

## justify-content, align-content, justify-items, align-items và place-items (viết trong grid container)

Đầu tiên, trước khi bước vào các khái niệm này, hãy cùng ôn lại về **justify** và **align** trong **flexbox**. Ở **flexbox**, **justify** điều chỉnh các phần tử theo trục chính (main axis), trong khi **align** điều chỉnh các **content** theo trục phụ (cross axis).

Ở Grid CSS, khái niệm này vẫn duy trì quy ước như vậy, thuộc tính **justify** vẫn làm việc với trục chính. Tuy nhiên chúng ta hiểu trục chính (main axis) luôn luôn là trục nằm ngang, tức là quản lý và phân bổ các cột. Ngược lại **align** làm việc với trục dọc (phân bổ các hàng).

### Các giá trị phổ biến:

- **start**: Căn về phía trái (nếu là hàng ngang), căn về phía trên (nếu là hàng dọc).
- **end**: Căn về phía phải (nếu là hàng ngang), căn về phía dưới (nếu là hàng dọc).
- **center**: căn tập trung vào giữa.
- **space-between**: Các track phân bổ đều với khoảng trống thừa giữa chúng.
- **space-around**: Các track được phân bổ đều với cả khoảng cách trước và sau mỗi track (giống margin cho từng phần tử cách đều các bên),
- **space-evenly**: Các track phân bổ đều giữa chúng và cũng cách đều 2 bên.
- **stretch**: Kéo dài phần tử để lấp đầy toàn bộ một chiều nào đó của ô lưới.
- **baseline**: Align-items có thể nhận giá trị **baseline**, giá trị này ghi tràn nội dung và lấy kích thước phần tử theo dòng cuối cùng của content (ít khi được sử dụng).

### Bây giờ hãy đi vào ví dụ trước:

```
<style>
.grid-container {
  display: grid;
  grid-template-columns: repeat(3, 100px);
  grid-template-rows: repeat(2, 100px);
  grid-gap: 10px;
  background-color: #f0f0f0;
}

.grid-item {
  background-color: #4caf50;
  color: white;
  text-align: center;
  padding: 20px;
  font-size: 18px;
}

.justify-content {
  justify-content: space-around;
  /* Căn theo chiều ngang */
}

.align-content {
  height: 300px;
}
```

```
    align-content: space-around;
    /* Căn theo chiều dọc */
}

.justify-items {
    justify-items: center;
    /* Tại không gian mà phần tử chiếm chỗ, phần tử co lại sao cho vừa đủ
    nội dung hiển thị theo chiều ngang, tức là vừa đủ chiều rộng cho content
    và padding. */
}

.align-items {
    align-items: stretch;
    /* Tại không gian mà phần tử chiếm chỗ, phần tử duỗi ra sao cho chiếm
    hết phần không gian hiển thị theo chiều dọc, tức là bằng 100px theo quy
    ước grid-template-rows */
}
</style>

<h2>justify-content: space-between</h2>
<div class="grid-container justify-content">
    <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>
<h2>align-content: center</h2>
<div class="grid-container align-content">
    <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>

<h2>justify-items: center</h2>
<div class="grid-container justify-items">
    <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>
<h2>align-items: stretch</h2>
<div class="grid-container align-items">
    <div class="grid-item">1</div>
    <div class="grid-item">2</div>
```

```
<div class="grid-item">lorem</div>
<div class="grid-item">4</div>
<div class="grid-item">5</div>
<div class="grid-item">6</div>
</div>
```

### Giải thích:

- **`**justify-content**`** kiểm soát cách các track (các hàng hoặc cột) của grid được căn chỉnh theo trục chính (ngang) trong container grid. Nó ảnh hưởng đến cách các track được phân bố nếu chúng không chiếm toàn bộ chiều rộng của container.
- **`**align-content**`** kiểm soát cách các track được căn chỉnh theo trục phụ (dọc) trong container grid. Nó hoạt động tương tự như justify-content, nhưng trên trục dọc.
- **`justify-items`** kiểm soát cách các phần tử con của grid (grid items) được căn chỉnh theo trục ngang bên trong ô của chúng. Điều này ảnh hưởng đến từng phần tử con riêng lẻ, không phải toàn bộ lưới khi phần tử con không chiếm toàn bộ không gian của một ô lưới (cell) thì thuộc tính này phát huy tác dụng.
- **`align-items`** kiểm soát cách các phần tử con của grid được căn chỉnh theo trục dọc bên trong ô của chúng.
- **`place-items`** là cú pháp rút gọn kết hợp **`align-items`** và **`justify-items`**.