}
```

Hàm `minmax()` và `repeat()`

Trong CSS Grid, chúng ta có thể sử dụng hàm `minmax()` để xác định một phần tử có thể co giãn từ một giá trị tối thiểu đến tối đa.

Ví dụ:

```
.container {
  display: grid;
  grid-template-columns: repeat(3, minmax(100px, 1fr));
}
```

Trong ví dụ trên, chúng ta sẽ tạo ra một lưới với 3 cột, mỗi cột sẽ co giãn từ 100px đến 1 phần của lưới (nếu có 3 phần thì mỗi hàng sẽ được chia thành 3 cột đều nhau vì mỗi cột đều chiếm 1fr). Chúng ta sử dụng `repeat()` để lặp lại 3 lần giá trị `minmax(100px, 1fr)` -> điều này hạn chế việc lặp lại code, code thừa.

`auto-fill` và `auto-fit`

- `auto-fill` tự động tạo thêm tối đa các cột trên cùng hàng, ngay cả khi nội dung trống. Độ rộng tối thiểu của cột phụ thuộc vào giá trị min trong `minmax()`.
- `auto-fit` cũng tạo thêm các cột trên cùng hàng như `auto-fill`, nhưng nó sẽ co giãn các cột để lấp đầy không gian trống.

Ví dụ:

```
<style>
  .grid-container {
    display: grid;
  }

  .grid-container--fill {
    grid-template-columns: repeat(auto-fill, minmax(100px, 1fr));
  }

  .grid-container--fit {
    grid-template-columns: repeat(auto-fit, minmax(100px, 1fr));
  }

  .grid-element {
    background-color: deepPink;
    padding: 20px;
    color: #fff;
    border: 1px solid #fff;
  }

  body {
    padding: 2em;
  }

  hr {
    margin: 80px;
  }
</style>
```

```
<h2>Auto-fill</h2>
<div class="grid-container grid-container--fill">
  <div class="grid-element">1</div>
  <div class="grid-element">2</div>
  <div class="grid-element">3</div>
  <div class="grid-element">4</div>
  <div class="grid-element">5</div>
  <div class="grid-element">6</div>
  <div class="grid-element">7</div>
</div>
```

```
<hr />
```

```
<h2>Auto-fit</h2>
<div class="grid-container grid-container--fit">
  <div class="grid-element">1</div>
  <div class="grid-element">2</div>
  <div class="grid-element">3</div>
  <div class="grid-element">4</div>
  <div class="grid-element">5</div>
  <div class="grid-element">6</div>
  <div class="grid-element">7</div>
</div>
```

max-content() và min-content()

max-content và **min-content** là hai giá trị đặc biệt trong CSS Grid, chúng sẽ giúp chúng ta xác định kích thước của một phần tử dựa trên nội dung bên trong.

- **max-content** khiến một ô lưới mở rộng tối đa để nội dung không bị xuống dòng.
- **min-content** sẽ giữ kích thước nhỏ nhất có thể mà nội dung không bị tràn ra ngoài.

Ví dụ:

```
<style>
  .container {
    display: grid;
    grid-template-columns: min-content max-content 1fr;
    gap: 10px;
  }

  .item {
    padding: 10px;
    background-color: #4caf50;
    color: white;
  }
</style>

<div class="container">
  <div class="item">
    Lorem ipsum dolor sit amet consectetur adipisicing elit. Totam
    obcaecati fuga molestiae asperiores sunt expedita
    labore vitae exercitationem accusamus atque.
  </div>
  <div class="item">
    Lorem ipsum dolor sit amet consectetur adipisicing elit. Totam
    obcaecati fuga molestiae asperiores sunt expedita
    labore vitae exercitationem accusamus atque.
  </div>
  <div class="item">This is a shorter text.</div>
</div>
```