



# HTML5 & CSS3

CYBERLEARN.VN



03

## CSS3 là gì?

CYBERLEARN  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH



## CSS3 khác gì với CSS

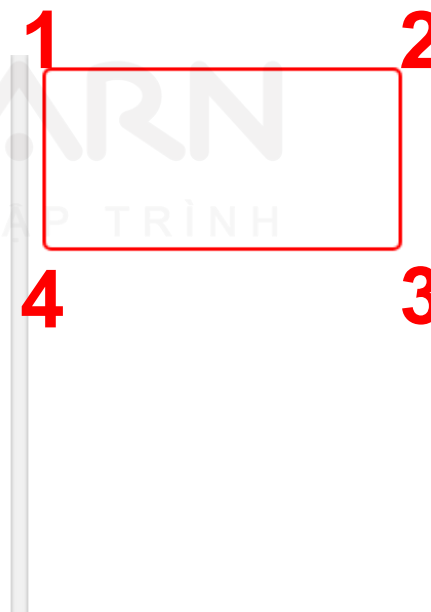
- CSS3 là một chuẩn (phiên bản) mới nhất của CSS.
- CSS3 tạo ra flexbox thay thế cho float.
- CSS3 tạo ra hệ thống lưới (grid) phục vụ dàn layout
- CSS3 cung cấp các thuộc tính mới làm những điều CSS không thể làm.
- CSS3 có thể tạo được các animation.

# Các thuộc tính của CSS3

## ➤ Border-radius:

- Thuộc tính border-radius : bo tròn góc các phần tử
- Có thể theo px , rem, em hoặc %;
- Cú pháp và kết quả :

```
<!DOCTYPE html>
<html>
<head>
<style>
div{
width:200px;
height:100px;
border: 2px solid red;
border-radius:10px;
}
</style>
</head>
<body>
<div></div>
</body>
</html>
```



- Các cách viết khác của border-radius
  - 2 tham số : **border-radius : [gốc 1,3] [gốc 2,4]**
  - 3 tham số : **border-radius : [gốc 1] [gốc 2,4] [gốc 3]**
  - 4 tham số : **border-radius : [gốc 1] [gốc 2] [gốc 3] [gốc 4]**

# Các thuộc tính của CSS3

- background-size :
  - Theo px ,rem hoặc em
  - Theo %
  - Contain, cover ,auto

```
<!DOCTYPE html>
<html>
<head>
<style>
#example1 {
    border: 1px solid black;
    background:url(img_flwr.gif);
    background-size: 100px 80px;
    background-repeat: no-repeat;
    padding:15px;
}

#example2 {
    border: 1px solid black;
    background:url(img_flwr.gif);
    background-repeat: no-repeat;
    padding:15px;
}
</style>
</head>
<body>
```

Điều chỉnh kích thước background:

## Lorem Ipsum Dolor

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Mặc định

## Lorem Ipsum Dolor

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

# Các thuộc tính của CSS3

## ➤ background-size :

- Phân biệt cover và contain
  - Contain: Hình nằm trong content , tùy theo kích thước khung chứa mà nó sẽ full theo chiều rộng hoặc chiều cao , không bao phủ hết.
  - Cover : Hình bao phủ hết nền, tuy nhiên hình ảnh có thể bị scale để bao phủ hoặc nếu quá lớn thì sẽ mất đi một số phần.
  - Chúng ta có thể kết hợp với background- position để lấy vị trí hình

**background-size: contain:**



**background-size: cover:**



# Các thuộc tính của CSS3

## ➤ @font-face

- Cài đặt font chữ vào file từ CSS

```
<!DOCTYPE html>
<html>
<head>
<style>
@font-face {
  font-family: myFirstFont;
  src: url(sansation_light.woff);
}
* {
  font-family: myFirstFont;
}
</style>
```

Change Orientation

CYBERLEARN

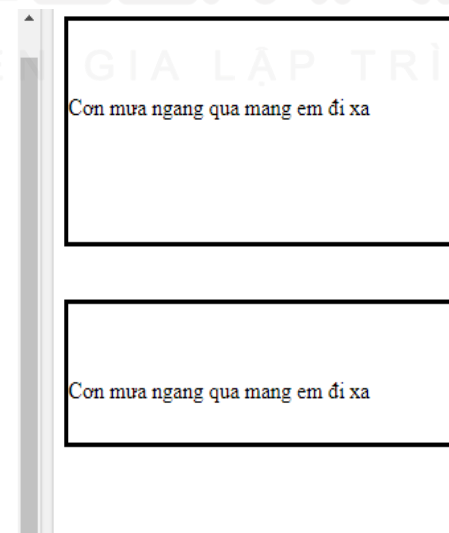
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# Các thuộc tính của CSS3

## ➤ Resizing (box-sizing)

- Cho phép thêm kích thước của border và padding vào width và height của 1 element (phần tử) hay không.
- Cách tính khi không có box-sizing :
  - width+padding+border = chiều rộng của phần tử
  - height+padding+border = chiều cao của phần tử
- Cách tính khi có box-sizing:  
Vẫn lấy width height xét kích thước, nhưng nếu có padding thì không cộng thêm padding vào.

```
<html>
<head>
<style>
#example1{
width:300px;
height:100px;
border:3px solid black;
padding-top:50px;
}
#example2{
width:300px;
height:100px;
border:3px solid black;
padding-top:50px;
box-sizing:border-box;
}
</style>
</head>
<body>
```







CYBERLEARN

ĐÀO TẠO CHUYÊN SẴN

# CSS3 Transition

# CSS3 Transition

## ➤ Transition

- Giúp chúng ta nhìn thấy được quá trình chuyển đổi của css
- Để thực hiện transition cần có:
  - Thuộc tính CSS chúng ta cần thực hiện transition
  - Duration của hiệu ứng.
- Ngoài ra chúng ta có các thuộc tính khác (có thể có hoặc không)
  - Timing-function
  - delay

# CSS3 Transition

## ➤ Transition property

- Chỉ định thuộc tính css chúng ta muốn thực hiện transition
- Cú pháp:
  - Transition-property: [Tên thuộc tính] / all ;
- Ví dụ :
  - Transition-property: width;
  - Transition-property: all;

CYBERLEARN

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# CSS3 Transition

## ➤ Transition duration

- Thời gian thực hiện transition.
- Cú pháp:
  - Transition-duration: [time];
- Ví dụ : Transition-duration : 2s;

CYBERLEARN  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# CSS3 Transition

## ➤ Transition timing function

- Thuộc tính transition-timing-function xác định tốc độ của hiệu ứng
- Bao gồm các giá trị:
  - Ease – Hiệu ứng ban đầu chậm, sau đó nhanh và chậm dần
  - Linear – Tốc độ giữ nguyên từ đầu tới cuối
  - Ease-in – Bắt đầu chậm, sau đó nhanh
  - Ease-out – Bắt đầu nhanh, sau đó chậm
  - Ease-in-out – Bắt đầu và kết thúc chậm

# CSS3 Transition

## ➤ Transition Delay

- Khoảng thời gian delay trước khi thực hiện hiệu ứng.
- Cú pháp:
  - `Transition-delay: [Time] ;`
- Ví dụ : `Transition-delay: 1s;`

CYBERLEARN  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# CSS3 Transition

- Transition – cách viết shorthand và đầy đủ

```
.demo1,{  
    transition-property: width;  
    transition-duration: 1s;  
    transition-timing-function: linear;  
    transition-delay: 1s;  
}  
.demo2{  
    width: 200px;  
    height: 200px;  
    background-color: red;  
    transition:all 2s linear 1s;  
}
```

# CSS3 Transition

- Ví dụ: transition giúp thấy rõ thay đổi kích thước của thẻ div khi hover

```
.demo1{  
  width: 200px;  
  height: 200px;  
  background-color: red;  
  transition:all 1s linear 0s;  
}  
.demo1:hover{  
  width: 100px;  
}
```





CYBERLEARN

ĐÀO TẠO CHUYÊN NGHIỆP

# Xử lý phần Header

# Xử lý phần Header

BAKE

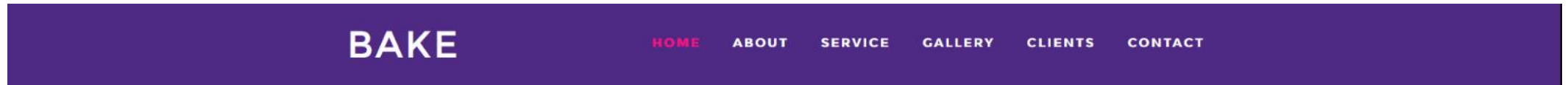
HOME ABOUT SERVICE GALLERY CLIENTS CONTACT

```
<header>
<div class="header__content">
  <p>BAKE</p>
  <nav>
    <ul>
      <li class="active"><a href="#">HOME</a></li>
      <li><a href="#">ABOUT</a></li>
      <li><a href="#">SERVICE</a></li>
      <li><a href="#">GALLERY</a></li>
      <li><a href="#">CLIENTS</a></li>
      <li><a href="#">CONTACT</a></li>
    </ul>
  </nav>
</div>
</header>
```

## ➤ Xây dựng HTML:

- Phần Header sẽ có 1 thẻ con trực tiếp.
- Bên trong div con header\_\_content có 2 thẻ con là **p** chứa chữ logo, **nav** chứa phần navigation bar

# Xử lý phần Header



- Xử lý phần navigation bar (navbar):
- Áp dụng các kiến thức đã học ở phần trước ta xử lý từng bước cho navbar:
- Xóa dấu tròn của danh sách (thẻ ul)
  - Xóa đường underline và đổi màu chữ cho thẻ a
  - Thêm màu nền để đọc được chữ
  - Xử lý các mục ( thẻ li )của navbar nằm trên 1 hàng

# Xử lý phần Header

BAKE

HOME

ABOUT

SERVICE

GALLERY

CLIENTS

CONTACT

## ➤ Xử lý phần logo:

- Dùng đơn vị **rem** để xét font-size cho logo
- Rem phụ thuộc vào font-size khai báo ở thẻ **<html>**. VD: thẻ html có font-size: 18px thì 1 rem = 18px . Nếu không khai báo font-size cho thẻ **<html>**, thì mặc định 1 rem = 16px.
- Rem thường được sử dụng cho font-size, margin, padding . Giúp chỉnh sửa nhanh chóng giá trị của css ở nhiều chỗ khác nhau trong website.

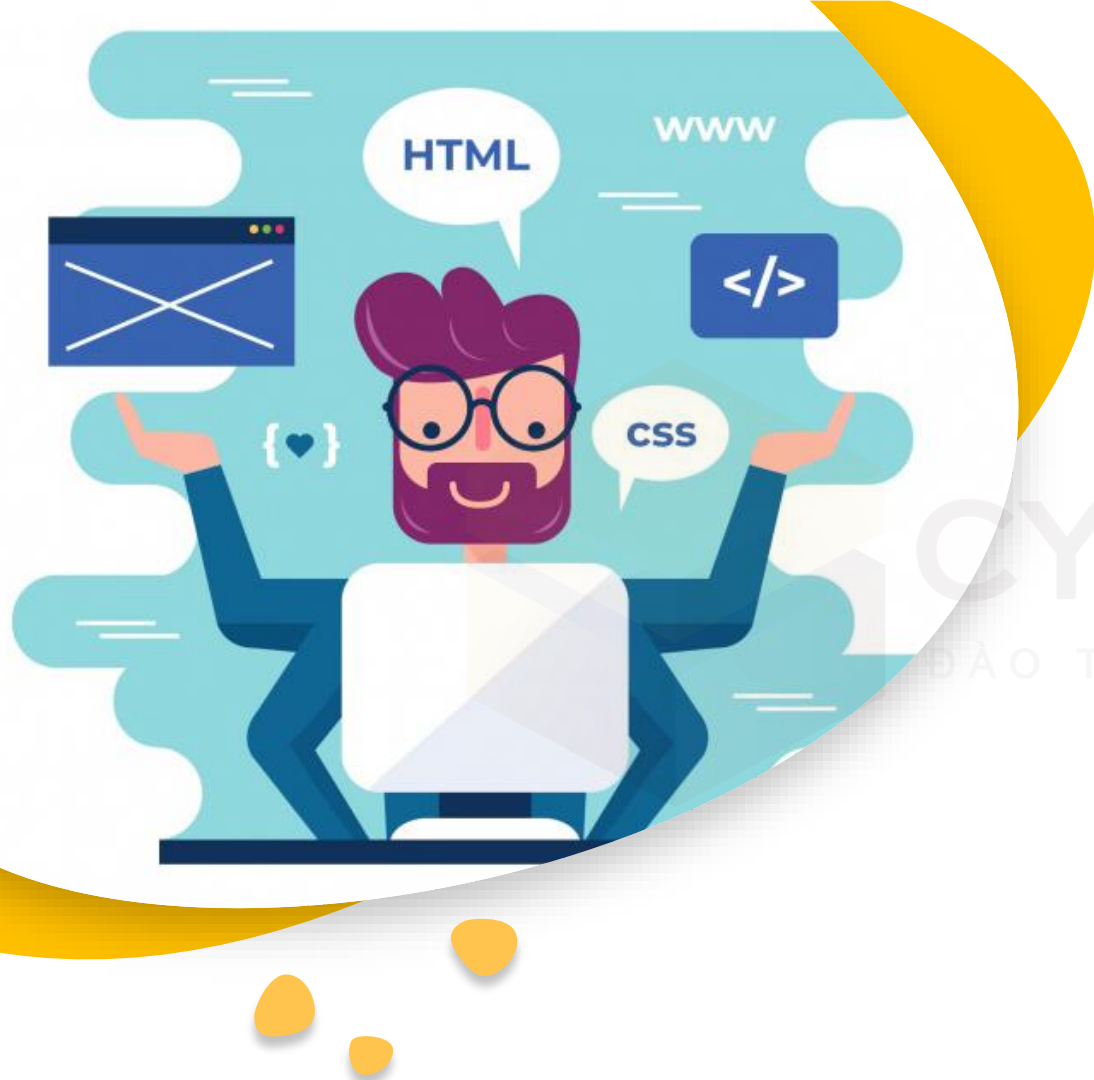
```
html{
  font-size: 16px;
}
body{
  font-family: 'Montserrat', sans-serif;
  color: #212529;
  background-color: #fff;
  letter-spacing: 1px;
}
```



CYBERLEARN

ĐÀO TẠO CHUYÊN GI

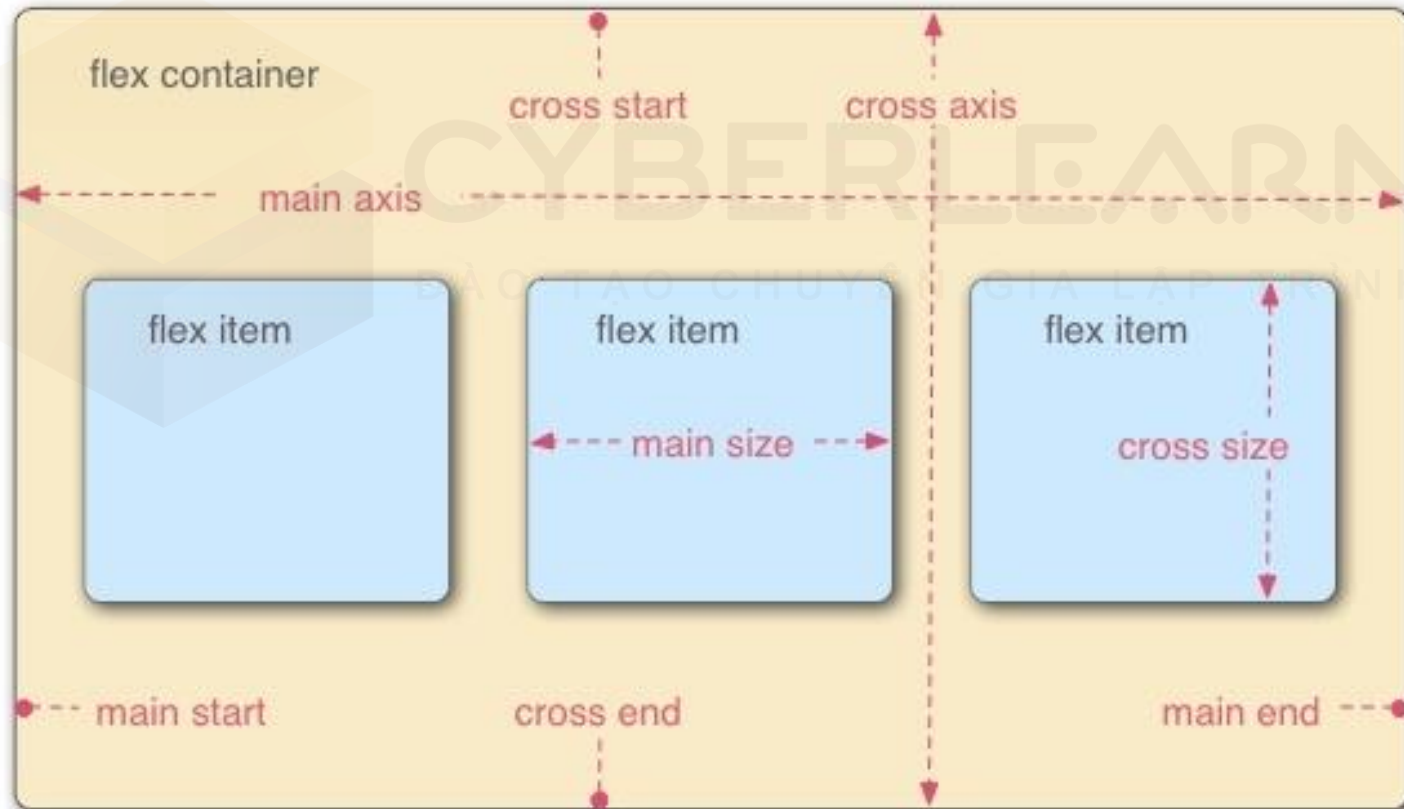
Flexbox



## Flexbox là gì

- Thay thế cho thuộc tính float và hạn chế vấn đề của float như clear float
- Tự cân đối kích thước các phần tử trên mọi thiết bị
- Chỉ nên dùng để chia bố cục trong phạm vi nhỏ

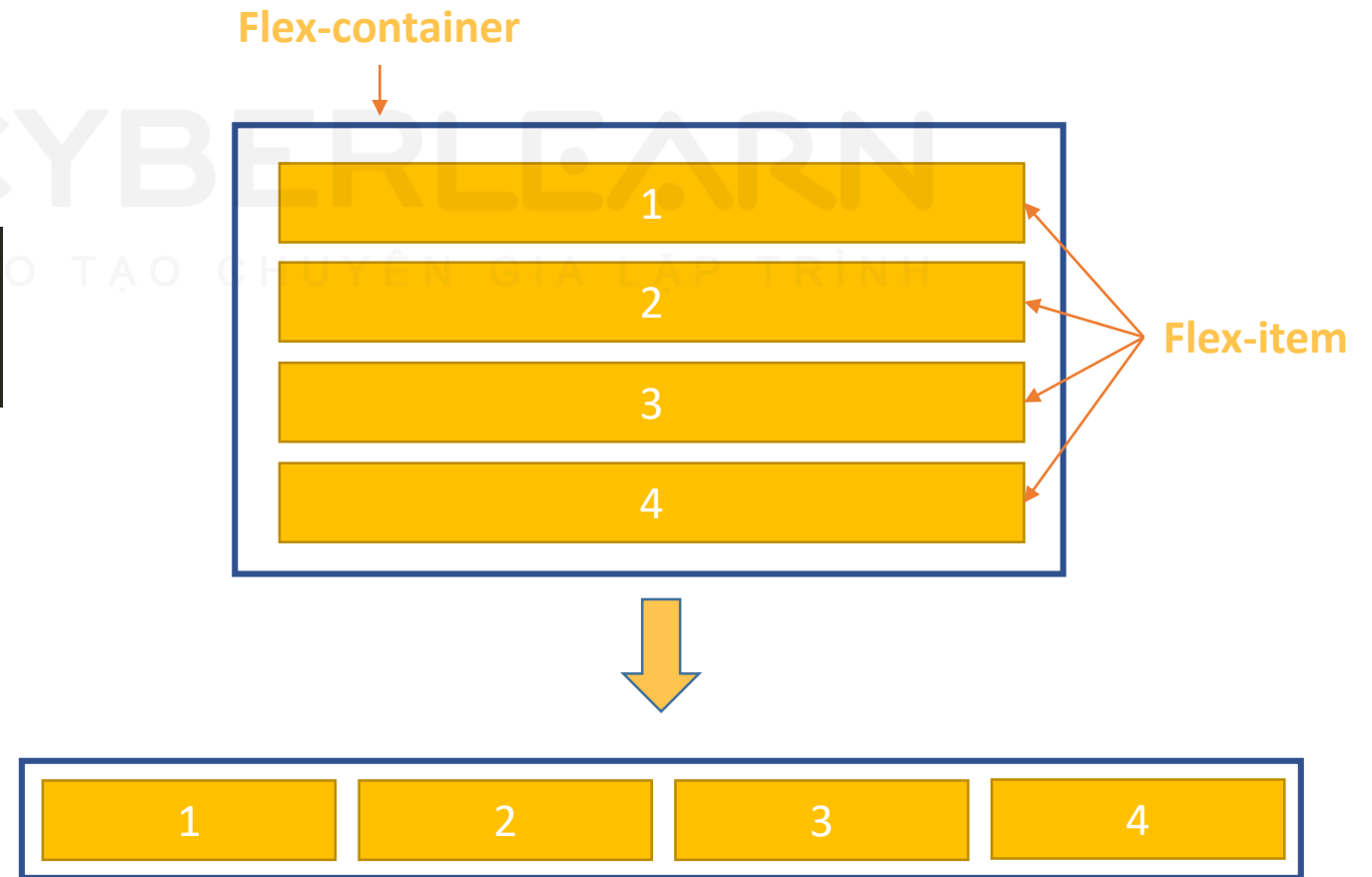
# Các thành phần trong flexbox



# Các thuộc tính của flexbox

➤ display: flex

```
.flex-container{  
  display: flex;  
}
```





# Các thuộc tính của flexbox

## ➤ flex-direction:

Dùng để đổi hướng sắp xếp của các phần tử

```
.flex-container{  
  flex-direction: row | column | row-reverse | column-reverse;  
}
```

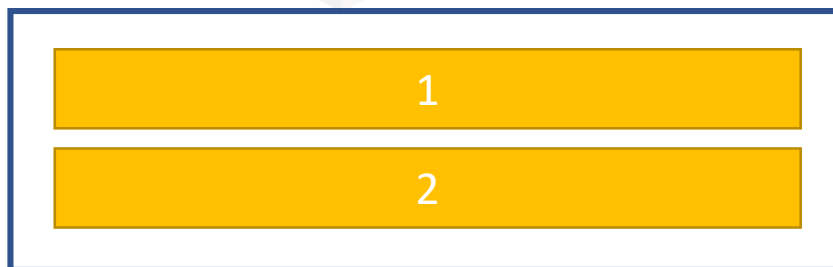
row



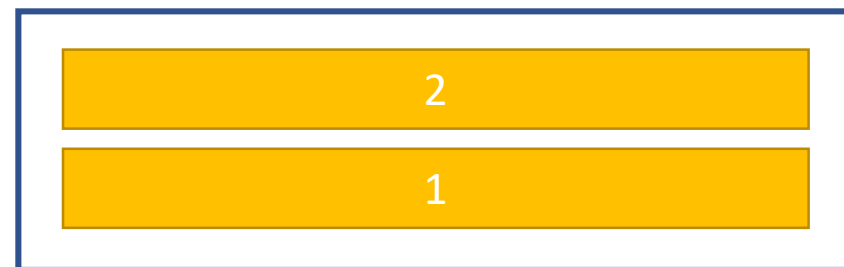
row-reverse



column



column-reverse



# Các thuộc tính của flexbox

## ➤ flex-wrap :

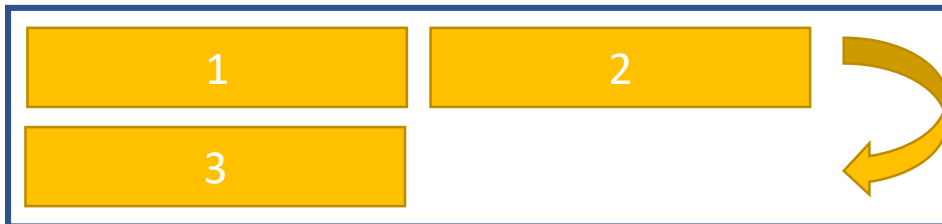
Cho phép các phần tử được xuống hàng khi không đủ chỗ chứa

```
.flex-container{  
  flex-wrap: wrap | nowrap | wrap-reverse;  
}
```

nowrap



wrap



# Các thuộc tính của flexbox

## ➤ justify-content

Canh chỉnh vị trí phần tử theo chiều ngang

```
.flex-container{  
  justify-content: flex-start | flex-end | center | space-between | space-around | space-evenly;  
}
```

Flex-start



Flex-end



center



Space-around



Space-between



Space-evenly



# Các thuộc tính của flexbox

## ➤ Align-item

Canh chỉnh vị trí phần tử theo chiều dọc

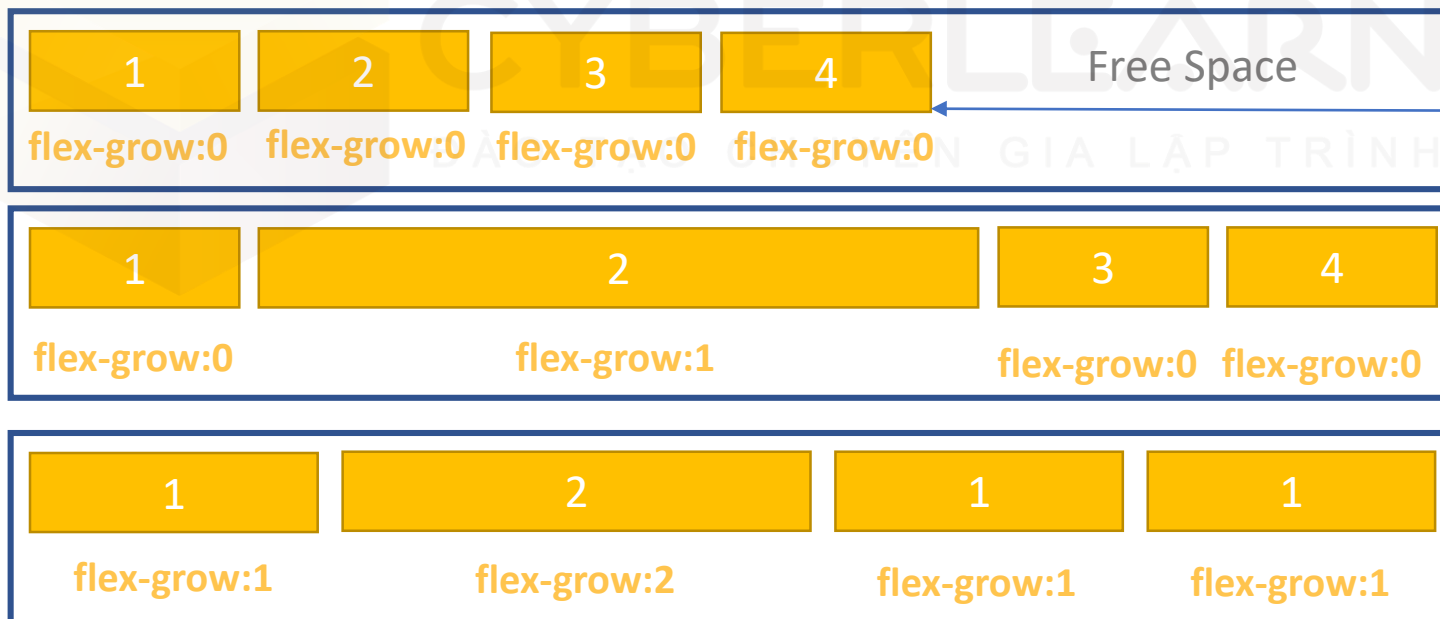
```
.flex-container{  
  align-items: flex-start | flex-end | center;  
}
```



# Các thuộc tính của flexbox

## ➤ flex-grow:

- Item được khai báo thuộc tính này sẽ được chiếm phần khoảng trống còn lại của flex.
- Mặc định: mỗi item sẽ có flex-grow là 0
- Nếu 1 item có flex-grow:2, các item còn lại là 1 thì phần trống thêm vào cho item đó sẽ gấp 2 lần so với các item khác





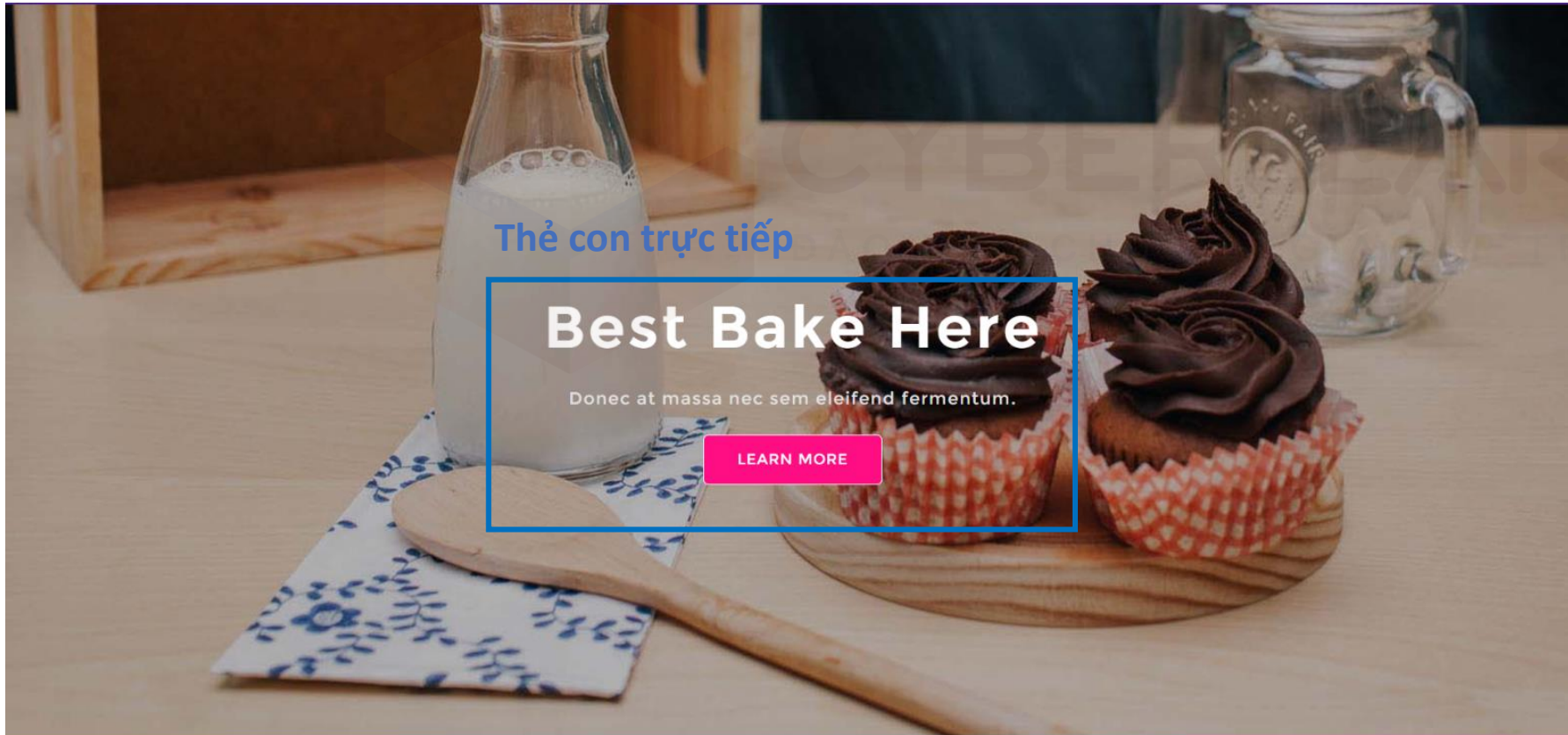
CYBERLEARN

ĐÀO TẠO CÔNG NGHỆ THÔNG TIN

# Xử lý phần Carousel

# Xử lý phần Carousel

## Thẻ cha flex



Nếu thẻ cha flex chỉ có 1 thẻ con trực tiếp, ta có thể dùng flex để căn giữa:

- Justify-content (căn chiều ngang)
- Align-items (căn chiều dọc)



CYBERLEARN

ĐÀO TẠO CHUYÊN NGHIỆP

Thank You