

# HTML5 & CSS3

CYBERLEARN.VN

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

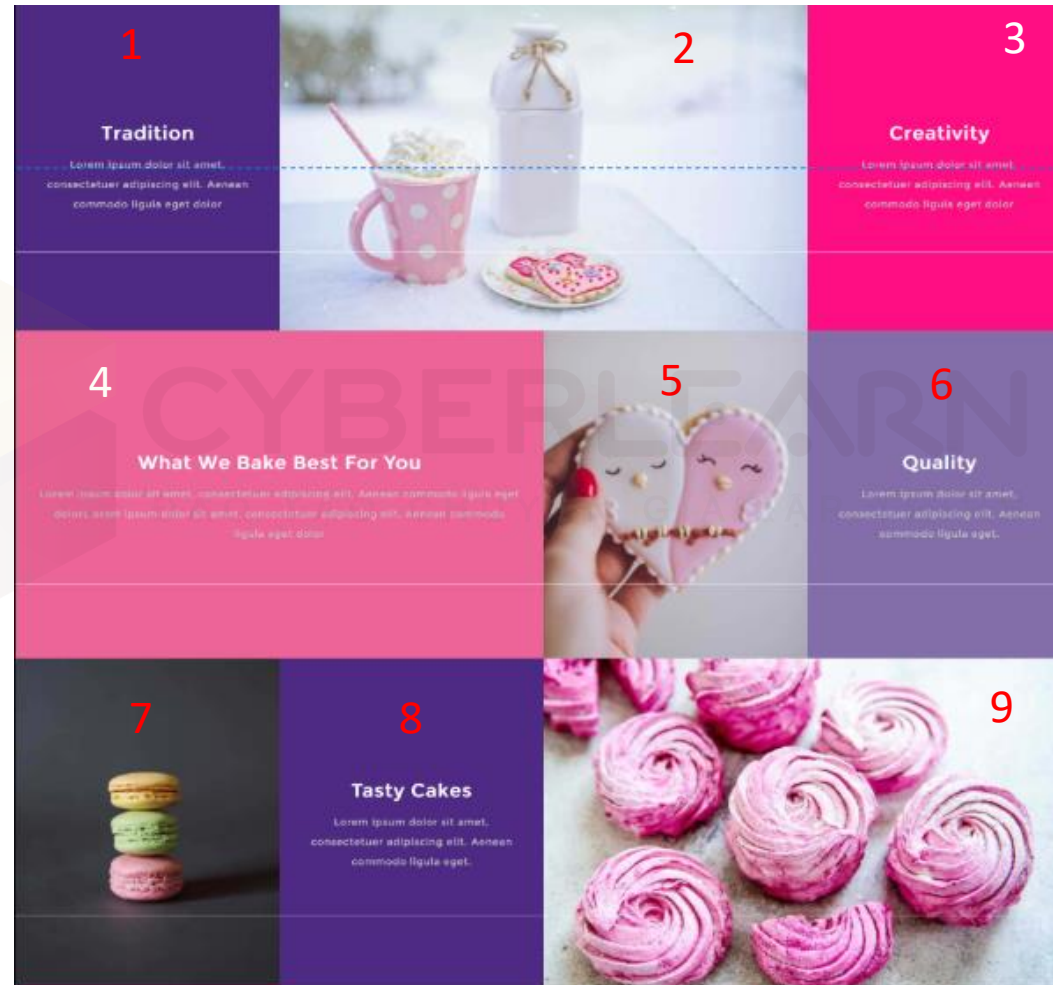


CYBERLEARN

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# Xử lý phần Intro

# Xử lý phần Intro



# Xử lý phần Intro

## ➤ Xây dựng HTML cho Intro:

Phần này ta chỉ cần liệt kê ra các phần tử html theo thứ tự, không cần chia bố cục cụ thể, vì ta sẽ dùng css grid layout để dàn phần này.

```
<!--INTRO-->
<section class="intro">
  <div class="intro_item intro_tradition">
    <div>
      <h2>Tradition</h2>
      <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Minima dolor harum expedita aspernatur, facilis quo.
    </p>
    </div>
  </div>
  <div class="intro_item intro_img-1">
    
  </div>
  <div class="intro_item intro_creativity">
    <div>
      <h2>Creativity</h2>
      <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Minima dolor harum expedita aspernatur, facilis quo.
    </p>
    </div>
  </div>
  <div class="intro_item intro_things">
    <div>
      <h2>What We Bake Best For You</h2>
      <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Minima dolor harum expedita aspernatur, facilis quo.
    </p>
    </div>
  </div>
</section>
```

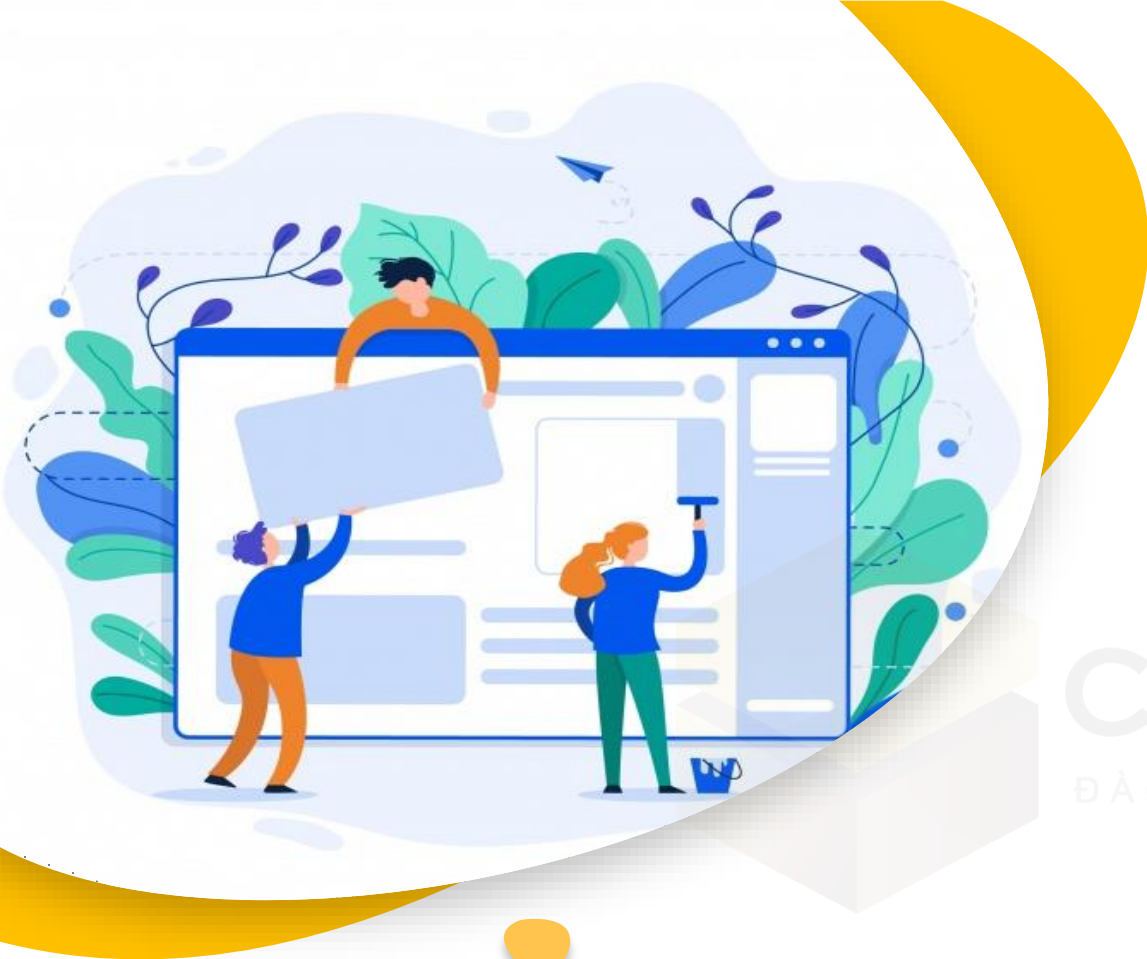
```
<div class="intro_item intro_img-2">
  
</div>
<div class="intro_item intro_quality">
  <div>
    <h2>Quality</h2>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Minima dolor harum expedita aspernatur, facilis quo.
  </p>
  </div>
</div>
<div class="intro_item intro_img-3">
  
</div>
<div class="intro_item intro_cakes">
  <div>
    <h2>Tasty Cakes</h2>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Minima dolor harum expedita aspernatur, facilis quo.
  </p>
  </div>
</div>
<div class="intro_item intro_img-4">
  
</div>
</section>
<!-- END INTRO -->
```



CYBERLEARN

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Grid

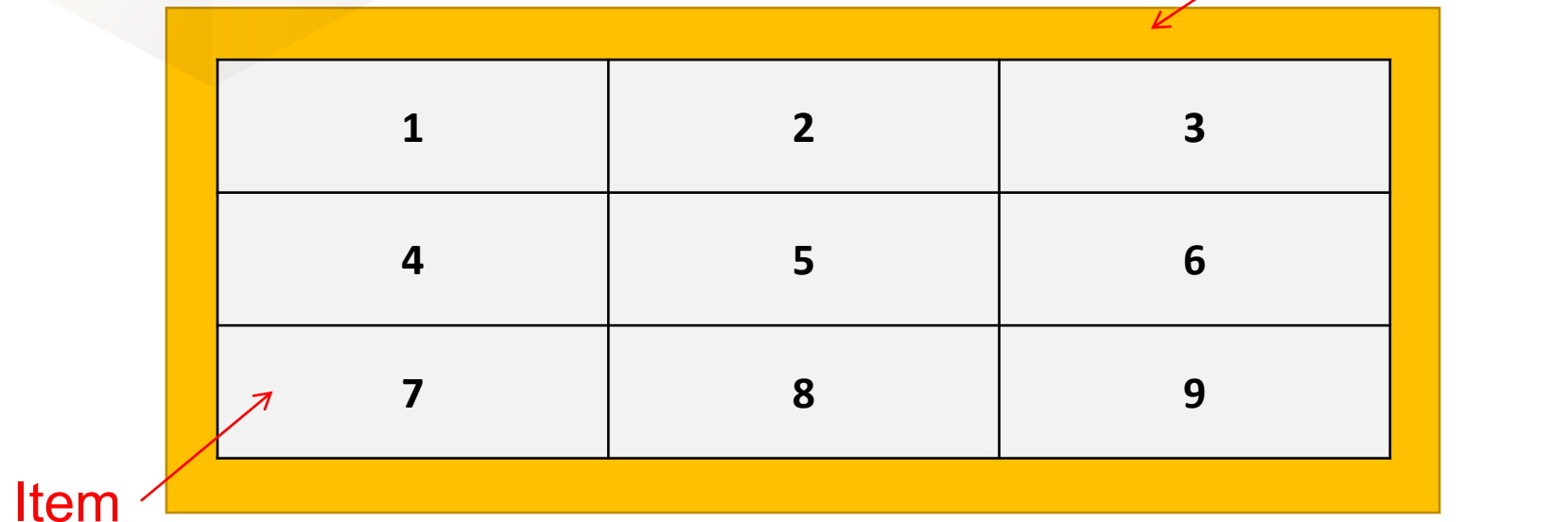


## Grid là gì?

- Cung cấp một hệ thống lưới cơ bản
- Bao gồm các cột và các hàng
- Dàn layout dễ hơn mà không phải sử dụng float hay position.

## Các thành phần trong Grid

- Gồm 2 thành phần chính :
  - Grid Container
  - Grid Item
- Muốn sử dụng css Grid phải có các thuộc tính :
  - Display: grid
  - Grid-template-columns: chia cột cho phần tử



# Các thuộc tính của Grid

## ➤ Grid-template-columns

- Dùng để chia cột cho phần tử
- Cú pháp :
  - `Grid-template-columns:[col1] [col2] [col3]...;`
  - `Grid-template-columns: repeat(số cột, kích thước);`
- Đơn vị kích thước ưu tiên dùng “fr”
- Ví dụ: `grid-template-columns: 1fr 1fr 2fr`  
=> chia phần tử ra thành 4 phần bằng nhau, cột 1 và 2 chiếm một phần, cột 3 chiếm 2 phần

Column	Column	Column



# Các thuộc tính của Grid

## ➤ Grid-template-columns

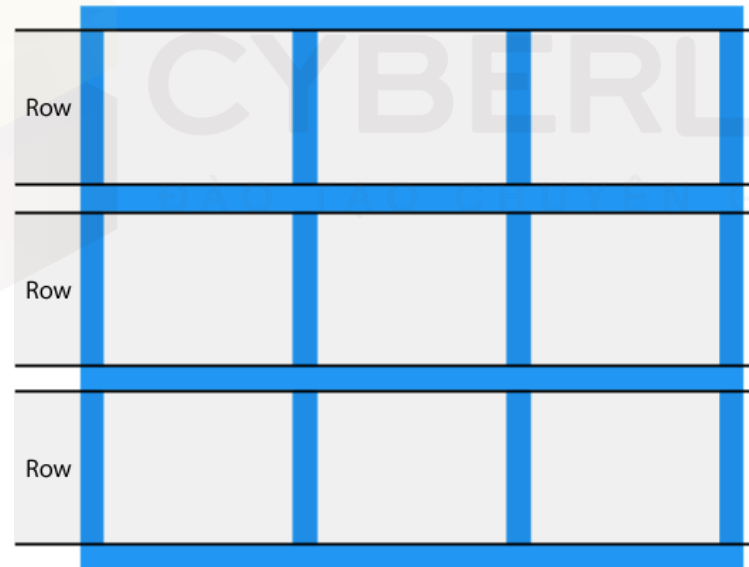
```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  grid-gap: 10px;  
  background-color: gold;  
  padding: 10px;  
}  
  
.grid-container > div {  
  background-color: rgba(255, 255, 255, 0.8);  
  text-align: center;  
  padding: 20px 0;  
  font-size: 30px;  
}
```

1	2	3
4	5	6

# Các thuộc tính của Grid

## ➤ Grid-template-rows

- Cú pháp :
  - Grid-template-rows:[row1] [row2] [row3]...;
  - Chỉ nên xét khi muốn chỉnh height cho các rows.



# Các thuộc tính của Grid

## ➤ Grid-template-rows

```
.grid-container-{
  display: grid;
  grid-template-columns: auto auto auto;
  grid-template-rows: 100px 300px;
  grid-gap: 10px;
  background-color: gold;
  padding: 10px;
}

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

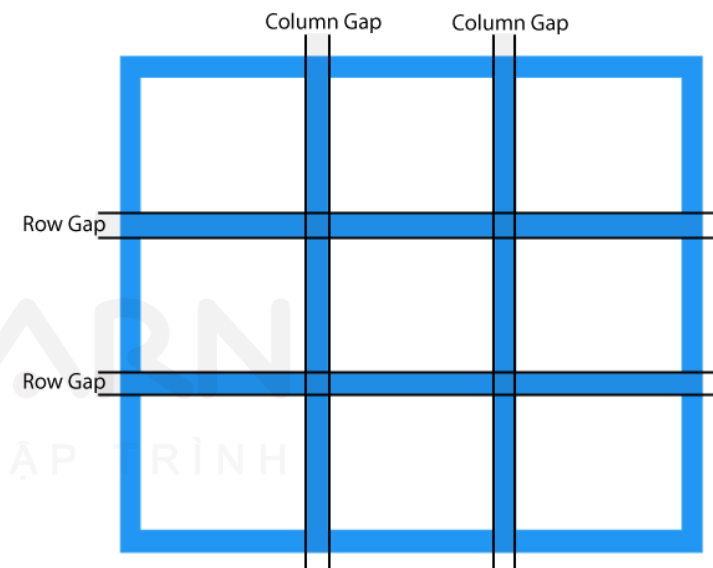
1	2	3
4	5	6

# Các thuộc tính của Grid

## ➤ Grid Gaps

- Là khoảng cách giữa các cột và các hàng.
- Cú pháp : **Grid-column-gap** | **grid-row-gap** | **grid-gap** : [khoảng cách];

```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  grid-gap: 30px;  
  background-color: gold;  
  padding: 10px;  
}  
  
.grid-container > div {  
  background-color: rgba(255, 255, 255, 0.8);  
  text-align: center;  
  padding: 20px 0;  
  font-size: 30px;  
}
```

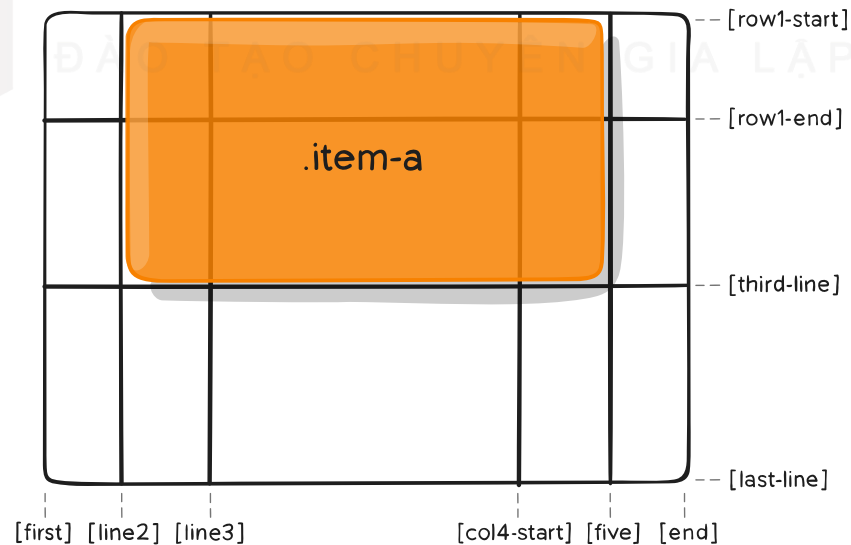


1	2	3
4	5	6
7	8	9

# Các thuộc tính của Grid

## ➤ Grid-column & Grid-row:

- Dùng để gom cột hoặc hàng tương tự như table
- Thêm ở item con cần gom
- Cần xác định **line** bắt đầu và kết thúc của item đó.
- Ví dụ: Item-a chiếm từ cột 2 đến cột 4 và hàng 1 đến hàng 2 nên sẽ phủ từ [line2] đến [five] và từ [row1-start] đến [third-line]



# Các thuộc tính của Grid

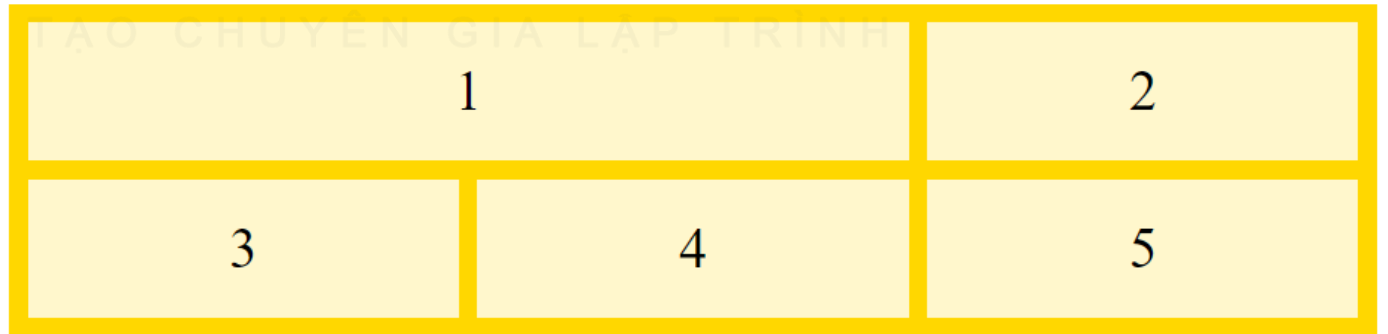
## ➤ Grid-column:

- Cú pháp: **grid-column: n / m;**
- Trong đó: n là vị trí line bắt đầu của cột đầu tiên mà item con này chiếm  
m là vị trí line kết thúc của cột cuối cùng mà item con này chiếm

```
.grid-container-{
  display: grid;
  grid-template-columns: auto auto auto;
  grid-gap: 10px;
  background-color: gold;
  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: 1 / 3;
}
```

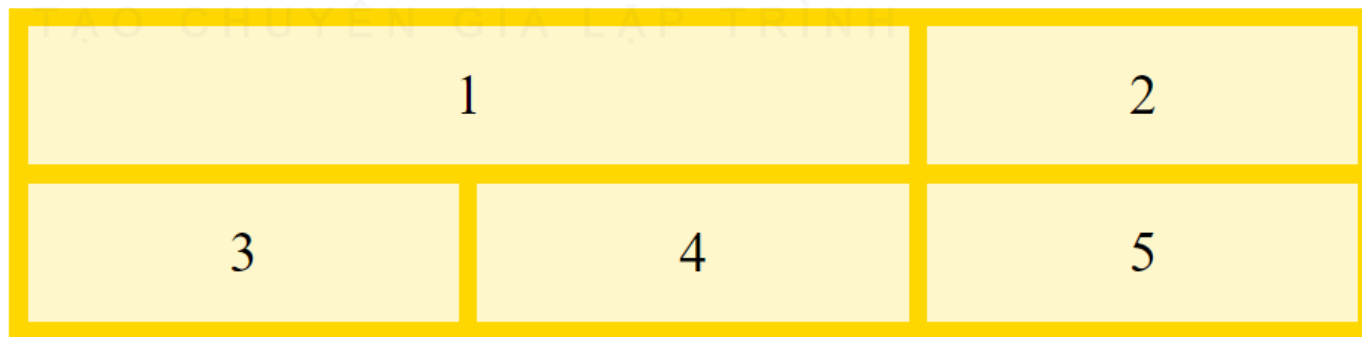


# Các thuộc tính của Grid

## ➤ Grid-column:

- Cú pháp: **grid-column: n / span m**
- Trong đó: n là vị trí line bắt đầu của cột đầu tiên mà item con này chiếm  
m là số cột ta muốn phần tử này chiếm

```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  grid-gap: 10px;  
  background-color: gold;  
  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: 1 / span 2;  
}
```



# Các thuộc tính của Grid

## ➤ Grid-row:

- Cú pháp gần giống với grid-column
  - **grid-row: n / m**
  - **grid-row: n / span row**
- Trong đó:
  - n là vị trí line bắt đầu của hàng đầu tiên mà item con này chiếm
  - m là vị trí line kết thúc của hàng cuối cùng mà item con này chiếm
  - row : số hàng mà item con này chiếm

1	2	3
	4	5

```
.grid-container-{
  display: grid;
  grid-template-columns: auto auto auto;
  grid-gap: 10px;
  background-color: gold;
  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: 1 / span 2;
}
```

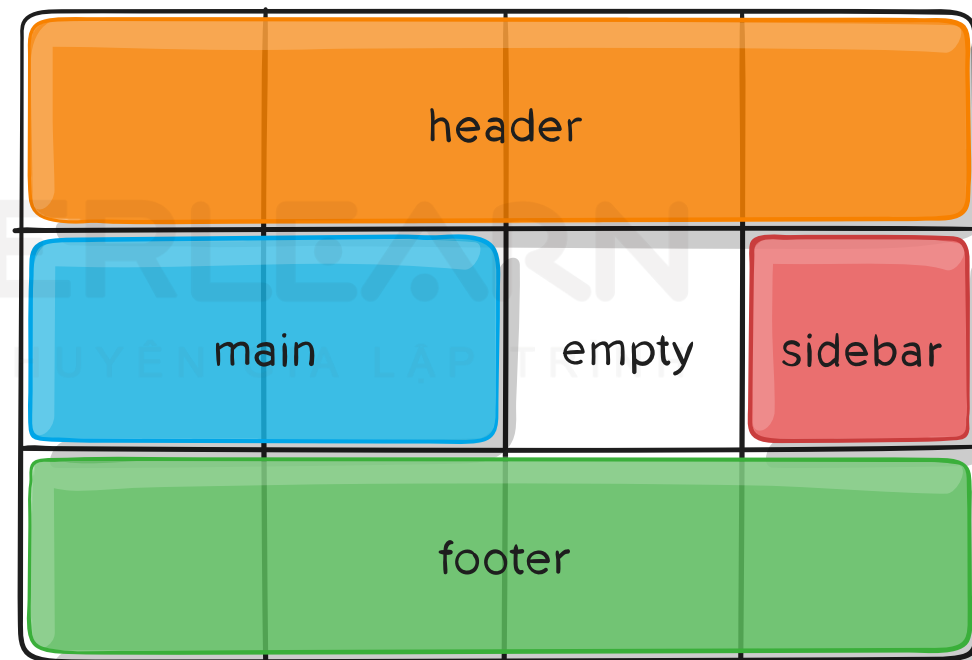


# Các thuộc tính của Grid

## ➤ Naming Grid Item:

Ta có thể đặt tên cho các grid items để sắp xếp vị trí trong trang web.

```
.item-a {  
  grid-area: header;  
}  
.item-b {  
  grid-area: main;  
}  
.item-c {  
  grid-area: sidebar;  
}  
.item-d {  
  grid-area: footer;  
}  
  
.container {  
  display: grid;  
  grid-template-columns: 50px 50px 50px 50px;  
  grid-template-rows: auto;  
  grid-template-areas:   
    "header header header header"  
    "main main . sidebar"  
    "footer footer footer footer";  
}
```



1

1 column

2

3

4

5

### Tradition

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. Aenean  
commodo ligula eget dolor



### Creativity

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. Aenean  
commodo ligula eget dolor

### What We Bake Best For You

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget  
dolor, aenean ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo  
ligula eget dolor



### Quality

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. Aenean  
commodo ligula eget.



### Tasty Cakes

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. Aenean  
commodo ligula eget.



# Xử lý phần Intro

```
.intro{
  display:grid;
  grid-template-columns: 1fr 1fr 1fr 1fr;
}
.intro__item{
  display: flex;
  align-items: center;
  text-align: center;
  color: white;
}
.intro__item img{
  max-width: 100%;
  display: block;
}
.intro__item h2{
  font-size:1.5rem;
  margin-bottom:1.5rem;
}
.intro__item p{
  font-size:0.7rem;
  padding:0 2rem;
  line-height: 1.5rem;
}
.intro__tradition{
  background-color: #4E2A84;
}
.intro__creativity{
  background-color: #FF0E83;
}
.intro__quality{
  background-color: #836EAA;
}
```

- (1) • Display: grid -> khai báo sử dụng css grid layout bên trong intro  
• Grid-template-column: chia cột cho phần intro  
• Ở đây ta chia phần intro thành 4 cột bằng nhau( mỗi cột 1fr), mỗi div con bên trong sẽ chiếm 1 cột, đồng nghĩa 1 hàng bây giờ chứa được 4 div, nếu nhiều hơn sẽ xuống hàng
- (2) Sử dụng flexbox, để xét div con bên trong .intro\_\_item nằm ra giữa theo chiều dọc

```
.intro__cakes{
  background-color: #4E2A84;
}
.intro__things{
  grid-column: 1/span 2;
  background-color: #ED6498;
}
.intro__img-1{
  grid-column: 2/span 2;
}
.intro__img-3{
  background-color: #4E2A84;
}
.intro__img-4{
  grid-column: 3/span 2;
}
```

- (3) Đối với grid, ta có thể gộp cột, gộp hàng tương tự như table  
Ở đây, ta đang gộp cột cho div có class.intro\_\_things, trong đó ta truyền vào 2 tham số : n/ span m. Trong đó: n là vị trí cột mà div này đang ở, m là số cột mà ta muốn nó chiếm

## So sánh Float, Flexbox, Grid

	Float	Flexbox	Grid
<b>Phiên bản</b>	CSS	CSS3	CSS3
<b>Khai báo</b>	<ul style="list-style-type: none"><li>• float: left hoặc float:right</li><li>• Thêm ở thẻ con</li><li>• Xét width cho mỗi phần tử con</li></ul>	<ul style="list-style-type: none"><li>• Display: flex</li><li>• Thêm ở thẻ cha</li></ul>	<ul style="list-style-type: none"><li>• Display:grid</li><li>• Thêm ở thẻ cha</li><li>• Số lượng column</li><li>• Số lượng row (nếu element có gồm hàng)</li></ul>
<b>Hỗ Trợ</b>	<ul style="list-style-type: none"><li>• Hỗ trợ cho tất cả các browsers</li><li>• Chỉ nên dùng khi flex và grid không dùng được</li></ul>	<ul style="list-style-type: none"><li>• Áp dụng cho vùng nội dung đơn giản, cần dàn các thẻ con nằm trên 1 hàng</li><li>• Hỗ trợ cho các browsers phiên bản mới</li></ul>	<ul style="list-style-type: none"><li>• Áp dụng cho vùng nội dung có thiết kế gần giống table, có nhiều hàng và cột</li><li>• Hỗ trợ cho các browsers phiên bản mới</li></ul>

# So sánh Float, Flexbox, Grid

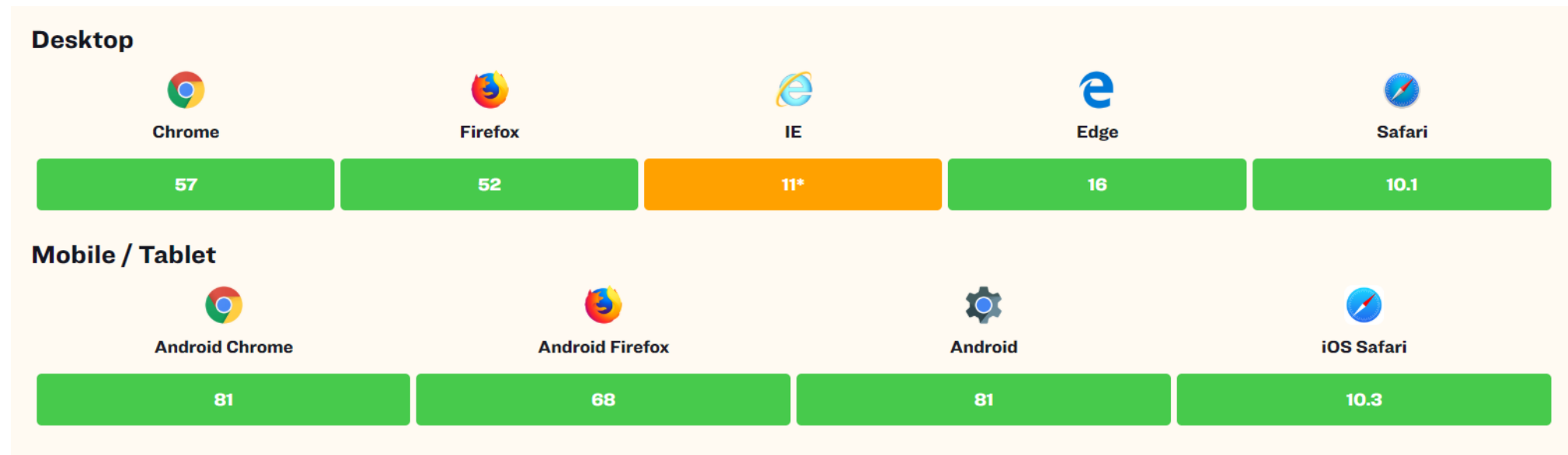
- Browser Support cho flex:
- (new): khai báo display: flex;
  - (tweener): khai báo display: flexbox;
  - (old): khai báo display: box;

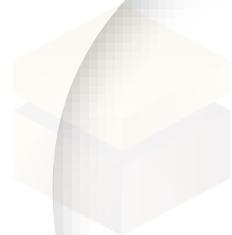
Chrome	Safari	Firefox	Opera	IE	Edge	Android	iOS
20- (old) 21+ (new)	3.1+ (old) 6.1+ (new)	2-21 (old) 22+ (new)	12.1+ (new)	10 (tweener) 11+ (new)	17+ (new)	2.1+ (old) 4.4+ (new)	3.2+ (old) 7.1+ (new)

# So sánh Float, Flexbox, Grid

## ➤ Browser Support cho grid:

Nếu như trình duyệt không hỗ trợ grid thì dùng flexbox thay thế, trường hợp cả flex và grid đều không được thì dùng float





CYBERLEARN  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# Thank You