

```
<head>
  <!--meta-->
  <title></title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0" />
  <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
  <link rel="icon" href="/favicon.ico" type="image/x-icon">
  <!--CSS-->
  <link type="text/css" rel="stylesheet" href="css/materialize.min.css" media="screen,projection" />
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" />
  <link rel="stylesheet" href="css/animate.css">
  <link rel="stylesheet" href="css/theme.css">
</head>
```

FRONT END LAYOUT

Giới Thiệu HTML5

HTML5 là gì?

- ❖ HTML5 là một chuẩn mới và là thế hệ tiếp theo của HTML
- ❖ Giảm thiểu nhu cầu dung plugin bên ngoài.
- ❖ Cách khai báo ngắn gọn và đơn giản hơn HTML4
- ❖ Tính năng mới:
 - Tags mới: semantic elements, attributes của form, graphic elements, multimedia elements
 - APIs: geolocation, drag và drop, local storage,... (through qua javascript)

- ❖ Các trình duyệt mới hỗ trợ html5.
- ❖ Trình duyệt cũ sẽ bị lỗi “Unknown Element”.
- ❖ Dùng đoạn mã:

```
4 header, section, footer, aside, nav, main, article, figure {  
5     display: block;  
6 }
```

- ❖ Trình Duyệt IE 8 trở xuống không cho phép định dạng các unknown Elements.
- ❖ Không thể sử dụng các thẻ ngữ nghĩa ở IE8 trở xuống.
- ❖ HTML5Shiv là một file javascript và được đặt trong thẻ `<head>`.
- ❖ Link Download : <https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js>

```
<!--[if lt IE 9]>
  <script src="/js/html5shiv.js"></script>
<![endif]-->
```

- ❖ Cdn :

```
<!--[if lt IE 9]>
  <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
<![endif]-->
```

Từ HTML4 sang HTML5

Sử dụng các thẻ ngũ nghĩa thay cho div (kèm id và class)

Typical HTML4

```
<div id="header">
```

```
<div id="menu">
```

```
<div id="content">
```

```
<div class="article">
```

```
<div id="footer">
```

Typical HTML5

```
<header>
```

```
<nav>
```

```
<section>
```

```
<article>
```

```
<footer>
```

❑ DOCTYPE

HTML4: <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd">

HTML5: <!DOCTYPE html>

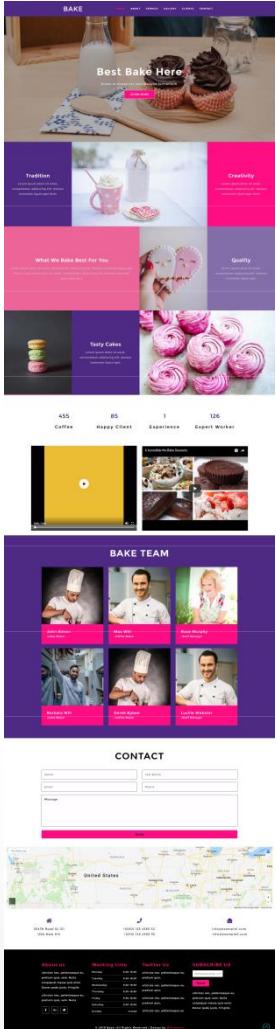
❑ ENCODING

HTML4: <meta http-equiv="Content-Type" content="text/html; charset=utf-8">

HTML5: <meta charset="utf-8">

BAKE LAYOUT

BAKE LAYOUT



- ❖ Dàn layout với HTML5
- ❖ Sử dụng các kĩ thuật mới: flexbox, css grid layout
- ❖ Tạo ra form nhập liệu trong html
- ❖ Hiển thị video, audio lên trang web
- ❖ Tạo hiệu ứng sinh động cho web

Phần 1: Tạo cấu trúc trang

- ❖ Tạo cấu trúc trang HTML5 chuẩn với HTML5 shiv
- ❖ Fix lỗi unknown element trong css
- ❖ Cấu hình css dùng chung cho cả trang web

```
<!DOCTYPE html>
<html lang="en">

  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
    <link rel="stylesheet" href="css/index.css">      Sử dụng google font Roboto thông qua cdn
    <link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700" rel="stylesheet">
    <!--[if lt IE 9]>
      <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js">
    </script>
    <![endif]-->

  </head>

  <body>

  </body>

</html>
```

```
header,section,footer,aside,nav,main,article,figure{
  display:block
}
/* GLOBAL CSS */
body{
  margin:0;
  padding:0;
  font-family: 'Roboto', sans-serif;
}
html{
  font-size:16px;
}
*{
  box-sizing: border-box;
}
.section-padding{
  padding:0 10%;
}
/* END GLOBAL CSS */
```

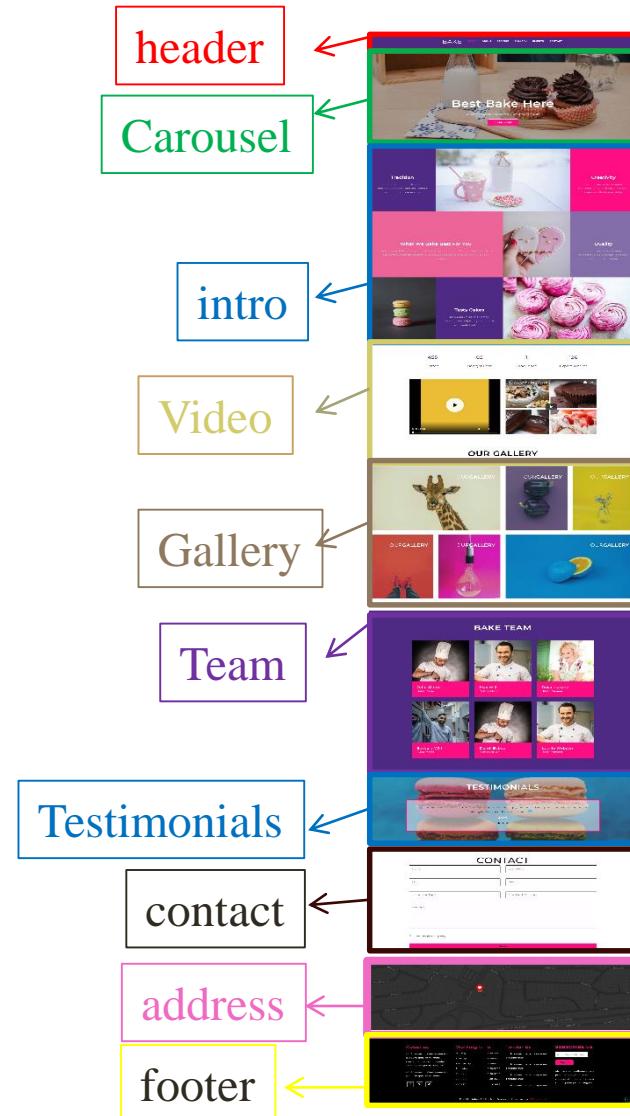
Để sử dụng đơn vị mới trong css là rem,
cần phải xét font-size ở thẻ html, ở đây
tương ứng 1rem = 16px

Phần 2: Chia bố cục trang web

```

<div class="wrapper">
    <!--BEGIN HEADER-->
    <header class="header"></header>
    <!--END HEADER-->
    <!--BEGIN CAROUSEL-->
    <section class="carousel"></section>
    <!--END CAROUSEL-->
    <!--BEGIN INTRO-->
    <section class="intro"></section>
    <!--END INTRO-->
    <!--BEGIN VIDEO-->
    <section class="video"></section>
    <!--END VIDEO-->
    <!--BEGIN GALLERY-->
    <section class="gallery"></section>
    <!--END GALLERY-->
    <!--BEGIN TEAM-->
    <section class="team"></section>
    <!--END TEAM-->
    <!--BEGIN TESTIMONIALS-->
    <section class="testimonials"></section>
    <!--END TESTIMONIALS-->
    <!--BEGIN CONTACT-->
    <section class="contact"></section>
    <!--END CONTACT-->
    <!--BEGIN ADDRESS-->
    <section class="address"></section>
    <!--END ADDRESS-->
    <!--BEGIN FOOTER-->
    <section class="footer"></section>
    <!--END FOOTER-->
</div>

```



Phần 3: Review (Sematic Elements)

THẺ NGỮ NGHĨA



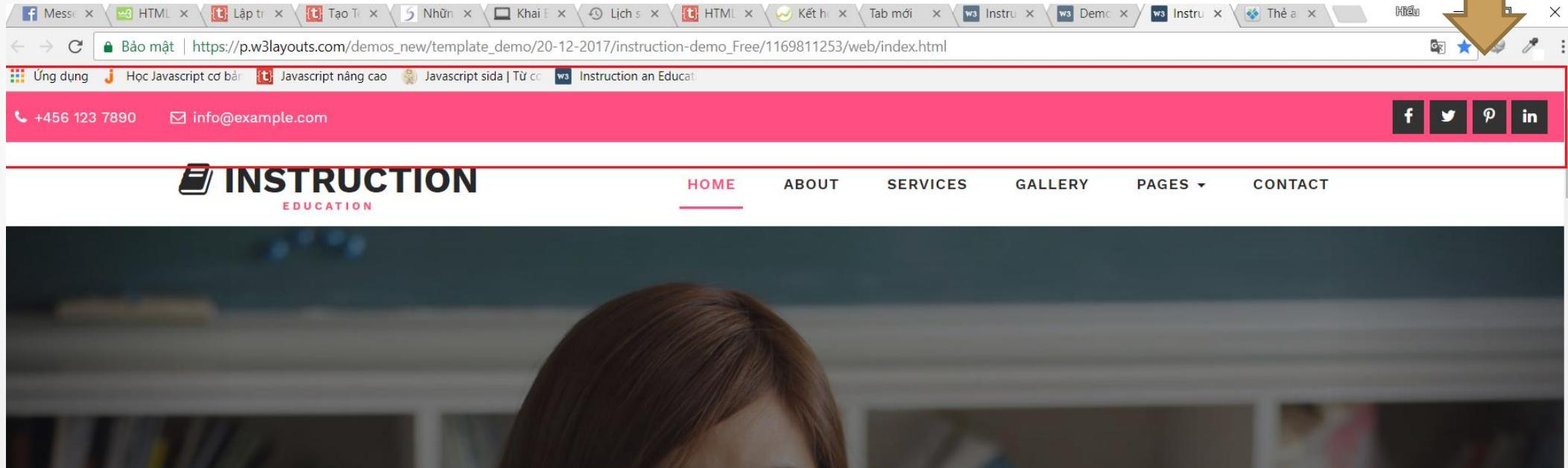
- HTML4 sử dụng **div** id/class cho các thẻ cơ bản: header, content, footer....
- HTML5 dung các thẻ ngữ nghĩa:<header>,<footer>,...do đó dễ xác định được nội dung trang web
- Để các công cụ tìm kiếm có thể tìm đúng được nội dung được hiển thị trên web, tránh lạm dụng thẻ <div>

<header>



Header trang web

- Định nghĩa header cho một section trang web, section, article....



<footer>



- Định nghĩa footer cho một section hoặc cho một trang
- Chứa thông tin: tác giả, copyright information, điều kiện sử dụng, thông tin liên hệ, form liên hệ...

```
<footer>
  <p>Posted by: Hege Refsnes</p>
  <p>Contact information: <a href="mailto:someone@example.com">
    someone@example.com</a>.</p>
</footer>
```

About Us

Nam libero tempore cum vulputate id est id, pretium semper enim. Morbi viverra congue nisi vel pulvinar posuere sapien eros.

Latest News

- Lorem ipsum neque vulputate
- Dolor amet sed quam vitae
- Lorem ipsum neque, vulputate
- Dolor amet sed quam vitae
- Lorem ipsum neque, vulputate

Contact Info

- 8088 USA, Honey block, New York City.
- info@example.com
- +09187 8088 9436

Sign up for our Newsletter

Enter Email Address...

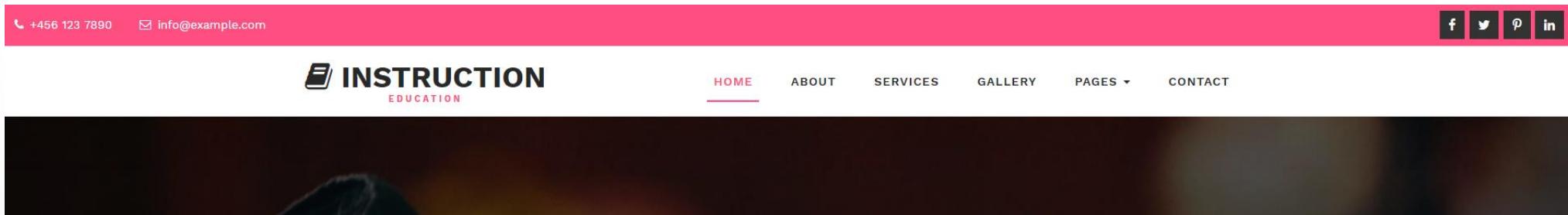
SUBMIT

© 2018 Instruction. All rights reserved | Design by W3layouts

<nav>

- Định nghĩa thanh navigation

```
<nav>
  <a href="/html/">HTML</a> |
  <a href="/css/">CSS</a> |
  <a href="/js/">JavaScript</a> |
  <a href="/jquery/">jQuery</a>
</nav>
```



A screenshot of a website header. The top bar is pink. On the left, there are contact links: a phone icon followed by '+456 123 7890' and an envelope icon followed by 'info@example.com'. On the right, there are social media icons for Facebook, Twitter, Pinterest, and LinkedIn. Below the pink bar is a dark banner with the text 'INSTRUCTION EDUCATION'. To the right of the banner is a navigation menu with links: HOME (underlined), ABOUT, SERVICES, GALLERY, PAGES ▾, and CONTACT.

<section>

- Định nghĩa các section:
 - Chia nội dung trang web thành nhiều khu vực chính. Mỗi khu vực chính sẽ là một section.
 - Mỗi section sẽ chứa nhiều nội dung liên quan với nhau.
 - Cách dùng giống như thẻ div... Có thể thêm id, class...

```
<section>
  <h1>SECTION TAG</h1>
  <p>The Section tag element defines a section in a document</p>
</section>
```

<article>



- Chứa các nội dung độc lập, khép kín với phần còn lại của trang web.
- Người dùng có thể đọc nội dung bên trong article mà không cần quan tâm tới các phần khác
- Ví dụ: **một bài viết, bài báo, bình luận, tin tức, sự kiện....**

```
<article>
  <h1>What Does WWF Do?</h1>
  <p>WWF's mission is to stop the degradation of our planet's natural environment,
  and build a future in which humans live in harmony with nature.</p>
</article>
```

Phần 4: Xử lý phần header



(1) : Quy tắc đặt tên biến BEM

__ là biểu hiện của class con. Vd: header__nav là class con của header

(2): thẻ nav của html5 dùng để chứa menu

(3): class chung, dùng để chỉnh padding cho thẻ , trái phải mỗi bên 10%

```
<!--HEADER-->
<header class="header section-padding"> (3)
    <a href="#" class="header__logo">BAKE</a>
    <nav class="header__nav"> (1)
        <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="">CLIENT</a></li>
            <li><a href="">CONTACT</a></li>
        </ul>
    </nav>
</header>
<!--END HEADER-->
```

HOME ABOUT SERVICE GALLERY CLIENTS CONTACT

```
/* HEADER CSS */
.header{
    background-color: #4E2A84;
    display: flex;
    align-items: center;
    justify-content:center;
    padding:10px 0;
}
.header__logo{
    color: white;
    font-size:2.5rem;
    text-decoration: none;
    font-weight: 500;
    margin-right:10%;
}
.header__nav ul{
    list-style: none;
    margin:0;
}
.header__nav ul li{
    display: inline;
}
.header__nav ul li a{
    color: white;
    font-size:0.8rem;
    text-decoration: none;
    font-weight: 500;
    margin:0 0.7rem;
}
.header__nav ul li.active a{
    color: #FF0E83;
}
/* END HEADER CSS */
```

(1) Kỹ thuật flexbox được tạo ra để thay thế cho float. Trong đó

- ❖ Display:flex là khai báo sử dụng flex, các phần tử con trực tiếp của .header sẽ nằm trên 1 hàng, mặc định sẽ tự điều chỉnh kích thước để luôn nằm trên 1 hàng và không xuống hàng trừ khi ta xét.
- ❖ Align-items là căn lè theo chiều dọc
- ❖ Justify-content là căn lè theo chiều ngang

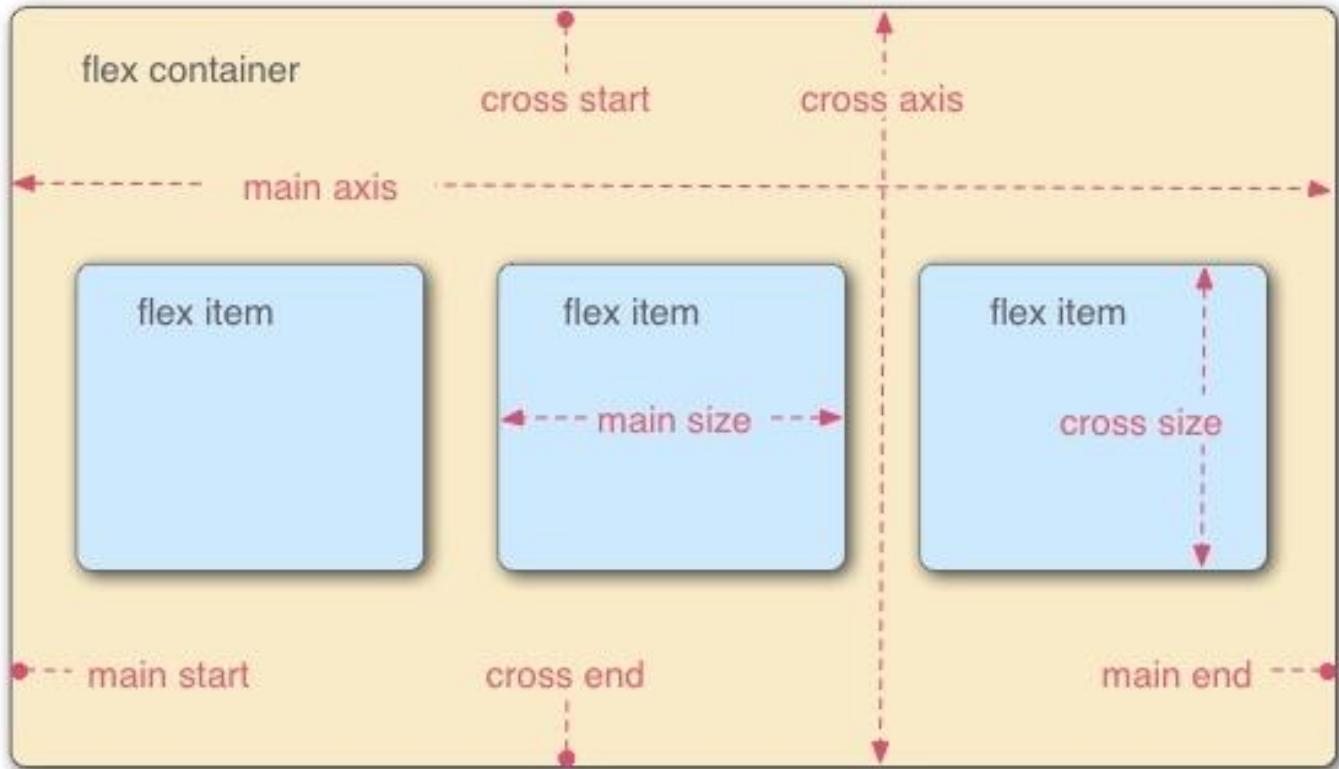
(2) Gọi tới thẻ a nằm bên trong li mà li đó được gắn class active

Phần 5: review (flexbox)

Flexbox là gì ?

- Thay thế cho thuộc tính float và kĩ thuật 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

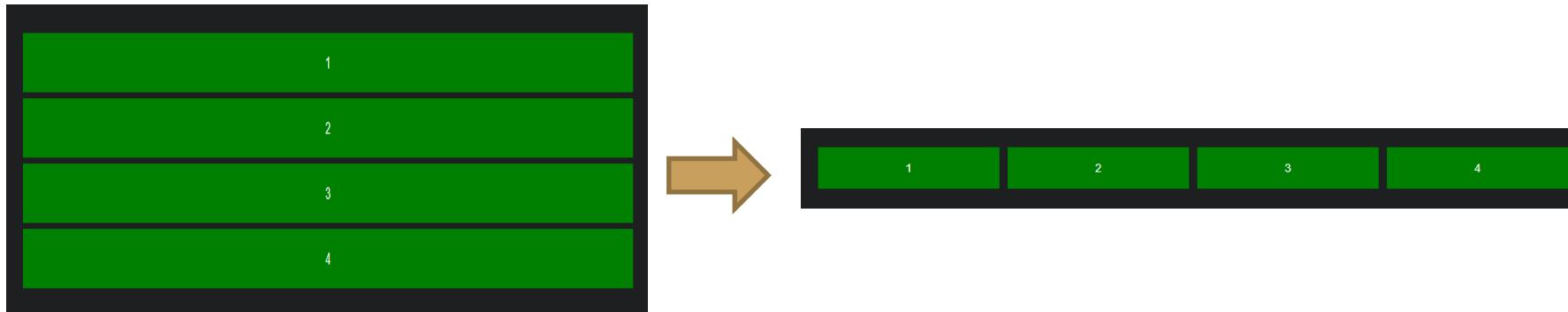
Các thành phần trong flexbox



Các thuộc tính của flexbox

1) display : flex

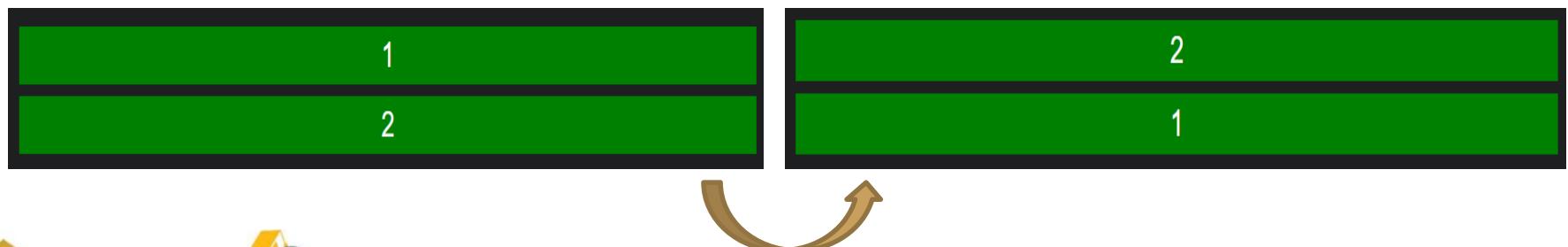
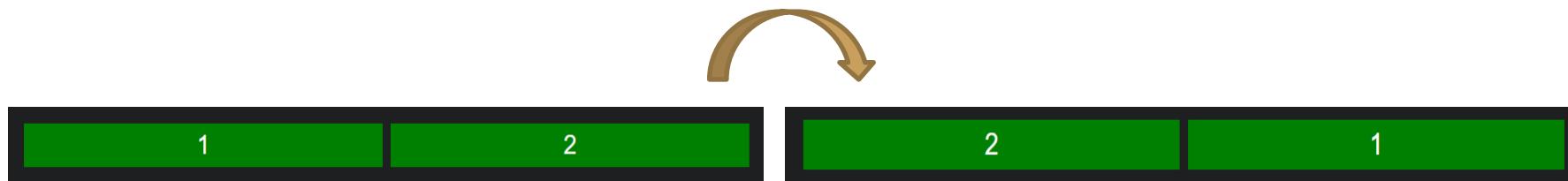
```
.flex-container{  
    display: flex;  
}
```



Các thuộc tính của flexbox

2) flex-direction: dùng để đổi hướng sắp xếp của các phần tử

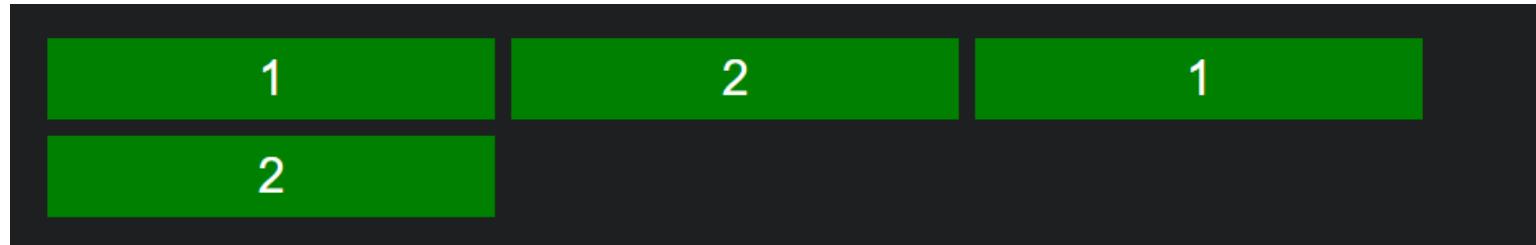
```
.flex-container{  
    flex-direction: row | column | row-reverse | column-reverse;  
}
```



Các thuộc tính của flexbox

3) flex-wrap: cho phép các phần tử xuống hàng khi không đủ chỗ
chứa hay ko

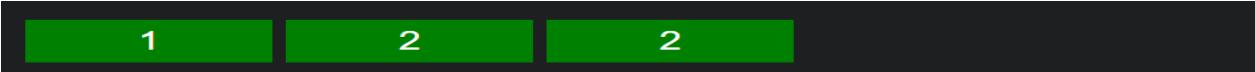
```
.flex-container{  
    flex-wrap: wrap | nowrap | wrap-reverse;  
}
```



Các thuộc tính của flexbox

3) justify-content: căn chỉnh vị trí phần tử theo chiều ngang
Align-item: căn chỉnh vị trí phần tử theo chiều dọc

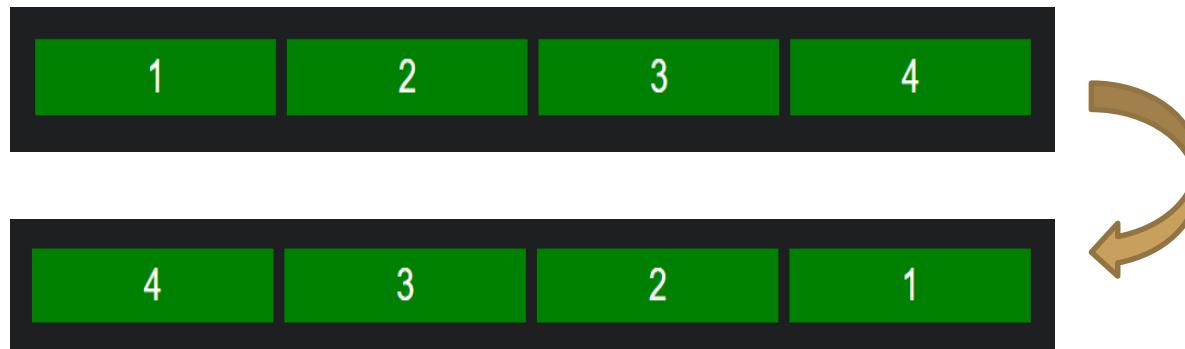
```
.flex-container{  
    justify-content:flex-start | flex-end | center;  
}
```



Các thuộc tính của flexbox

4) Order: sắp xếp thứ tự các phần tử trong flexbox

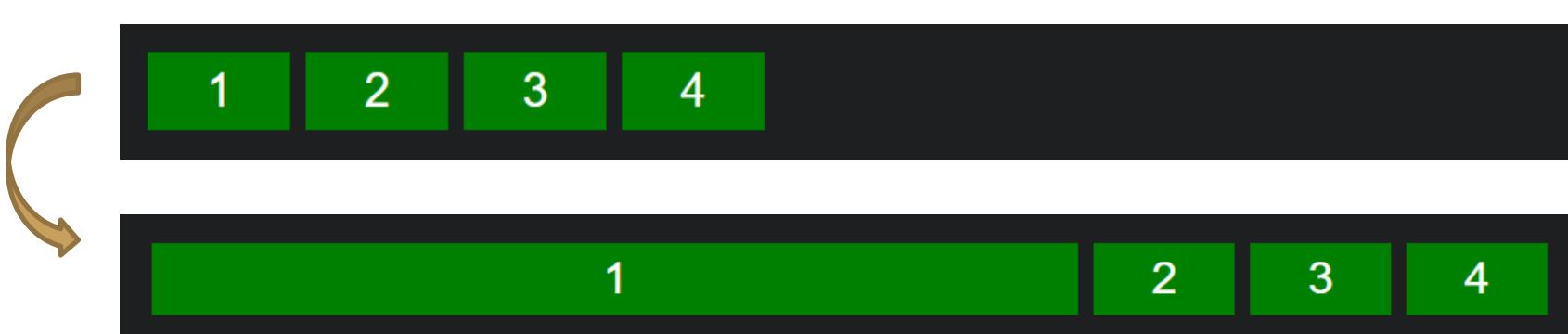
```
.item1{  
    order:4;  
}  
.item2{  
    order:3;  
}  
.item3{  
    order:2;  
}  
.item4{  
    order:1;  
}
```



Các thuộc tính của flexbox

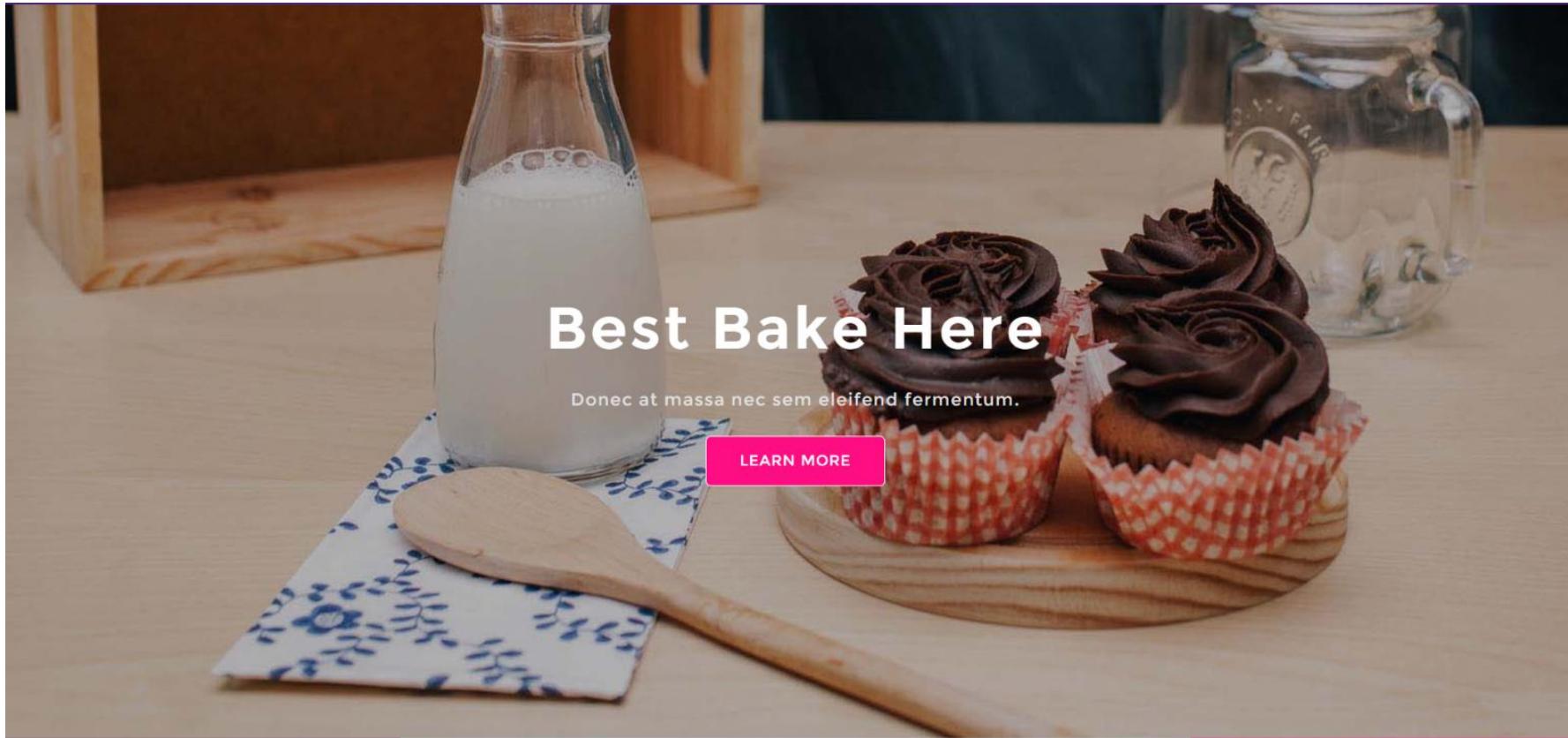
5) flex-grow: item được khai báo thuộc tính này sẽ chiếm toàn bộ diện tích còn trống trong flexbox

```
.item1{  
    flex-grow: 1;  
}
```



Phần 6: Xử lý phần carousel

Yêu cầu phần Carousel



Xử lý phần Carousel

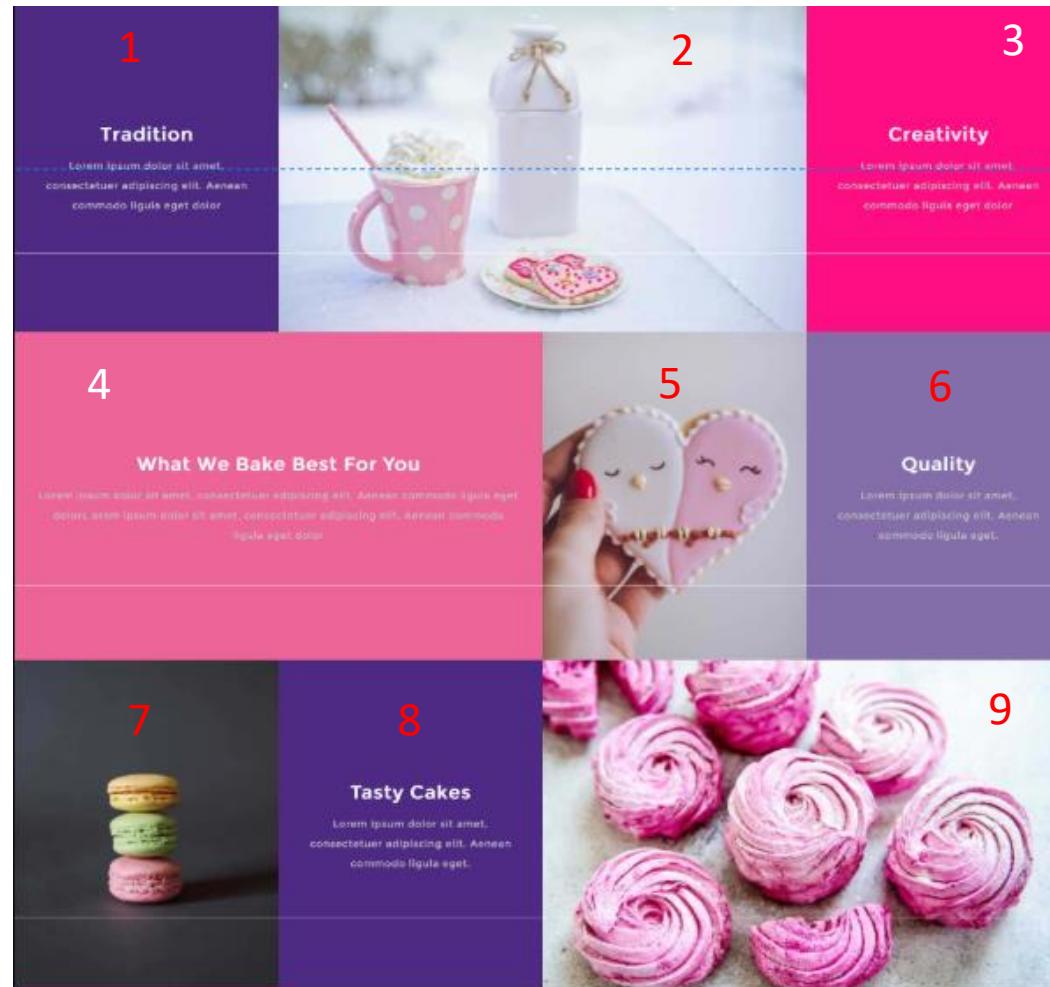


```
<!--CAROUSEL-->
<section class="carousel">
  <div class="carousel__content">
    <h1>Best Bake Here</h1>
    <p>
      Lorem ipsum dolor sit amet consectetur
      adipisicing elit. Consequuntur
    </p>
    <button class="btn-pink">Learn More</button>
  </div>
</section>
<!-- END CAROUSEL -->
```

```
/* CAROUSEL CSS */
.carousel{
  height:35rem;
  background:url(..../images/b3.jpg) no-repeat center center;
  background-size:cover;
  display: flex;
  align-items: center;
}
.carousel__content{
  width:100%;
  text-align: center;
  color:#white
}
.carousel__content h1{
  font-size:3rem;
  font-weight: 500;
  word-spacing: 0.4rem; Khoảng cách giữa các từ
  margin-bottom:0;
}
.carousel__content p{
  margin-top:1rem;
  margin-bottom:0.8rem;
  font-size:1rem;
  color:#rgb(221, 221, 221);
}
.btn-pink{
  background-color:#ff0c82;
  padding:0.5rem 1rem;
  border:1px solid #rgb(221, 221, 221);
  color:#white;
  text-transform: uppercase; Chuyển đổi text thành chữ hoa
  font-size:0.7rem;
  border-radius: 3px; Độ cong viền của phần tử
}
/* END CAROUSEL CSS */
```

Phần 7: Xử lý phần Intro

Yêu cầu phần Intro



Xử lý HTML phần Intro

- ❖ Ở HTML, 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 sắp xếp.

```
<!--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>
<!-- END INTRO -->
```

1

2

3

4

5

6

7

8

9

```
<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 -->
```

Xử lý CSS 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

Phần 8 : Review (CSS GRID LAYOUT)

CSS Grid layout là gì ?

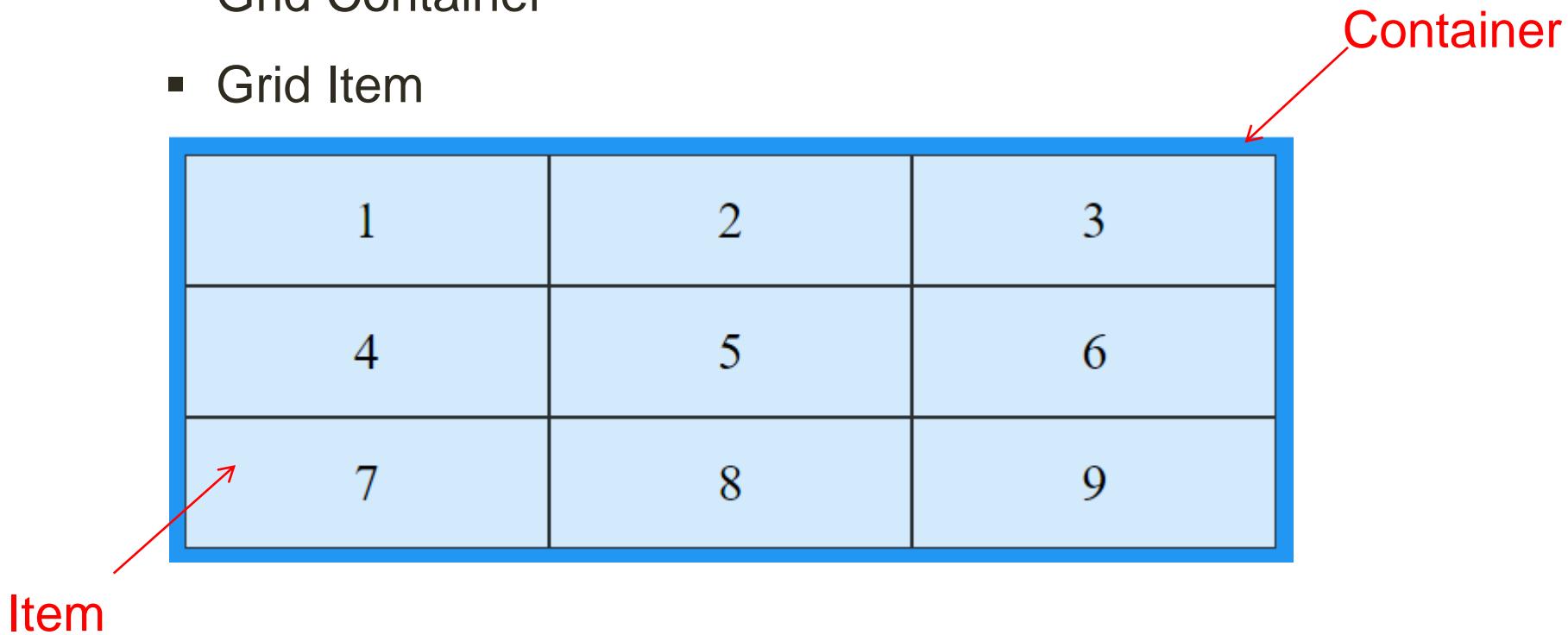
- Cung cấp một hệ thống lưới cơ bản
- Bao gồm các cột và các hàng
- Dàn layout dễ hơn mà không phải sử dụng float hay position.

				
57.0	16.0	52.0	10	44

Các thành phần chính

Gồm 2 thành phần chính :

- Grid Container
- Grid Item



Các thuộc tính của phần tử cha

Muốn sử dụng css Grid phải có các thuộc tính :

- ✓ Display: grid
- ✓ Grid-template-columns: chia cột cho phần tử

```
<!DOCTYPE html>
<html>
<head>
<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>
```

The display Property:

1	2	3
4	5	6
7	8	9

Set the *display* property to *grid* to make a block-level grid container.

Grid-template-columns

Dùng để chia cột cho phần tử

Cú pháp :

Grid-template-columns:[col1] [col2] [col3]...;

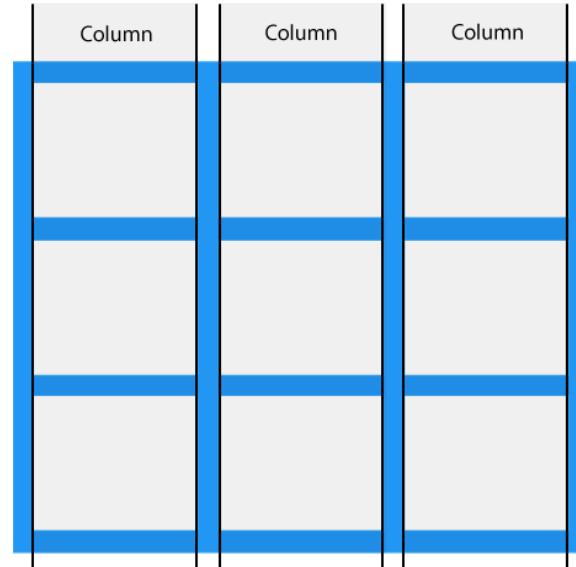
Grid-template-columns: repeat(số cột, kích thước);

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



Grid-template-columns

```
<!DOCTYPE html>
<html>
<head>
<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>
```

The display Property:

1	2	3
4	5	6
7	8	9

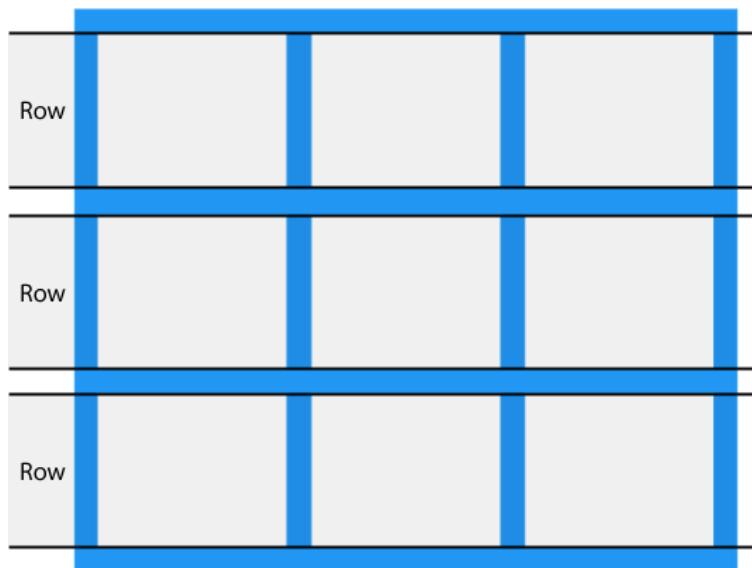
Set the *display* property to *grid* to make a block-level grid container.

Grid-template-rows

Cú pháp :

Grid-template-rows:[row1] [row2] [row3]...;

Chỉ nên xét khi muốn chỉnh height cho các rows.



Grid-template-rows

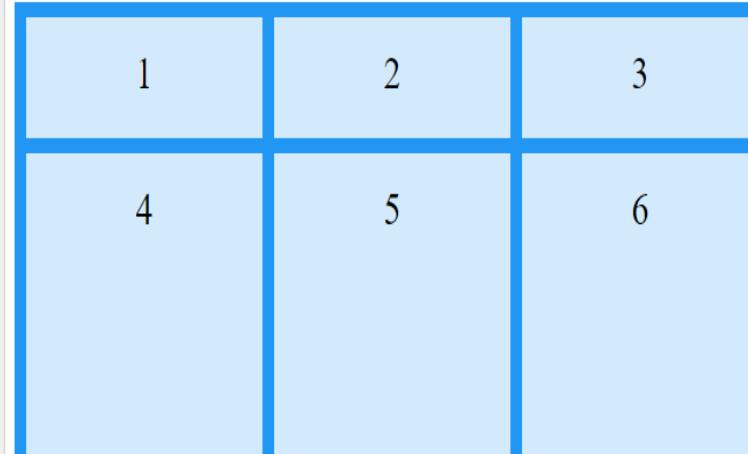
Cú pháp :

Grid-template-rows:[row1] [row2] [row3]...;

Chỉ nên xét khi muốn chỉnh height cho các rows.

```
<html>
<head>
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
  grid-template-rows: 80px 200px;
  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;
}
</style>
</head>
<body>
```

The grid-template-rows Property

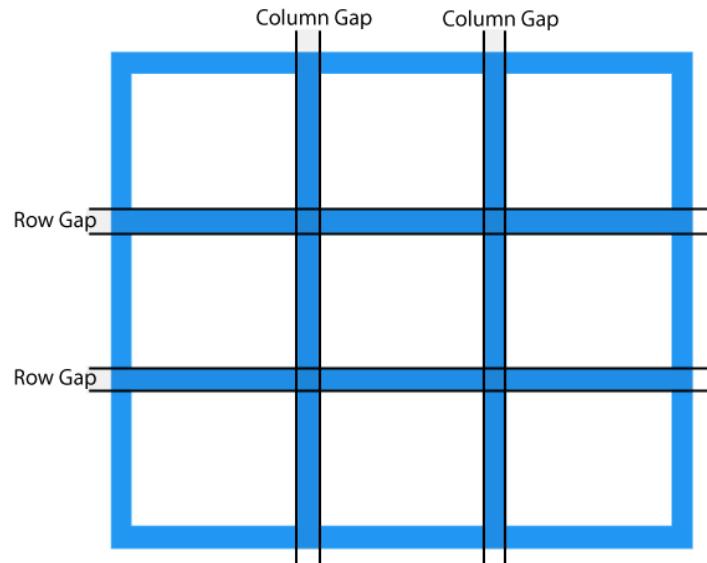


1	2	3
4	5	6

Grid Gaps

- Là khoảng cách giữa các cột và các hàng.
- Cú pháp :

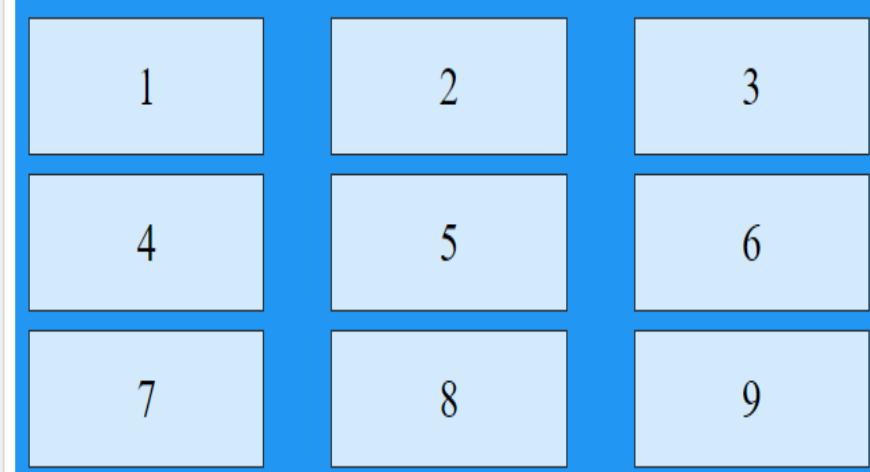
Grid-column-gap | grid-row-gap | grid-gap : [khoảng cách];



Grid Gaps

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
  display: grid;
  grid-column-gap: 50px;
  grid-row-gap: 10px;
  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>
```

The grid-column-gap Property:



Use the *grid-column-gap* property to adjust the space between the columns.

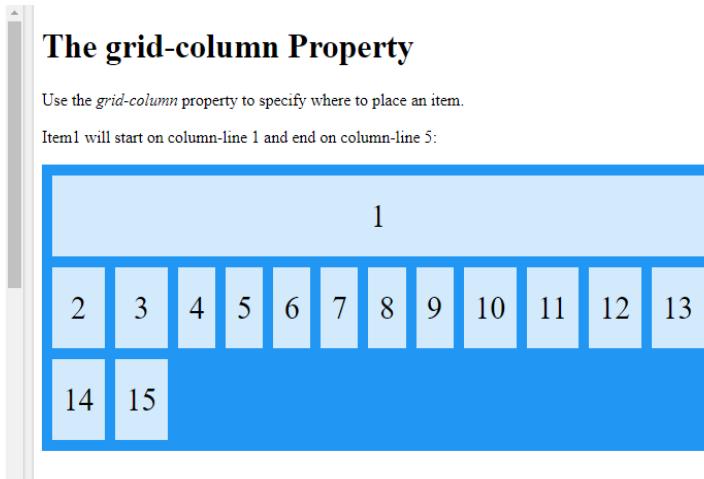
Các thuộc tính của phần tử con

❖ Grid-column

- Ta có thể tiến hành gom cột hoặc hàng tương tự như table
- Cú pháp:

grid-column: n / m;

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto auto auto auto auto auto 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: 1 / 13;
}
</style>
</head>
<body>
```



Grid-column

Cú pháp: **grid-column: n / span m**

Trong đó: n là vị trí cột phần tử đang nằm

m là số cột ta muốn phần tử này chiếm

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto 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: 1 / span 3;
}
</style>
</head>
<body>

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

The grid-column Property

Use the *grid-column* property to specify where to place an item.

Item1 will start on column-line 1 and span 3 columns:

1	2	3	4
5	6	7	8
11	12	13	14

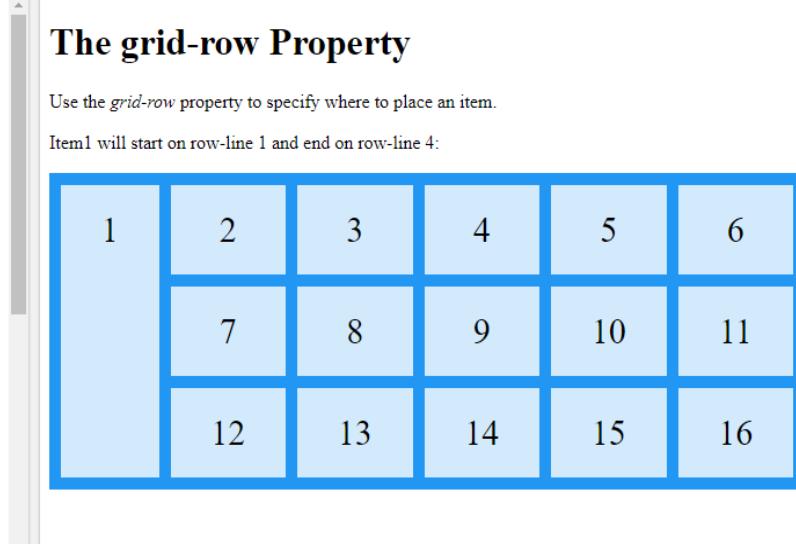
Grid-row

Cú pháp: **grid-row: n / span m**

Trong đó : n là vị trí dòng phần tử này đang nằm
m là số dòng ta muốn nó chiếm

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto 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: 1 / 4;
}
</style>
</head>
<body>

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

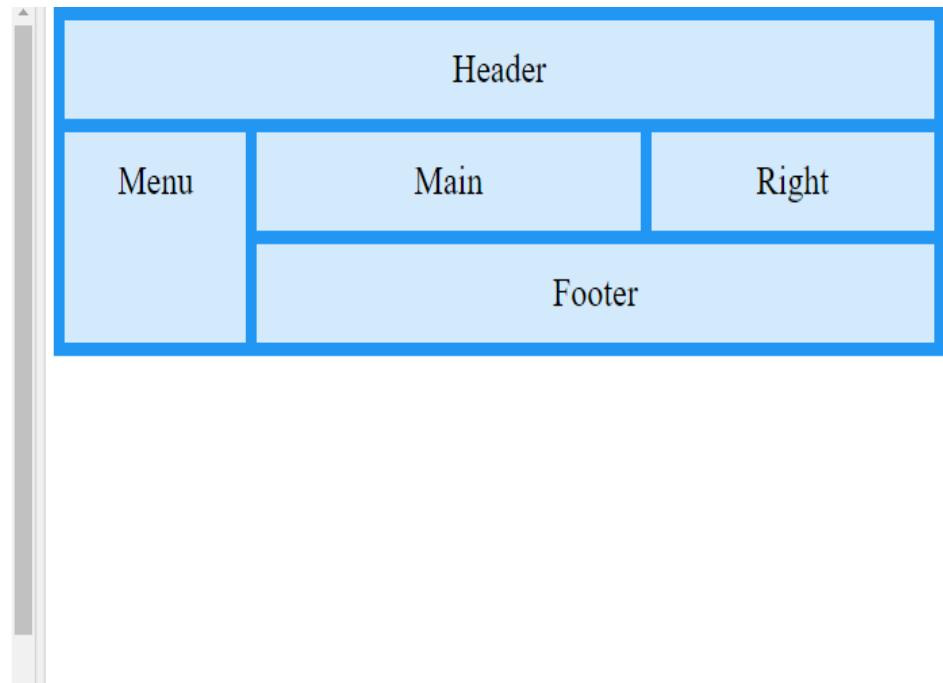


Naming Grid Item

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

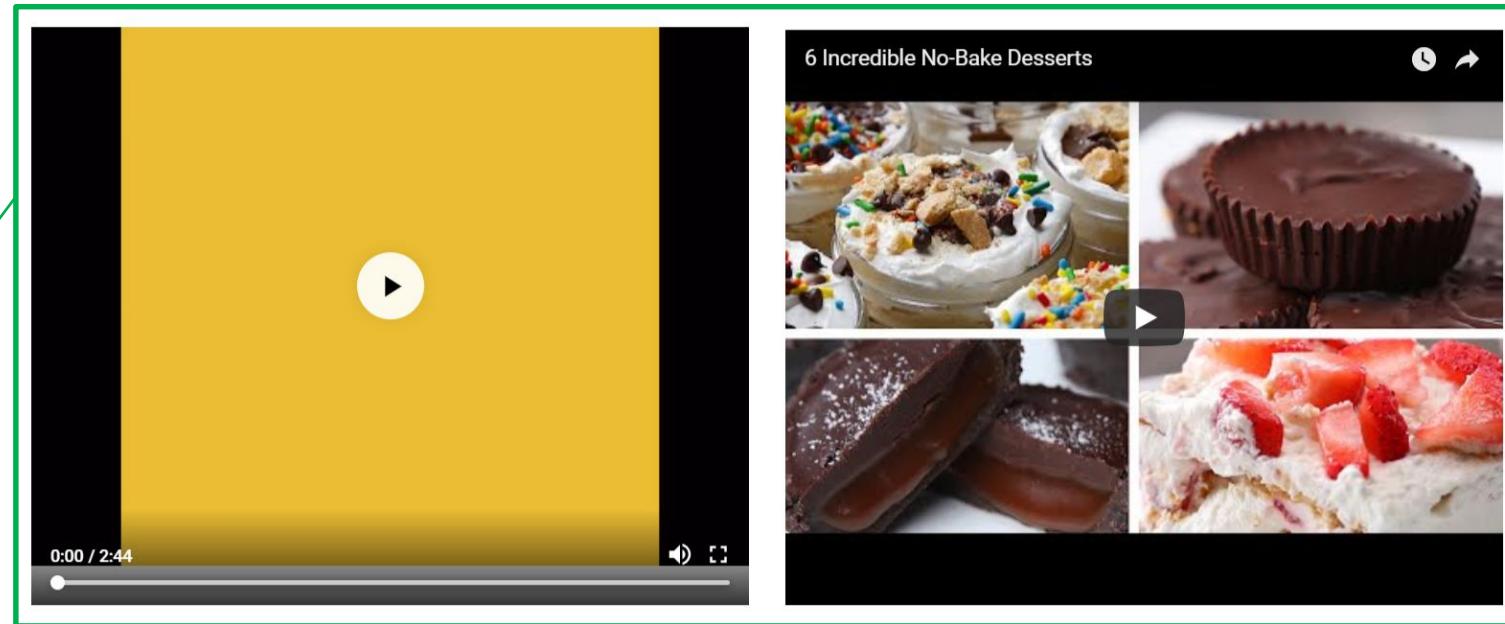
```
<!DOCTYPE html>
<html>
<head>
<style>
.item1 { grid-area: header; }
.item2 { grid-area: menu; }
.item3 { grid-area: main; }
.item4 { grid-area: right; }
.item5 { grid-area: footer; }

.grid-container {
  display: grid;
  grid-template-areas:
    'header header header header header'
    'menu main main main right right'
    'menu footer footer footer footer';
  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;
}
</style>
</head>
<body>
```



Phần 9: Xử lý phần Video

Yêu cầu phần Video



Xử lý HTML phần Video

```

<section class="video">
    <div class="statistic">
        <div class="statistic__item">
            <h2>455</h2>
            <p>Coffee</p>
        </div>
        <div class="statistic__item">
            <h2>85</h2>
            <p>Happy Client</p>
        </div>
        <div class="statistic__item">
            <h2>1</h2>
            <p>Experience</p>
        </div>
        <div class="statistic__item">
            <h2>126</h2>
            <p>Expert Worker</p>
        </div>
    </div>

    <div class="video__content">
        <div class="video_content_left">
            <video src="video/Gioi_thieu.mp4" controls></video> (1)
        </div>
        <div class="video_content_right">
            <iframe src="https://www.youtube.com/embed/vvR1pde64ZA" frameborder="0" allow="autoplay; encrypted-media"
                allowfullscreen>
            </iframe>
        </div>
    </div>
</section>
```

(1): thẻ video mới của html5 để hiển thị video lên trang web, ở thuộc tính src truyền vào đường dẫn tới video

(2): Thẻ iframe dùng để nhúng video youtube vào trang web.
Cách lấy: click chuột phải vào video youtube, chọn “sao chép mã nhúng”

Xử lý CSS phần Video

```
.video{  
    padding:0 15%;  
}  
  
.statistic{  
    display: grid;  
    grid-template-columns: repeat(4,1fr);  
    margin:4rem 0;  
}  
  
.statistic__item{  
    text-align: center;  
}  
  
.statistic__item h2{  
    font-size:1.5rem;  
    color:#4E2A84;  
    margin-top:0;  
    margin-bottom: 1.5rem;  
}  
  
.statistic__item p{  
    font-size:1rem;  
    color:black;  
    margin:0;  
}  
  
.video__content{  
    display: grid;  
    grid-template-columns: repeat(2,1fr);  
    grid-column-gap: 1.5rem; Grid-column-gap: dùng để chỉnh  
    khoảng cách giữa các cột  
}  
  
.video__content__right iframe{  
    width: 100%;  
    height: 20rem;  
}  
  
.video__content__left video{  
    width: 100%;  
    height: 20.5rem;  
    transform: translateY(-0.5rem);  
}
```

Chia phần statistic thành 4 cột , với hàm repeat là hàm lặp, ở đây nghĩa là ta muốn lặp lại cột 1fr 4 lần, tức là sẽ có 4 cột, mỗi cột 1 fr

Grid-column-gap: dùng để chỉnh khoảng cách giữa các cột

Dùng để dịch chuyển phần tử đi theo trục Y, dịch ngược lên trên 0.5rem

Phần 10: Review (Multimedia Elements)

<audio>



- Thêm âm thanh (audio) cho trang web với các thuộc tính:
 - Controls: hiển thị thanh điều khiển video
 - Autoplay: tự động chạy audio khi load trang web

```
<audio controls preload="none" autoplay>
    <source src="sources/Epic-Music-Brotherhood-John-Dreamer.mp3" type="audio/mpeg">
</audio>

<audio controls preload="none">
    <source src="sources/Epic-Music-Brotherhood-John-Dreamer.mp3" type="audio/mpeg">
</audio>

<audio controls preload="none" autoplay loop>
    <source src="sources/Epic-Music-Brotherhood-John-Dreamer.mp3" type="audio/mpeg">
</audio>
```

<video>



- Thêm video cho trang web với các thuộc tính
 - Controls: Hiển thị thanh điều khiển video
 - Preload: load trước video
 - Loop: lặp video

```
<h1>HTML5 TUTORIAL</h1>
<h3>VIDEO TAG</h3>
<video width="320" height="240" controls>
  <source src="sources/epic.mp4" type="video/mp4">
</video>
```

<iframe>

- Dùng để nhúng video vào trang web (từ youtube) : click chuột phải vào video trên youtube, chọn “sao chép mã nhúng”

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/rJN4GRoaSHs"
frameborder="0" allow="autoplay; encrypted-media" allowfullscreen></iframe>
<iframe width="560" height="315" src="https://www.youtube.com/embed/rJN4GRoaSHs?autoplay=1"
frameborder="0" allow="autoplay; encrypted-media" allowfullscreen></iframe>
```



Phần 11: Xử lý phần Team

Xử lý HTML phần Team

BAKE TEAM



John Simon
-Cake Maker



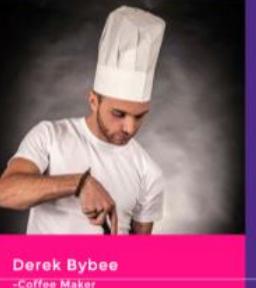
Max Will
-Coffee Maker



Rose Murphy
-Staff Manager



Barbara Will
-Cake Maker



Derek Bybee
-Coffee Maker



Lucille Webster
-Staff Manager

```
<!--TEAM-->
<section class="team">
  <h1 class="team_title ">BAKE TEAM</h1>
  <div class="team_content">
    <div class="team_member">
      
      <div class="team_member_detail">
        |   <h2>John Simon</h2><p>-Cake Maker</p>
      </div>
    </div>
    <div class="team_member">
      
      <div class="team_member_detail">
        |   <h2>Max Will</h2><p>-Coffee Maker</p>
      </div>
    </div>
    <div class="team_member">
      
      <div class="team_member_detail">
        |   <h2>Rose Murphy</h2><p>-Staff Manager</p>
      </div>
    </div>
    <div class="team_member">
      
      <div class="team_member_detail">
        |   <h2>Barbara Will</h2><p>-Cake Maker</p>
      </div>
    </div>
    <div class="team_member">
      
      <div class="team_member_detail">
        |   <h2>Derek Bybee</h2><p>-Coffee Maker</p>
      </div>
    </div>
    <div class="team_member">
      
      <div class="team_member_detail">
        |   <h2>Lucille Webster</h2><p>-Staff Manager</p>
      </div>
    </div>
  </div>
</section>
<!-- END TEAM -->
```

Xử lý CSS phần Team

```
.team{
    background-color: #4E2A84;
    padding: 4rem 15%;
}
.team__title{
    font-size: 3rem;
    color: white;
    text-align: center;
    margin-top: 0;
}
.team__content{
    display: grid;
    grid-template-columns: repeat(3, 1fr);
    grid-gap: 1.5rem;
}
.team__member img{
    width: 100%;
    display: block;
}
.team__member{
    position: relative;
}
.team__member__detail{
    position: absolute;
    bottom: 0;
    left: 0;
    width: 100%;
    background-color: #ff0e83;
    color: white;
    padding: 1rem;
}
.team__member__detail h2{
    font-size: 1rem;
    margin: 0 0 0.5rem 0;
}
.team__member__detail p {
    font-size: 0.8rem;
    margin: 0;
}
```

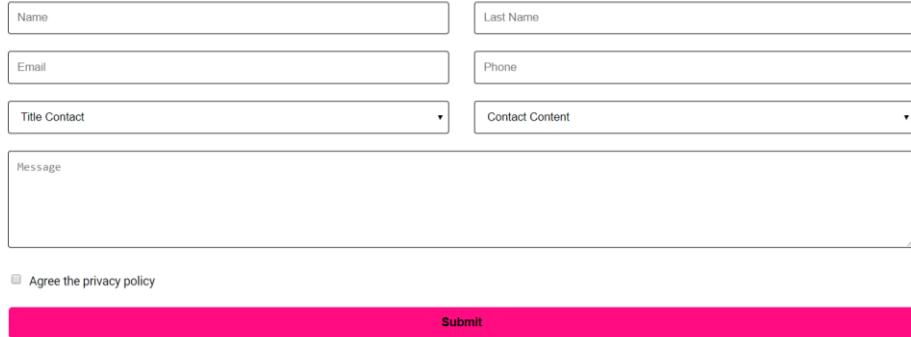
Ở phần này, ta có thể thấy, hình ảnh và phần detail nằm chồng lên nhau, do đó ta sử dụng position, chỉnh vị trí cho phần detail.Trong đó:

- Div cha cho position:relative;
- Div con là .team__member__detail ta cho position là absolute, do đó div sẽ di chuyển bên trong div cha

Phần 12: Xử lý phần Contact

Xử lý HTML phần Contact

CONTACT



The screenshot shows a contact form with the following fields:

- Name (text input)
- Last Name (text input)
- Email (text input)
- Phone (text input)
- Title Contact (dropdown menu with options: Title Contact, Reservation, Order Cakes)
- Contact Content (dropdown menu with options: Contact Content, Reservation, Order Cakes)
- Message (text area)
- Agree the privacy policy (checkbox)
- Submit button (pink button)

Trong đó:

- Nội dung form sẽ được chứa trong thẻ `<form>`
- Thẻ `<input type="text">` dùng để tạo ra các ô cho người dùng gõ text
- Thẻ `<select>` dùng để tạo ra dropdown list cho người dùng chọn
- Thẻ `<textarea>` tạo ra ô text nhiều dòng để người dùng nhập
- Thẻ `<input type="checkbox">` để tạo ra ô check, người dùng có thể tích chọn
- Placeholder: thông tin gợi ý của textbox, khi người dùng nhập text vào sẽ mất

```

<!--CONTACT-->
<section class="contact">
  <h1 class="contact__title">CONTACT</h1>
  <form action="" method="POST" class="contact__form">
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Name">
    </div>
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Last Name">
    </div>
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Email">
    </div>
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Phone">
    </div>
    <div class="form-group">
      <select class="form-control">
        <option value="">Title Contact</option>
        <option value="">Reservation</option>
        <option value="">Order Cakes</option>
      </select>
    </div>
    <div class="form-group">
      <select class="form-control">
        <option value="">Contact Content</option>
        <option value="">Reservation</option>
        <option value="">Order Cakes</option>
      </select>
    </div>
    <div class="form-group message">
      <textarea class="form-control" rows="6" placeholder="Message"></textarea>
    </div>
    <div class="form-group privacy">
      <input type="checkbox">
      <span>Agree the privacy policy</span>
    </div>
    <button class="btn-pink btn-submit" type="submit">Submit</button>
  </form>
</section>
<!-- END CONTACT -->

```

Tài sử dụng class `btn-pink` đã có sẵn

Xử lý CSS phần Contact



CONTACT

Name

Last Name

Email

Phone

Title Contact

Contact Content

Message

Agree the privacy policy

Submit

```
.contact{  
    padding:4rem 15%;  
}  
.contact__title{  
    font-size:3rem;  
    color:#black;  
    text-align: center;  
    margin-top:0;  
}  
.contact__form{  
    display: grid;  
    grid-template-columns: repeat(2,1fr);  
    grid-column-gap: 1.5rem;  
    grid-row-gap: 1rem;  
}  
.form-control{  
    width: 100%;  
    border:1px solid #black;  
    border-radius: 3px;  
    padding:0.5rem;  
}  
.message{  
    grid-column: 1/span 2;  
}  
.privacy span{  
    font-size:0.7rem;  
}  
.btn-submit{  
    grid-column: 1/span 2;  
    color:#black;  
    font-size:0.7rem;  
    font-weight: 700;  
}
```

Phần 13: Review (Form)

GIỚI THIỆU VỀ FORM

- Được dùng để nhận dữ liệu từ phía người dùng
- Giúp gởi yêu cầu của người dùng đến trang xử lý trong ứng dụng web
- Tag **<form>** dùng để chứa các thành phần khác của form
- Những thành phần nhập liệu được gọi là form field:
 - Text field
 - Password field
 - Multi-line text field

Thẻ <form></form>

- Là container chứa các thành phần nhập liệu khác

```
<form name="" action="" method="">
    <!-- Các thành phần của form --&gt;
&lt;/form&gt;</pre>
```

- Các thuộc tính của <form>:
 - name: tên form
 - action: chỉ định trang web nhận xử lý dữ liệu từ form này khi có sự kiện click của button submit
 - target: chỉ định nơi hiển thị kết quả
 - method: xác định phương thức chuyển dữ liệu (POST, GET)

PHƯƠNG THỨC GET/POST

- Get/Post là giá trị của thuộc tính method trong thẻ <form>
- Quy định phương thức HTTP khi submit form
- GET or POST do backend quy định

PHƯƠNG THỨC GET

- Các đối số của Form được chèn vào đường dẫn URL của thuộc tính action trong thẻ `<form action="trangchu.php"></form>`
- Lượng dữ liệu đối số truyền đi của Form bị giới hạn bởi chiều dài tối đa của một URL trên Address bar (khoảng 3.000 ký tự)
- Không dung GET cho các dữ liệu “nhạy cảm” (password, mã pin,...)
- Thuận tiện trong trường hợp người dùng muốn bookmark lại kết quả
- Thích hợp cho dữ liệu không cần bảo mật (ví dụ input search google)

PHƯƠNG THỨC GET

```
<form action="/action_page.php" method="GET">
  First name:<br>
  <input type="text" name="firstname" value="Mickey">
  <br>
  Last name:<br>
  <input type="text" name="lastname" value="Mouse">
  <br><br>
  <input type="submit" value="Submit">
</form>
```

ⓘ file:///D:/action_page.php?firstname=Mickey&lastname=Mouse

First name:

Mickey

Last name:

Mouse

Submit

Thông tin người dùng nhập hiển thị trên url

PHƯƠNG THỨC POST

- Thông tin không được chèn thêm vào URL
- Sử dụng post khi form chứa thông tin nhạy cảm hoặc thông tin cá nhân
- Phương thức post KHÔNG có giới hạn về size → sử dụng được khi có lượng lớn dữ liệu
- Form submission có phương thức post thì KHÔNG thể bookmark

PHƯƠNG THỨC POST

```
<form action="/action_page.php" method="POST">
    First name:<br>
    <input type="text" name="firstname" value="Mickey">
    <br>
    Last name:<br>
    <input type="text" name="lastname" value="Mouse">
    <br><br>
    <input type="submit" value="Submit">
</form>
```

① file:///D:/action_page.php

Thông tin người dùng nhập không hiển thị trên url

First name:

Mickey

Last name:

Mouse

Submit

Các thành phần của form

- Text field, Password field, Hidden Text field
- Checkbox
- Radio Button
- File Form Control
- Submit Button, Reset Button, Generalized Button
- Multiple-line text field
- Label
- Pull-down menu
- Scrolled list

THUỘC TÍNH ENCTYPE

- Quy định dữ liệu có được mã hóa hoặc mã hóa như thế nào trước khi được gửi đến server

Giá trị	Mô tả
application/x-www-form-urlencoded	Mặc định. Tất cả các ký tự được mã hóa trước khi gửi. (Khoảng trống được mã hóa thành "+", ký tự đặc biệt được chuyển thành mã ASCII HEX)
multipart/form-data	Không có ký tự nào được mã hóa. Sử dụng trong trường hợp file upload control
text/plain	Khoảng trống được mã hóa thành "+", ký tự đặc biệt không mã hóa

<input>

- Các thuộc tính:
 - Type: quy định loại kiểu nhập liệu (text, password, color, radio,...)
 - Name: tên của input, nếu thiếu thuộc tính này, dữ liệu sẽ KHÔNG được gửi lên server
 - Readonly: Boolean. Nếu true → không cho phép nhập liệu
 - Size: kích thước của khung nhập liệu
 - Maxlength: chiều dài tối đa chia phép nhập liệu
 - Tabindex: Thứ tự focus khi nhấn nút tab
 - Value: giá trị trong ô input
- https://www.w3schools.com/tags/tag_input.asp

<input>

- Ví dụ về thẻ input

```
<form name="Login"
      action="/admin/LoginProcess.php"
      method="Post">
    <label>Tên đăng nhập: </label>
    <input type="text" name="TenDangNhap">
</form>
```

Tên đăng nhập:

<input>

- Thuộc tính value dùng để set giá trị mặc định cho ô input, sẽ tự hiển thị khi load trang

```
<form action="/action_page.php" method="GET">
  First name:<br>
  <input type="text" name="firstname" value="Mickey">
  <br>
  Last name:<br>
  <input type="text" value="Mouse">
  <br><br>
  <input type="submit" value="Submit">
</form>
```

ⓘ file:///D:/action_page.php?firstname=Mickey

First name:

Mickey

Last name:

Mouse

Submit

<input>

- Thuộc tính **readonly** : chỉ cho phép người dùng nhìn, không được phép nhập
- Maxlength: độ dài ký tự tối đa người dùng có thể nhập vào ô
- Tab-index: thứ tự khi ta nhấn nút “tab”
- Input với type là “**password**”: mã hóa ký tự gõ vào

```
<form name="Login"
      action="/admin/LoginProcess.php"
      method="Post">
    <span>Email: </span>
    <input type="email" name="Email" size="20" maxlength="20" value="phonghiavan@gmail.com" readonly><br>

    <span>Tên đăng nhập</span>
    <input type="text" name="TenDangNhap" size="20" maxlength="20" value="Nghĩa Văn" tabindex="1"><br>

    <span>Quê quán</span>
    <input type="text" name="QueQuan" size="20" maxlength="20" tabindex="3"><br>

    <span>Mã pin</span>
    <input type="password" name="TenDangNhap" size="10" maxlength="4" tabindex="2"><br>

    <span>Không thể nhìn thấy</span>
    <input type="hidden" name="CantSee" size="10" maxlength="10" tabindex="4">
</form>
```

Email:

Tên đăng nhập

Quê quán

Mã pin

Không thể nhìn thấy

Checkbox

- Có thể check nhiều ô

```
<form name="Login"
      action="/admin/LoginProcess.php"
      method="Post">
    <span>Sở thích:</span><br>
    <input type="checkbox" name="sothich" value="1"><span>Đọc sách</span><br>
    <input type="checkbox" name="sothich" value="2"><span>Tập thể dục</span><br>
    <input type="checkbox" name="sothich" value="3"><span>Chơi game</span><br>
    <input type="checkbox" name="sothich" value="4"><span>Code</span><br>
    <input type="checkbox" name="sothich" value="5"><span>Khác</span>
</form>
```

Sở thích:

- Đọc sách
- Tập thể dục
- Chơi game
- Code
- Khác

Radio

- Chỉ Có thể check 1 ô (cùng name)

```
<form name="Login"
      action="/admin/LoginProcess.php"
      method="Post">
    <span>Nghề nghiệp:</span><br>
    <input type="radio" name="nghe" value="dev"><span>Lập trình viên</span><br>
    <input type="radio" name="nghe" value="sci"><span>Khoa học dữ liệu</span><br>
    <input type="radio" name="nghe" value="lec"><span>Giảng viên</span><br>
    <input type="radio" name="nghe" value="sts"><span>Sinh viên</span>
    <input type="radio" name="nghe" value="etc"><span>Khác</span>
</form>
```

Nghề nghiệp:

- Lập trình viên
- Khoa học dữ liệu
- Giảng viên
- Sinh viên
- Khác

<input type="button">

```
<form name="Login"
      action="/admin/LoginProcess.php"
      method="Post">
    <input type="button" name="DangNhap" value="Đăng nhập">
</form>
```

Đăng nhập

<label>

- Định nghĩa label cho <input>
- Có thuộc tính for="..." tương đương với thuộc tính id="..." của <input>
- Khi user click vào label → input tương ứng sẽ được focus

```
<form action="/action_page.php">
  <label for="male">Male</label>
  <input type="radio" name="gender" id="male" value="male"><br>
  <label for="female">Female</label>
  <input type="radio" name="gender" id="female" value="female"><br>
  <label for="other">Other</label>
  <input type="radio" name="gender" id="other" value="other"><br><br>
  <input type="submit" value="Submit">
</form>
```

Male
Female
Other

Submit

FILE UPLOAD CONTROL

- Dùng để upload 1 file lên server

```
<form method="post" enctype="multipart/form-data">
    <input type="file" name="">
</form>
```

Choose File No file chosen

SUBMIT BUTTON

- Nút phát lệnh và gửi dữ liệu của form đến trang xử lý
- Mỗi form chỉ có một nút submit
- Mỗi form có thể có nhiều nút button (ko phải submit)

```
<form method="post" action="trangchu.php">
    <input type="submit" name="" value="Submit">
</form>
```

Submit

RESET BUTTON

- Trả về giá trị mặc định cho các field

```
<form method="post" action="trangchu.php">
    <label>Họ tên: </label><input type="text" name="" value="Nguyễn Văn A">
    <label>SĐT: </label><input type="number" name="" placeholder="Nhập SĐT">
    <label>Email: </label><input type="email" name="">
    <input type="reset" name="" value="Reset">
</form>
```

Họ tên: Nghĩa Văn

SĐT: 0963

Email: phonghiavan@gmail.com

Reset



Họ tên: Nguyễn Văn A

SĐT: Nhập SĐT

Email:

Reset

Textarea

- Dùng để nhập văn bản nhiều dòng
- Textarea có các thuộc tính: cols, rows, disabled, name, readonly, tabindex

```
<form method="post" action="trangchu.php">
    <textarea cols="40" rows="10"></textarea>
</form>
```

COMBOBOX

- Tạo ra một danh sách, cho phép chọn
- Các thuộc tính: autofocus, disabled, multiple, size, name, required

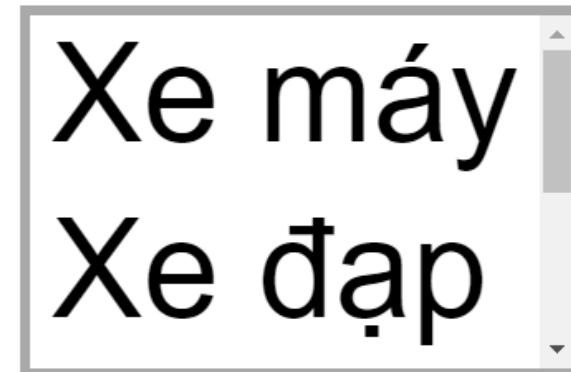
```
<form method="post" action="trangchu.php">
  <select>
    <option value="xeMay">Xe máy</option>
    <option value="xeDap">Xe đạp</option>
    <option value="xeHoi">Xe hơi</option>
    <option value="dep">Dép</option>
  </select>
</form>
```



COMBOBOX

- Set multiple và size: cho phép chọn nhiều option cùng lúc

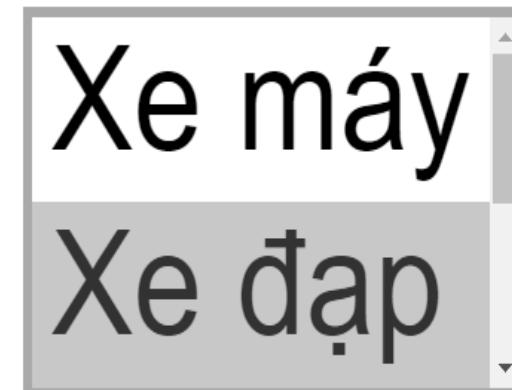
```
<form method="post" action="trangchu.php">
  <select multiple size="2">
    <option value="xeMay">Xe máy</option>
    <option value="xeDap">Xe đạp</option>
    <option value="xeHoi">Xe hơi</option>
    <option value="dep">Dép</option>
  </select>
</form>
```



COMBOBOX

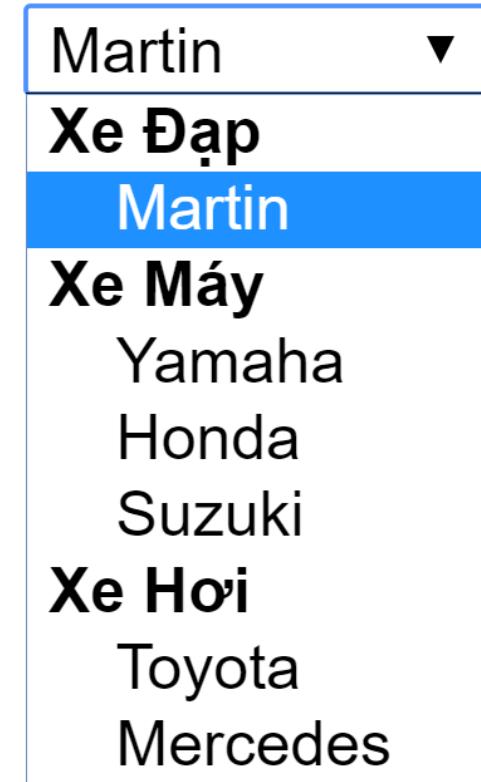
- Khi thêm thuộc tính selected vào <option>: định nghĩa option đang được chọn

```
<form method="post" action="trangchu.php">
    <select multiple size="2">
        <option value="xeMay">Xe máy</option>
        <option value="xeDap" selected>Xe đạp</option>
        <option value="xeHoi">Xe hơi</option>
        <option value="dep">Dép</option>
    </select>
</form>
```



COMBOBOX

```
<form method="post" action="trangchu.php">
<select>
    <optgroup label="Xe Đạp">
        <option value="Martin">Martin</option>
    </optgroup>
    <optgroup label="Xe Máy">
        <option value="Yamaha">Yamaha</option>
        <option value="Honda">Honda</option>
        <option value="Suzuki">Suzuki</option>
    </optgroup>
    <optgroup label="Xe Hơi">
        <option value="Toyota">Toyota</option>
        <option value="Mercedes">Mercedes</option>
    </optgroup>
</select>
</form>
```



<fieldset> và <legend>

- <fieldset>: nhóm những trường liên quan với nhau
- <legend>: tạo caption cho những nhóm được tạo ra từ <field><set>

```
<form action="/action_page.php">
<fieldset>
  <legend>Personal information:</legend>
  First name:<br>
  <input type="text" name="firstname" value="Mickey"><br>
  Last name:<br>
  <input type="text" name="lastname" value="Mouse"><br><br>
  <input type="submit" value="Submit">
</fieldset>
</form>
```

Personal information:

First name:

Mickey

Last name:

Mouse

Submit

Phân 14: Review (Form HTML 5)

NEW TYPE

- HTML5 cung cấp nhiều types cho input: **email**, **date**, range, color,... tùy theo mục đích sử dụng
- Các types có kèm theo validation

```
<form>
  <input type="email">
  (<input type=tel size=3>) <input type=tel size=3> - <input type=tel size=4>
  <input type=date step=7 min=2014-09-08>
  <input type=color>
  <input type=range min=10 max=30 value=10 step=1>
  <button type="submit"></button>
</form>
```

PLACEHOLDER

- Thuộc tính placeholder là thuộc tính mới trong html5
- Tạo phần chữ chìm trong input
- Set thuộc tính CSS cho placeholder → input::placeholder

```
<title></title>
<style type="text/css">
    input::placeholder{
        color: blue;
    }
</style>
</head>
<body>
    <label>Họ và tên: </label><input type="text" name="" placeholder="Điền vào tên">
</body>
```

AUTOFOCUS

- Con trỏ chuột tự động nằm ở khung input mà ko cần trỏ chuột vào

```
<label>Điền Email: </label><input type="email" autofocus>
```

MIN & MAX

- Set GTNN và GTLN cho input

```
<form action="/action_page.php">
  Enter a date before 1980-01-01:
  <input type="date" name="bday" max="1979-12-31"><br>

  Enter a date after 2000-01-01:
  <input type="date" name="bday" min="2000-01-02"><br>

  Quantity (between 1 and 5):
  <input type="number" name="quantity" min="1" max="5"><br>

  <input type="submit">
</form>
```

Phần 15: GRAPHIC ELEMENTS

<canvas>

- Vẽ hình ảnh thông qua JavaScript.
- Hình ảnh: đường thẳng, hình tròn, hình hộp, chữ, hình ảnh,...

```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;"></canvas>
<script type="text/javascript">
    c = document.getElementById("myCanvas");
    ctx = c.getContext("2d");
    ctx.moveTo(0,0);
    ctx.lineTo(200,100);
    ctx.stroke();
</script>
```

<SVG>

- SVG viết tắt của Scalable Vector Graphics
- Vẽ hình (không thông qua javascript)

```
<svg width="400px" height="300px" style="border:2px solid #666">
    <rect x="100px" y="50px" rx="10px" ry="10px" width="150px" height="150px"
        style="fill:red;stroke:black;stroke-width:4px"/>
</svg>

<svg width="400px" height="300px">
    <circle cx="200px" cy="150px" r="80px" style="fill:red"/>
    <text fill="#FFF" font-size="70px" font-family="Times New Roman" font-weight=
        "Bold" x="150px" y="172px">GF</text>
</svg>
```

Phần 16: Lý thuyết về CSS3

CSS3 khác gì so với CSS ?

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

Các thuộc tính của css3 được sử dụng:



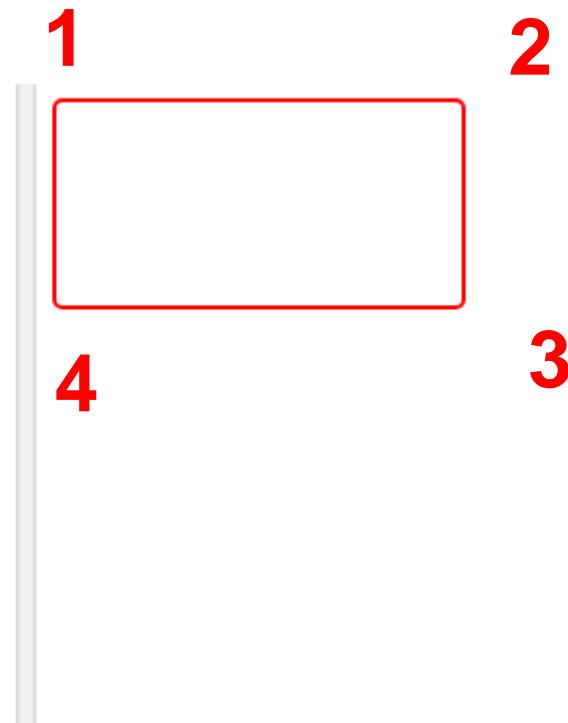
- 1 Border-radius
- 2 Background-size
- 3 Linear-gradient
- 4 Text-shadow
- 5 Box-shadow
- 6 Before,after
- 7 CSS variables
- 8 @font-face

Các thuộc tính mới 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>
```



❖ Border-radius

- ✓ 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]

Property					
border-radius	5.0 4.0 -webkit-	9.0	4.0 3.0 -moz-	5.0 3.1 -webkit-	10.5

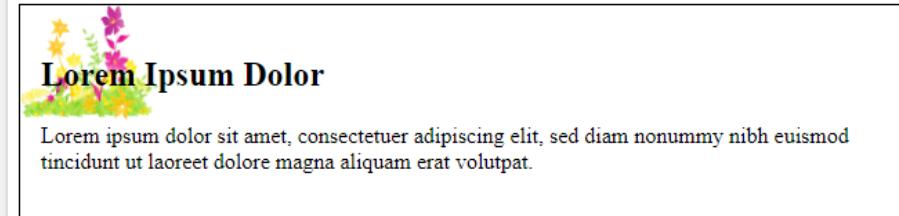
❖ CSS3 background-size

- Theo px ,rem hoặc em
- Theo %
- Contain, cover ,auto

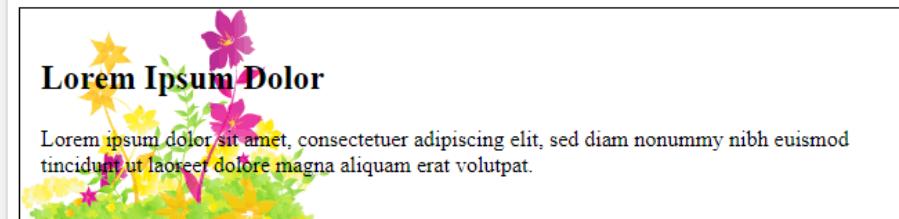
```
<!DOCTYPE html>
<html>
<head>
<style>
#example1 {
    border: 1px solid black;
    background:url(img_flwr.gif);
    background-size: 100px 80px;
    background-repeat: no-repeat;
    padding:15px;
}

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

Điều chỉnh kích thước background:



Mặc định



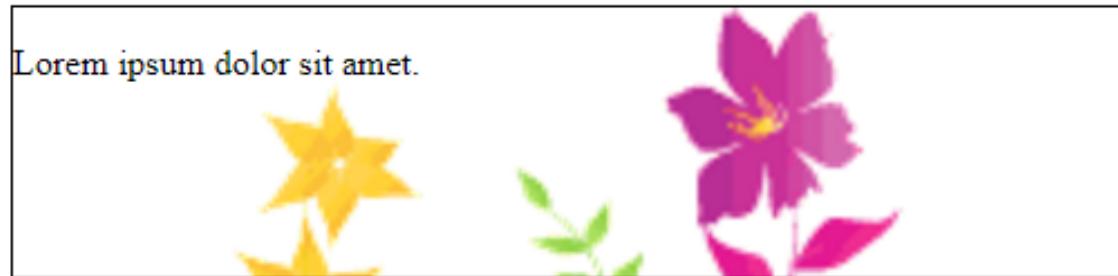
❖ CSS3 background-size

Phân biệt cover và contain

background-size: contain:



background-size: cover:



❖ CSS3 background-size

- **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

❖ Color : RGBA

Giá trị màu RGBA là một phần mở rộng của các giá trị màu RGB với 1 kênh alpha – xác định độ trong suốt của màu

```
rgba(255, 0, 0, 0.2);
```

```
rgba(255, 0, 0, 0.4);
```

```
rgba(255, 0, 0, 0.6);
```

```
rgba(255, 0, 0, 0.8);
```

```
<!DOCTYPE html>
<html>
<head>
<style>
#p1 {background-color:rgba(255,0,0,0.3);}
#p2 {background-color:rgba(0,255,0,0.3);}
#p3 {background-color:rgba(0,0,255,0.3);}
#p4 {background-color:rgba(192,192,192,0.3);}
#p5 {background-color:rgba(255,255,0,0.3);}
#p6 {background-color:rgba(255,0,255,0.3);}
</style>
</head>
<body>

<h1>Define Colors With RGBA Values</h1>

<p id="p1">Red</p>
<p id="p2">Green</p>
<p id="p3">Blue</p>
<p id="p4">Grey</p>
<p id="p5">Yellow</p>
<p id="p6">Cerise</p>
```

Define Colors With RGBA Values

Red

Green

Blue

Grey

Yellow

Cerise

❖ CSS3 Linear Gradients

CSS3 hỗ trợ 2 loại màu Gradients :

- Linear Gradients : trải màu theo dưới/trên/trái/phải/chéo
- Radial Gradients: trải từ tâm ra.

Property					
linear-gradient	26.0 10.0 -webkit-	10.0	16.0 3.6 -moz-	6.1 5.1 -webkit-	12.1 11.1 -o-
radial-gradient	26.0 10.0 -webkit-	10.0	16.0 3.6 -moz-	6.1 5.1 -webkit-	12.1 11.6 -o-
repeating-linear-gradient	26.0 10.0 -webkit-	10.0	16.0 3.6 -moz-	6.1 5.1 -webkit-	12.1 11.1 -o-
repeating-radial-gradient	26.0 10.0 -webkit-	10.0	16.0 3.6 -moz-	6.1 5.1 -webkit-	12.1 11.6 -o-

❖ CSS3 Linear Gradients

Cú pháp: `background: linear-gradient(direction, color-stop1, color-stop2, ...);`

- **Linear Gradient : Trải từ Top đến bottom (Mặc định)**

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
    height: 200px;
    background: red; /* For browsers that do not support gradients */
    background: -webkit-linear-gradient(red, yellow);
    background: -moz-linear-gradient(red, yellow);
    background: -o-linear-gradient(red, yellow);
    background: linear-gradient(red, yellow); /* Standard syntax (must be
last) */
}
</style>
</head>
<body>

<h1>Linear Gradient - Top to Bottom</h1>
<p>This linear gradient starts at the top. It starts red, transitioning to
yellow at the bottom.</p>

</body>
</html>
```

Linear Gradient - Top to Bottom

This linear gradient starts at the top. It starts red, transitioning to yellow:



Note: Internet Explorer 9 and earlier versions do not support gradients.

❖ CSS3 Linear Gradients

Cú pháp: **background: linear-gradient(*direction*, *color-stop1*, *color-stop2*, ...);**

- **Linear Gradient : Trải từ left sang right (Mặc định)**

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
    height: 200px;
    background: red; /* For browsers that do not support gradients */
    background: -webkit-linear-gradient(to right, red , yellow);
    background: -moz-linear-gradient(to right, red , yellow);
    background: -o-linear-gradient(to right, red , yellow);
    background: linear-gradient(to right, red , yellow); /* Standard syntax
(must be last) */
}
</style>
</head>
<body>

<h1>Linear Gradient - Left to Right</h1>
<p>This linear gradient starts at the left. It starts red, transitioning to


```

Linear Gradient - Left to Right

This linear gradient starts at the left. It starts red, transitioning to yellow:



Note: Internet Explorer 9 and earlier versions do not support gradients.

❖ CSS3 Linear Gradients

Cú pháp: **background: linear-gradient(*direction*, *color-stop1*, *color-stop2*, ...);**

- **Linear Gradient : Trải từ Top đến bottom (Mặc định)**

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
    height: 200px;
    background: red; /* For browsers that do not support gradients */
    background: -webkit-linear-gradient(red, yellow);
    background: -moz-linear-gradient(red, yellow);
    background: -o-linear-gradient(red, yellow);
    background: linear-gradient(red, yellow); /* Standard syntax (must be last) */
}
</style>
</head>
<body>

<h1>Linear Gradient - Top to Bottom</h1>
<p>This linear gradient starts at the top. It starts red, transitioning to yellow.
-->
```

Linear Gradient - Top to Bottom

This linear gradient starts at the top. It starts red, transitioning to yellow:



Note: Internet Explorer 9 and earlier versions do not support gradients.

❖ CSS3 Linear Gradients

Cú pháp: **background: linear-gradient(*direction*, *color-stop1*, *color-stop2*, ...);**

- **Linear Gradient : Trải theo đường chéo**

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
    height: 200px;
    background: red; /* For browsers that do not support gradients */
    background: linear-gradient(to bottom right, red, yellow); /* Standard
syntax (must be last) */
}
</style>
</head>
<body>

<h1>Linear Gradient - Diagonal</h1>
<p>This linear gradient starts at top left. It starts red, transitioning to
yellow:</p>

<div id="grad1"></div>
```

Linear Gradient - Diagonal

This linear gradient starts at top left. It starts red, transitioning to yellow:



Note: Internet Explorer 9 and earlier versions do not support gradients.

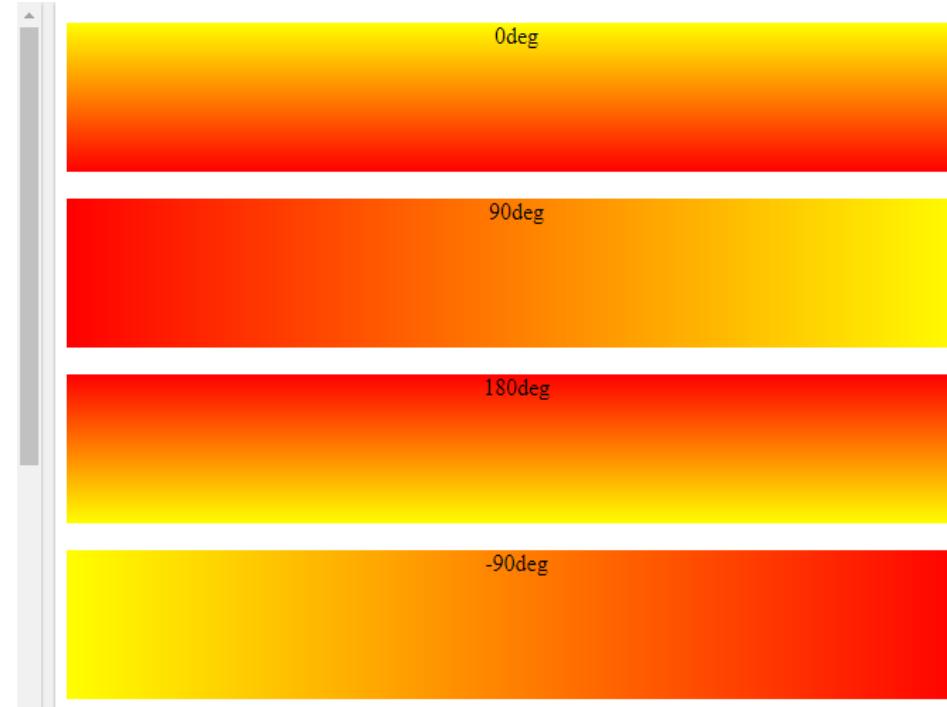
❖ CSS3 Linear Gradients

Cú pháp: **background: linear-gradient(*direction*, *color-stop1*, *color-stop2*, ...);**

- **Linear Gradient : sử dụng degree**

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
    height: 100px;
    background: red; /* For browsers that do not support gradients */
    background: linear-gradient(0deg, red, yellow);
}
#grad2 {
    height: 100px;
    background: red; /* For browsers that do not support gradients */
    background: linear-gradient(90deg, red, yellow);
}
#grad3 {
    height: 100px;
    background: red; /* For browsers that do not support gradients */
    background: linear-gradient(180deg, red, yellow);
}

#grad4 {
    height: 100px;
    background: red; /* For browsers that do not support gradients */
    background: linear-gradient(-90deg, red, yellow); /* Standard syntax
(must be last) */
|
```



❖ CSS3 Linear Gradients

Cú pháp: **background: linear-gradient(*direction*, *color-stop1*, *color-stop2*, ...);**

```
<!DOCTYPE html>| Change Orientation
<html>
<head>
<style>
#grad1 {
    height: 150px;
    background: red; /* For browsers that do not support gradients */
    background: linear-gradient(to right, red, orange, yellow, green, blue,
indigo, violet); /* Standard syntax (must be last) */
}
</style>
</head>
<body>
```



Note: Internet Explorer 9 and earlier versions do not support gradients.

❖ CSS3 Text Shadow



- Vertical + Horizontal Shadow

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
    text-shadow: 2px 2px;
}
</style>
</head>
<body>
```

Text-shadow effect!

Note: Internet Explorer 9 and earlier versions, do not support the text-shadow property.

- Vertical + Horizontal + color

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
    text-shadow: 2px 2px red;
}
</style>
</head>
<body>

<h1>Text-shadow effect!</h1>

<p><b>Note:</b> Internet Explorer 9 and earlier versions, do not support the text-shadow property.</p>
```

Text-shadow effect!

Note: Internet Explorer 9 and earlier versions, do not support the text-shadow property.

CSS3 Text Shadow



- Vertical + Horizontal + blur + color Shadow

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
    text-shadow: 2px 2px 5px red;
}
</style>
</head>
<body>

<h1>Text-shadow effect!</h1>

<p><b>Note:</b> Internet Explorer 9 and earlier versions, do not support the
```

Text-shadow effect!

Note: Internet Explorer 9 and earlier versions, do not support the text-shadow property.

CSS3 Box Shadow

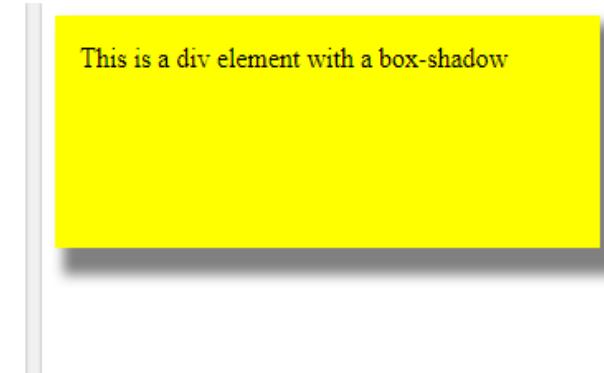


- Thuộc tính giúp làm nổi một khối các phần tử

Box-shadow:[val1] [val2] [val3] [val4] [color];

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
    width: 300px;
    height: 100px;
    padding: 15px;
    background-color: yellow;
    box-shadow: 10px 10px 10px 5px grey;
}
</style>

```



Property	Chrome	Edge	Firefox	Safari	Opera
box-shadow	10.0 4.0 -webkit-	9.0	4.0 3.5 -moz-	5.1 3.1 -webkit-	10.5

CSS3 @font-face

- Cài đặt font chữ vào file từ CSS

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

* {
    font-family: myFirstFont;
}
</style>
```

CSS3 Resizing (box-sizing)

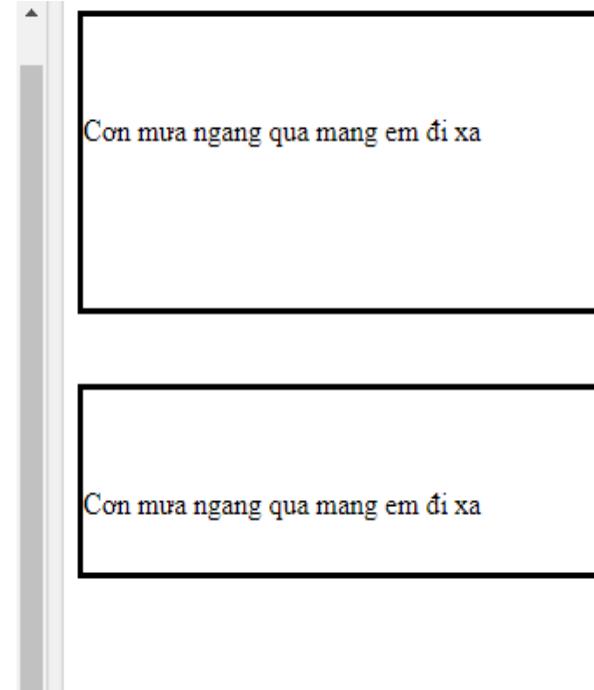
- Tương tự thuộc tính border nhưng đúng hơn về mặt thực tế về kích thước dài rộng
 - ❖ Cách tính của border(không có border-sizing):
 $\text{width} + \text{padding} + \text{border} = \text{chiều rộng của phần tử}$
 $\text{height} + \text{padding} + \text{border} = \text{chiều cao của phần tử}$
 - ❖ Cách tính của border(có border-sizing):
Vẫn lấy width height xét kích thước, nhưng nếu có padding thì vẫn padding mà không cộng thêm vào.

CSS3 Resizing (box-sizing)



- Tương tự thuộc tính border nhưng đúng hơn về mặt thực tế về kích thước dài rộng

```
<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>
```

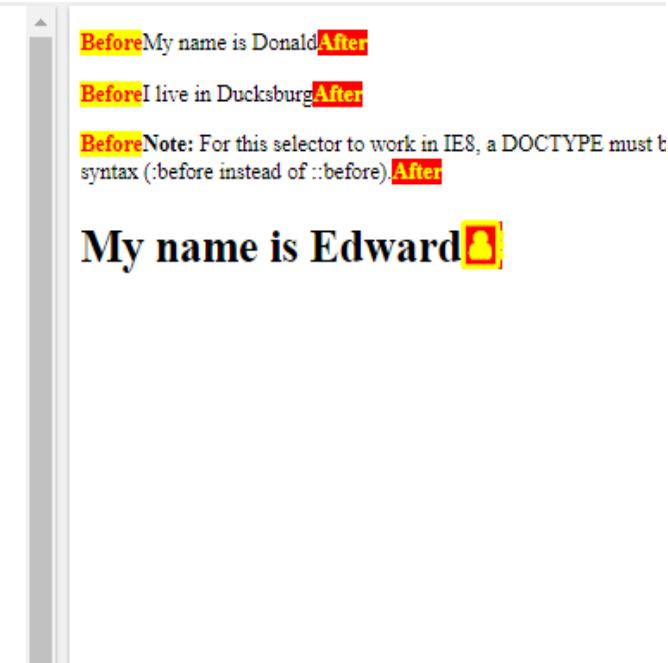


Before và After



- Cho phép chèn nội dung vào trước hoặc sau cái phần tử HTML.
- Thường được dùng với font awesome

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
<style>
p::before {
  content: "Before";
  background-color: yellow;
  color: red;
  font-weight: bold;
}
p::after{
  content:'After';
  background-color:red;
  color:yellow;
  font-weight:bold;
}
h1::after{
  font-family: FontAwesome;
  content: '\f2ba';
  background-color:red;
  color:yellow;
  font-weight:bold;
}
...
}
```



Before và After

- Sử dụng before hoặc after để tạo mũi tên

```
<style type="text/css">
    .demo{
        background-color:blue;
        width:300px;
        height:300px;
        position:relative;
        top:30px;
    }
    .demo::before{
        content:'';
        border-bottom:15px solid blue;
        border-left:15px solid transparent;
        border-right:15px solid transparent;
        position:absolute;
        top:-15px;
        left:40%;
    }
</style>
<body>

    <div class="demo"></div>

</body>
```



CSS Variable



- Tương tự như các ngôn ngữ lập trình khác, ta đặt và sử dụng biến thay thế cho giá trị của thuộc tính
- Tiết kiệm thời gian chỉnh sửa css

```
<style type="text/css">
  :root{
    --color-red:#f00;
    --font-size:60px;
  }
  h1{
    color:var(--color-red);
    font-size:var(--font-size);
  }
</style>
<body>
  <h1>CyberSoft Academy</h1>
</body>
</html>
```

Khu vực khai báo biến:
--Tên biến : giá trị

Cách sử dụng biến :
var(--Tên biến)

CyberSoft Academy

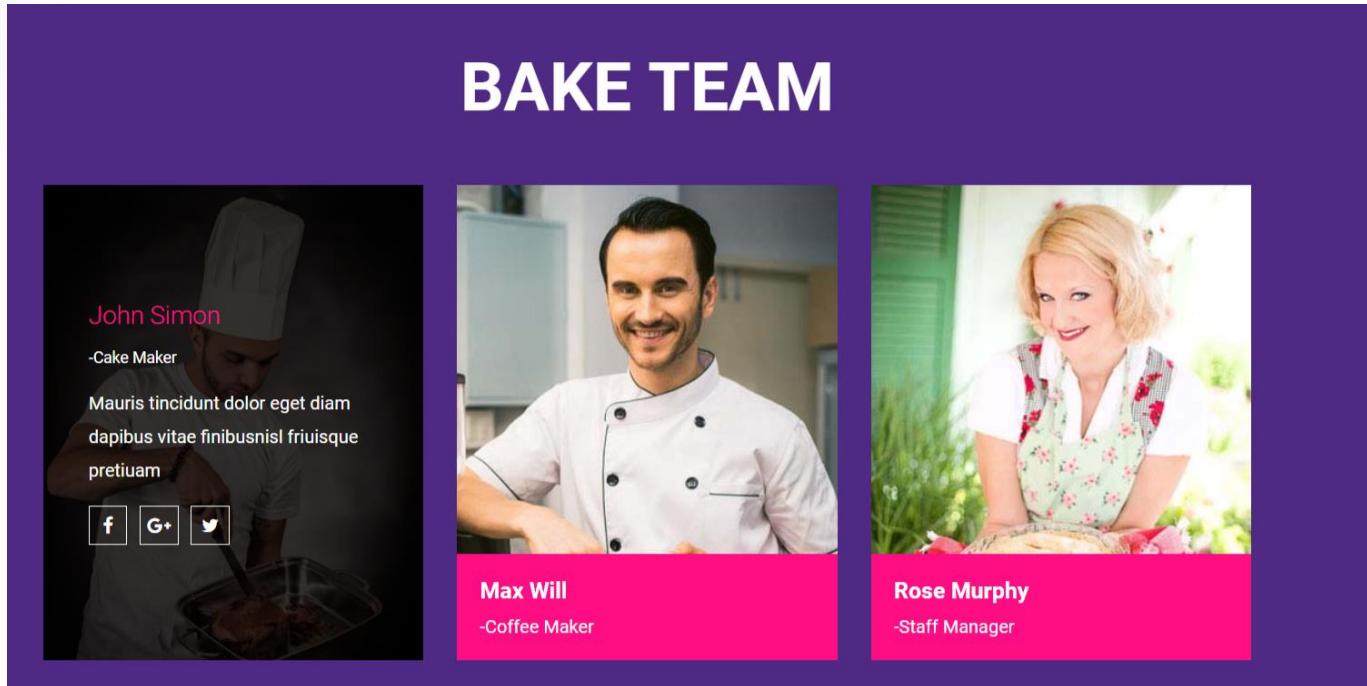
PHẦN 17: CSS3 TRANSFORM



CyberSoft Academy - CyberSoft.edu.vn - MyClass.vn

- ❖ CSS3 cung cấp cho ta một số thuộc tính để tạo hiệu ứng, làm cho trang web thêm sinh động
- ❖ Bây giờ ta sẽ áp dụng các animation vào baked layout.

CSS3 Transform



❖ Yêu cầu ở phần Team:

- Khi hover chuột vào div .team_member sẽ ẩn đi div .team_member_detail và hiển thị ra nội dung như thế này.

CSS3 Transform



- ❖ Bước 1: ở HTML, them vào đoạn code đê tạo ra div màu đen như hình, thêm 1 class vào để chỉnh sửa riêng cho div này

```
<div class="team_member team_cake_maker">
  
  <div class="team_member_detail">
    <h2>John Simon</h2><p>-Cake Maker</p>
  </div>
  <div class="overlay">
    <div class="overlay_content">
      <h2>John Simon</h2>
      <span>-Cake Maker</span>
      <p>Mauris tincidunt dolor eget diam dapibus vitae finibus nisl fruiusque pretium</p>
      <i class="fa fa-facebook"></i>
      <i class="fa fa-google-plus"></i>
      <i class="fa fa-twitter"></i>
    </div>
  </div>
```

CSS3 Transform

- ❖ Bước 2: ở CSS, ta chỉnh sửa như sau để được định dạng như hình



```
.overlay{  
background-color:rgba(0,0,0,0.8);  
color:white;  
width:100%;  
height: 100%;  
position: absolute;  
top:0;  
left:0;  
padding:0 2rem;  
display: flex;  
align-items: center;  
transition:all 0.8s;  
}  
.overlay h2{  
font-size:1.1rem;  
font-weight: 300;  
color:#FF0E83;  
margin:0;  
margin-bottom:0.5rem;  
}  
.overlay span{  
font-size:0.7rem;  
}  
.overlay p{  
font-size:0.8rem;  
line-height: 1.5rem;  
}  
.overlay i{  
width: 1.7rem;  
height: 1.7rem;  
border:1px solid white;  
font-size:0.8rem;  
text-align: center;  
line-height: 1.7rem;  
margin-right:0.3rem;  
}
```

CSS3 Transform

❖ Bước 3: Tạo hiệu ứng ẩn hiện khi hover vào div .team_member



```
.overlay{
    transition:all 0.8s;           (4)
}

.team_member:hover .team_member_detail{
    display: none;                (2)
}

.team_cake_maker .overlay{
    opacity: 0;                   (1)
}

.team_cake_maker:hover .overlay{
    opacity:1;                    (3)
}
```

- (1) Đầu tiên, ta cho div .overlay ẩn đi với thuộc tính opacity
- (2) Khi hover vào div .team_member, ta gọi ra class .team_member_detail và ẩn đi
- (3) Khi hover vào div cha là .team_cake_maker, ta gọi overlay ra và hiển thị nó lên.
- (4) Để thấy được quá trình overlay ẩn hiện, ta gắn cho nó thuộc tính transition

CSS3 Transform

❖ Bước 4: Tạo hiệu ứng trượt khi hover vào div .team_member



```
.team_member{  
    overflow: hidden;          (3)  
}  
.team_coffee_maker .overlay{  
    transform:translateY(-100%); (1)  
.team_coffee_maker:hover .overlay{  
    transform: translateY(0);    (2)  
}
```

- (1) Đầu tiên ta cho .overlay nằm bên ngoài bằng thuộc tính transform với hàm translateY() : di chuyển phần tử theo chiều dọc
- (2) Khi hover vào div cha, gọi overlay và translate lại về vị trí ban đầu
- (3) Ở team_member thêm thuộc tính overflow:hidden, những phần tử con nào bị tràn ra ngoài thì sẽ bị ẩn đi

CSS3 Transform

❖ Bước 5: Tạo hiệu ứng scale khi hover vào div .team_member



```
.team_staff_manager .overlay{  
    transform: scale(0);          (1)  
}  
.team_staff_manager:hover .overlay{  
    transform: scale(1,1);        (2)  
}
```

- (1) Đầu tiên ta cho .overlay mât đi với thuộc transform: scale(0), nghĩa là thu nhỏ phần tử lại về mức 0, lấy chiều cao * 0 và chiều rộng * 0 ra được kích thước mờ
- (2) Khi hover vào div cha, cho div overlay hiển thị lại cũng thông qua hàm scale(1,1) -> lấy chiều cao * 1, chiều rộng * 1 ra kích thước ban đầu

REVIEW TRANFORM

❖ CSS3 Transforms

- Cho phép di chuyển, xoay , phóng to, thu nhỏ, làm nghiêng các phần tử trong trang web.
- CSS3 hỗ trợ transform 2D và 3D
- Trình duyệt hỗ trợ

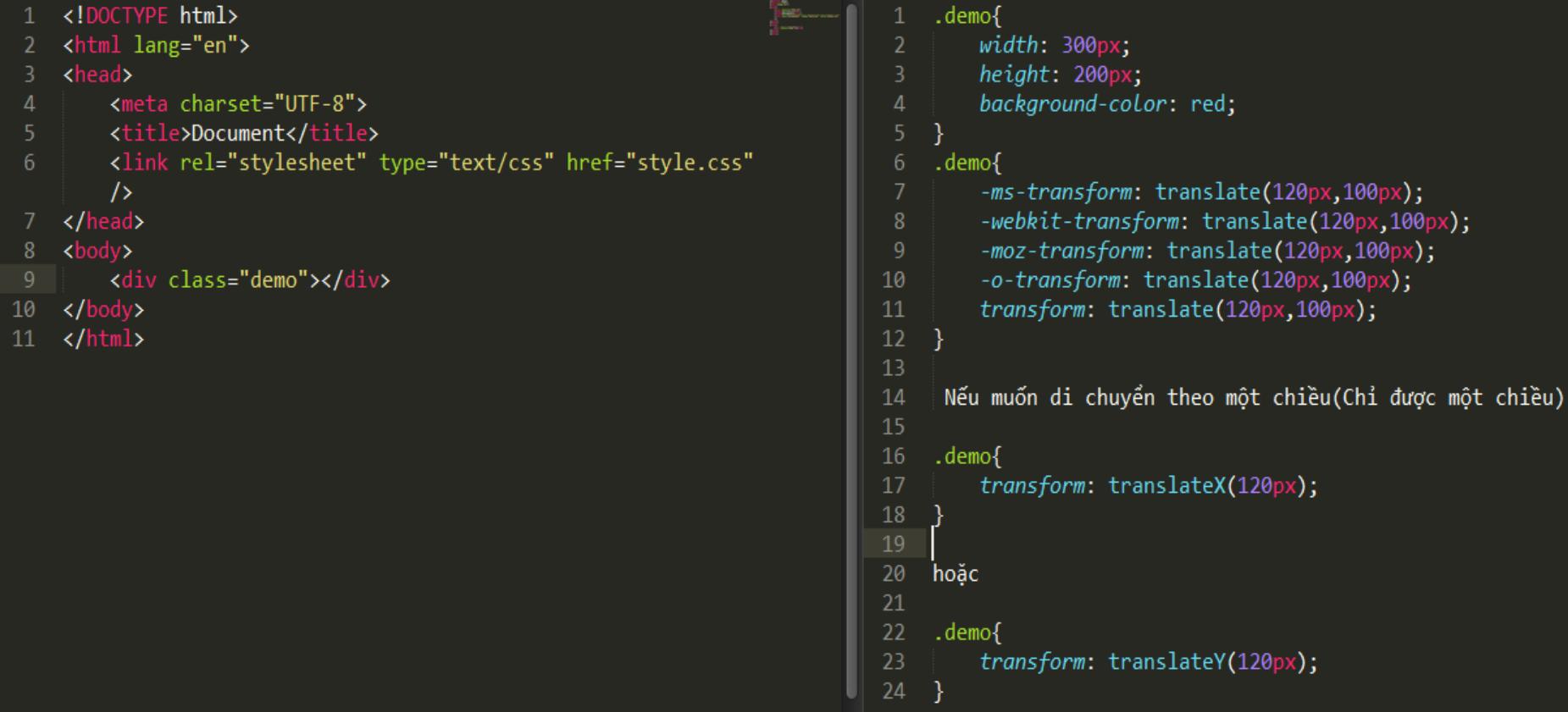
Property					
transform	36.0 4.0 -webkit-	10.0 9.0 -ms-	16.0 3.5 -moz-	9.0 3.2 -webkit-	23.0 15.0 -webkit- 12.1 10.5 -o-
transform-origin (two-value syntax)	36.0 4.0 -webkit-	10.0 9.0 -ms-	16.0 3.5 -moz-	9.0 3.2 -webkit-	23.0 15.0 -webkit- 12.1 10.5 -o-

❖ CSS3 2D Transform

❖ Các hàm sử dụng kèm với transform :

- Translate()
- Rotate()
- Scale()
- SkewX()
- SkewY()
- Matrix()

❖ CSS3 Translate



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <link rel="stylesheet" type="text/css" href="style.css">
7 </head>
8 <body>
9   <div class="demo"></div>
10 </body>
11 </html>
```

```
1 .demo{
2   width: 300px;
3   height: 200px;
4   background-color: red;
5 }
6 .demo{
7   -ms-transform: translate(120px,100px);
8   -webkit-transform: translate(120px,100px);
9   -moz-transform: translate(120px,100px);
10  -o-transform: translate(120px,100px);
11  transform: translate(120px,100px);
12 }
13
14 Nếu muốn di chuyển theo một chiều(Chỉ được một chiều)
15
16 .demo{
17   transform: translateX(120px);
18 }
19 hoặc
20
21 .demo{
22   transform: translateY(120px);
23 }
```

❖ CSS3 Rotate

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Document</title>
6     <link rel="stylesheet" type="text/css" href="style.css"
7     />
8 </head>
9 <body>
10    <div class="demo1"></div>
11    <div class="demo2"></div>
12 </body>
13 </html>
```

```
1 .demo1,.demo2{
2     width: 300px;
3     height: 200px;
4     background-color: red;
5     margin: 100px;
6     float: left;
7 }
8 .demo1{
9     -ms-transform: rotate(40deg);
10    -webkit-transform: rotate(40deg);
11    -moz-transform: rotate(40deg);
12    -o-transform: rotate(40deg);
13    transform: rotate(40deg);
14 }
15 .demo2{
16     -ms-transform: rotate(-40deg);
17     -webkit-transform: rotate(-40deg);
18     -moz-transform: rotate(-40deg);
19     -o-transform: rotate(-40deg);
20     transform: rotate(-40deg);
21 }
```

❖ CSS3 Scale

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Document</title>
6     <link rel="stylesheet" type="text/css" href="style.css"
7         />
8 </head>
9 <body>
10    <div class="demo1"></div>
11 </body>
12 </html>
```

```
1 .demo1,.demo2{
2     width: 300px;
3     height: 200px;
4     background-color: red;
5     margin: 100px;
6     float: left;
7 }
8 .demo1{
9     -ms-transform: scale(2,2);
10    -webkit-transform: scale(2,2);
11    -moz-transform: scale(2,2);
12    -o-transform: scale(2,2);
13    transform: scale(2,2);
14 }
15
```

❖ CSS3 SkewX

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <link rel="stylesheet" type="text/css" href="style.css"
7     />
8 </head>
9 <body>
10  <div class="demo1"></div>
11  <div class="demo2"></div>
12 </body>
13 </html>
```

```
1 .demo1,.demo2{
2   width: 300px;
3   height: 200px;
4   background-color: red;
5   margin: 100px;
6   float: left;
7 }
8 .demo1{
9   -ms-transform: skewX(40deg);
10  -webkit-transform: skewX(40deg);
11  -moz-transform: skewX(40deg);
12  -o-transform: skewX(40deg);
13  transform: skewX(40deg);
14 }
15 .demo2{
16  -ms-transform: skewX(-40deg);
17  -webkit-transform: skewX(-40deg);
18  -moz-transform: skewX(-40deg);
19  -o-transform: skewX(-40deg);
20  transform: skewX(-40deg);
21 }
22 }
```

❖ CSS3 SkewY

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <link rel="stylesheet" type="text/css" href="style.css"
7     />
8 </head>
9 <body>
10  <div class="demo1"></div>
11  <div class="demo2"></div>
12 </body>
13 </html>
```



```
1 .demo1,.demo2{
2   width: 300px;
3   height: 200px;
4   background-color: red;
5   margin: 100px;
6   float: left;
7 }
8 .demo1{
9   -ms-transform: skewY(40deg);
10  -webkit-transform: skewY(40deg);
11  -moz-transform: skewY(40deg);
12  -o-transform: skewY(40deg);
13  transform: skewY(40deg);
14 }
15 .demo2{
16  -ms-transform: skewY(-40deg);
17  -webkit-transform: skewY(-40deg);
18  -moz-transform: skewY(-40deg);
19  -o-transform: skewY(-40deg);
20  transform: skewY(-40deg);
21 }
22 }
```

❖ CSS3 Skew

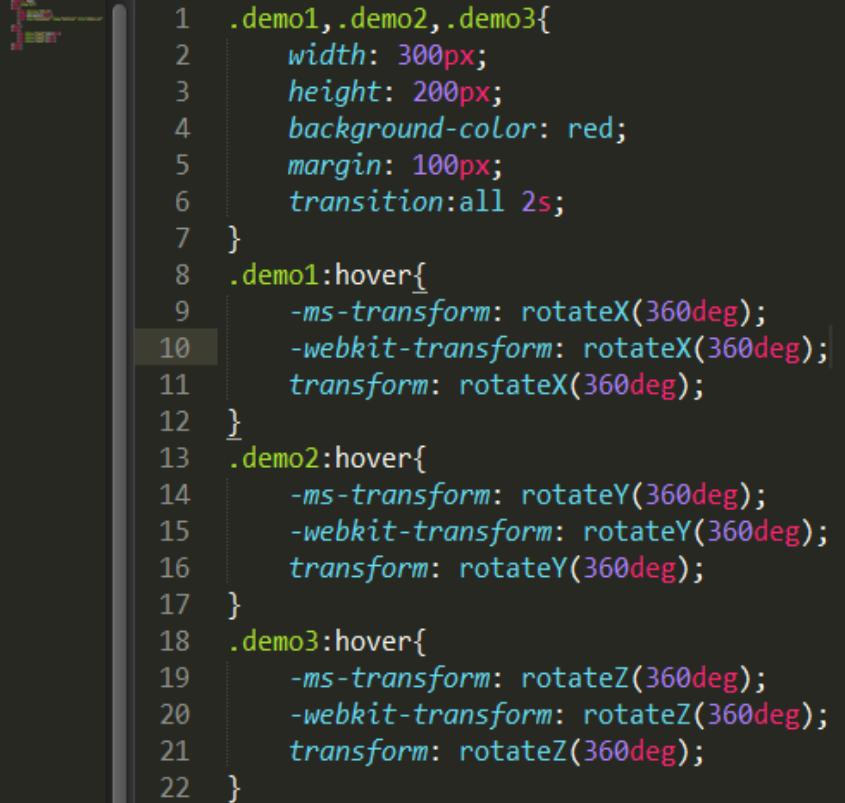
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <link rel="stylesheet" type="text/css" href="style.css"
7     />
8 </head>
9 <body>
10  <div class="demo1"></div>
11  <div class="demo2"></div>
12 </body>
13 </html>
```



```
1 .demo1,.demo2{
2   width: 300px;
3   height: 200px;
4   background-color: red;
5   margin: 100px;
6   float: left;
7 }
8 .demo1{
9   -ms-transform: skewY(40deg);
10  -webkit-transform: skewY(40deg);
11  -moz-transform: skewY(40deg);
12  -o-transform: skewY(40deg);
13  transform: skewY(40deg);
14 }
15 .demo2{
16  -ms-transform: skewY(-40deg);
17  -webkit-transform: skewY(-40deg);
18  -moz-transform: skewY(-40deg);
19  -o-transform: skewY(-40deg);
20  transform: skewY(-40deg);
21 }
```

❖ CSS3 3D Rotate

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <link rel="stylesheet" type="text/css" href="style.css"
7     />
8 </head>
9 <body>
10  <div class="demo1">abc</div>
11  <div class="demo2"></div>
12  <div class="demo3"></div>
13 </body>
14 </html>
```



```
1 .demo1,.demo2,.demo3{
2   width: 300px;
3   height: 200px;
4   background-color: red;
5   margin: 100px;
6   transition: all 2s;
7 }
8 .demo1:hover{
9   -ms-transform: rotateX(360deg);
10  -webkit-transform: rotateX(360deg);
11  transform: rotateX(360deg);
12 }
13 .demo2:hover{
14   -ms-transform: rotateY(360deg);
15   -webkit-transform: rotateY(360deg);
16   transform: rotateY(360deg);
17 }
18 .demo3:hover{
19   -ms-transform: rotateZ(360deg);
20   -webkit-transform: rotateZ(360deg);
21   transform: rotateZ(360deg);
22 }
```

❖ CSS3 Transition

- Giúp chúng ta nhìn thấy được quá trình transform
- 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 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;

❖ CSS3 Transition duration

- Thời gian thực hiện transition.
- Cú pháp:

Transition-duration: [time];

- Ví dụ : Transition-duration : 2s;

❖ CSS3 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 Delay

- Khoảng thời gian delay trước khi thực hiện hiệu ứng.
- Cú pháp:

Transition-delay: [Time] ;

- Ví dụ : Transition-delay: 1s;

❖ CSS3 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 + Transform

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

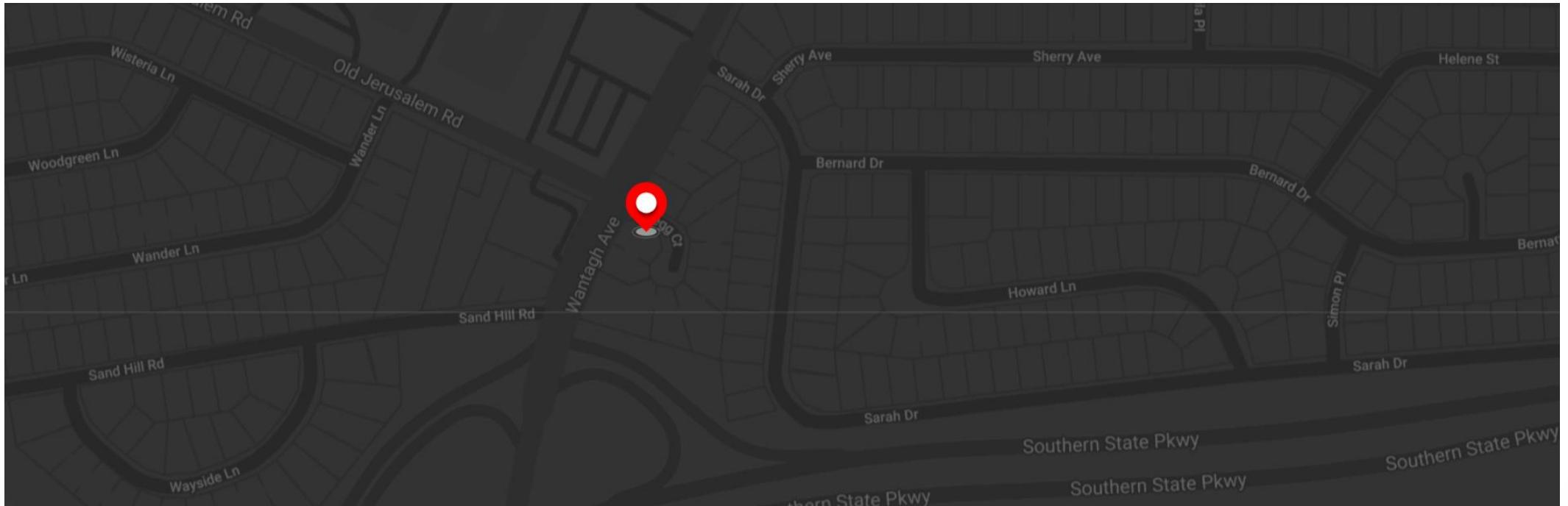
PHẦN 18 : CSS3 ANIMATION

Giới thiệu về animation

- ❖ Như ta đã thấy, transform và transition thường kết hợp với 1 sự kiện xảy ra
- ❖ Với animation, ta có set up hiệu ứng theo ý muốn và có thể mặc định chạy khi load trang web
- ❖ Vậy thì ta hãy cùng tạo ra một animation trong ở phần **address** của layout thử xem.

Yêu cầu

- ❖ Tạo ra bản đồ với animated pointer



Xử lý HTML

Các thành phần html:

- ❖ Màu đỏ là div chứa, có background-img là hình bản đồ
- ❖ Màu xanh là pointer trên map
- ❖ Pointer có 2 thành phần chính
 - ❖ Màu vàng là phần màu đỏ phía trên
 - ❖ Màu trắng mà phần chân phía dưới

```
<!-- END CONTACT -->
<section class="address">
    <div class="address_map">
        <div class="map_pointer">
            <span class="pointer"></span>
            <span class="pulse"></span>
        </div>
    </div>
</section>
```

Xử lý CSS

```
.address__map{
    height:25rem;
    background:url(..../images/map.png);
    position: relative;
    z-index: -2;
}

.map__pointer{
    position:absolute;
    top:35%;
    left:40%;
}

.pointer{
    width:2rem;
    height: 2rem;
    background-color: red;
    display: block;
    border-radius: 50%;
    border-bottom-right-radius: 0;
    transform: rotate(45deg);
}

.pointer:before{
    content: '';
    display: block;
    width:100%;
    height:100%;
    background-color: white;
    border-radius: 50%;
    transform: scale(0.5);
    box-shadow: 5px 4px 10px rgba(0,0,0,0.5)
}
```

```
.pulse{
    width:20px;
    height: 8px;
    background-color: #rgba(255,255,255,0.5);
    display: block;
    border-radius: 50%;
    position: absolute;
    bottom:-12px;
    left:50%;
    transform: translateX(-50%);
    z-index: -1;
}

.pulse:before{
    content: '';
    width:100%;
    height: 100%;
    display: block;
    border:1px solid #rgba(255,255,255,0.5);
    top:-1px;
    left:-1px;
    position: absolute;
    border-radius: 50%;

    z-index:20
}
```

Tạo animation với css3

```
.pulse:before{  
    animation-name: animate; /* Tên Hiệu ứng, do mình tự đặt */  
    animation-duration:0.8s; /* Thời gian thực hiện hiệu ứng */  
    animation-timing-function: linear; /* kiểu chuyển động, linear là chuyển động đều */  
    animation-delay: 0s; /* sau bao nhiêu thời gian thực hiện hiệu ứng */  
    animation-iteration-count: infinite; /* Số lần lặp của hiệu ứng, infinite nghĩa là lặp vô tận */  
    /* hoặc cách viết shorthand */  
    /* animation:animate 0.8s linear 0s infinite; */  
}  
  
@keyframes animate {  
    0%{  
        opacity: 0;  
        transform:scale(1,1);  
    }  
    100%{  
        opacity: 1;  
        transform:scale(1.5,1.5);  
    }  
}
```

- (1) Ta thêm các thuộc tính như hình cho thẻ ta muốn gắn hiệu ứng
- (2) Ta set up hiệu ứng theo ý muốn của mình với từ khóa @keyframes TenHieuUng. Chia thời gian thực hiện hiệu ứng(animation-duration) thành 100%, ở đây 0% là lúc bắt đầu, 100% là lúc kết thúc. Ví dụ, animation-duration là 0.8s được chia thành 100%, từ 0% -> 100% (từ 0s -> 0.8s), phần tử pulse:before sẽ chuyển từ opacity 0 sang opacity 1, từ scale(1,1) sang scale(1.5,1.5)

Phần 19: Review Animation

❖ CSS3 Animation

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <link rel="stylesheet" type="text/css" href="style.css" />
7 </head>
8 <body>
9   <div class="demo1">abc</div>
0 </body>
1 </html>
```

```
1 .demo1{
2   width: 200px;
3   height: 200px;
4   background-color: red;
5   -webkit-animation-name: hieudeptrai;
6   -webkit-animation-duration: 2;
7   animation-name: hieudeptrai;
8   animation-duration: 2s;
9 }
10 @keyframes hieudeptrai {
11   from{
12     width: 200px;
13   }
14   to{
15     width: 600px;
16   }
17 }
18 }
19 }
```

Tên hiệu ứng được đặt thoải mái.

Phải có duration.

Có thể chỉnh theo from...to hoặc theo %

❖ CSS3 Animation – theo %

```
.demo1{  
    width: 200px;  
    height: 200px;  
    background-color: red;  
    -webkit-animation-name: hieudeptrai;  
    -webkit-animation-duration: 2;  
    animation-name: hieudeptrai;  
    animation-duration: 4s;  
}  
@keyframes hieudeptrai {  
    0%{  
        width: 200px;  
    }  
    50%{  
        width: 600px;  
    }  
    100%{  
        width: 200px;  
    }  
}
```

❖ CSS3 Animation – nhiều thuộc tính

```
.demo1{  
    width: 200px;  
    height: 200px;  
    background-color: red;  
    -webkit-animation-name: hieudeptrai;  
    -webkit-animation-duration: 2;  
    animation-name: hieudeptrai;  
    animation-duration: 4s;  
}  
@keyframes hieudeptrai {  
    0%{  
        width: 200px;  
        height: 200px;  
        background-color: red;  
    }  
    50%{  
        width: 600px;  
        height:300px;  
        background-color:blue;  
    }  
    100%{  
        width: 200px;  
        height:200px;  
        background-color:green;  
    }  
}
```

❖ CSS3 Animation delay

- Khoảng thời gian delay trước khi thực hiện hiệu ứng.
- Cú pháp:

Animation-delay: [Time] ;

- Ví dụ : Animation-delay : 1s;

❖ CSS3 Animation iteration-count

- Số lần lặp của hiệu ứng.
- Cú pháp:

Animation-iteration-count: [Times] ;

- Ví dụ : Animation-iteration-count: 3;

Animation-iteration-count: infinite;

❖ CSS3 Animation timing function

- Xác định tốc độ của hiệu ứng
- Cú pháp:

Animation-timing-function: [Types] ;

- Ví dụ :
 Animation-timing-function : ease;

 Animation-timing-function : linear;

 Animation-timing-function : ease-in;

 Animation-timing-function : ease-out;

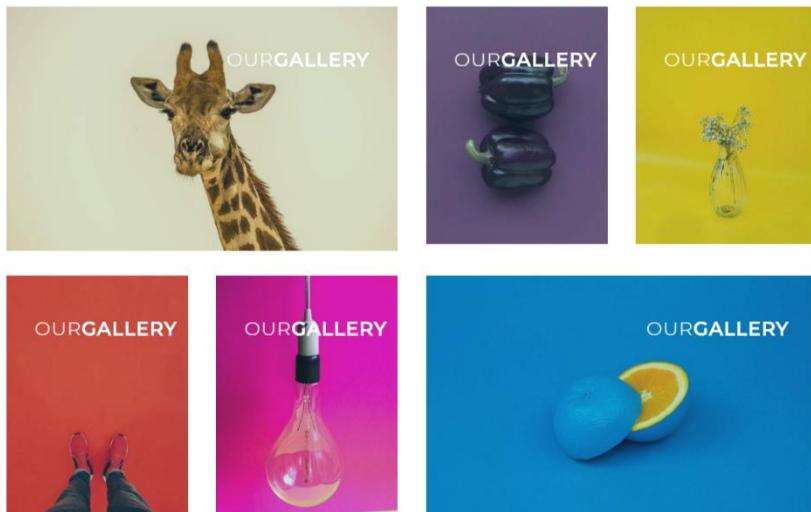
 Animation-timing-function : ease-in-out;

Phần 20: Xử lý phần our gallery với thư viện hiệu ứng

Yêu cầu phần Our gallery

- Thư viện sử dụng
 - ❑ Thư viện hiệu ứng hover: <https://tympanus.net/codrops/2014/06/19/ideas-for-subtle-hover-effects/>
 - ❑ Thư viện slideshow ảnh: <https://lokeshdhakar.com/projects/lightbox2/>

OUR GALLERY



Phần 20: Xử lý phần testimonials với thư viện hiệu ứng

Yêu cầu phần Our gallery

- Thư viện sử dụng
 - ❑ Thư viện hiệu ứng owl carousel: <https://owlcarousel2.github.io/OwlCarousel2/>



Homework (Hoàn tất các phần còn lại trong layout)

Homework

(Xem video form trên codepro và nộp l)
