

HTML5 & CSS3

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Mục Tiêu

01 HTML5 là gì ?

02 So sánh HTML4 và HTML5

03 Tìm hiểu CSS3

04 Thực hành HTML5&CSS3 thông
qua layout Meipaly



Tại sao cần sử dụng HTML5
và CSS3?



HTML5 là gì?

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
CYBERLEARN
Tính năng mới:
- Tags mới: semantic elements, attributes của form, graphic elements, multimedia elements
 - APIs: geolocation, drag và drop, local storage,... (through javascript)



Sematic Elements





Sematic Elements (thẻ ngữ nghĩa)

- HTML4 sử dụng **div** id/class cho các thẻ cơ bản: header, content, footer....
- HTML5 dùng 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>

Từ HTML4 sang HTML5

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



Typical HTML4	Typical HTML5
<div id="header">	<header>
<div id="menu">	<nav>
<div id="content">	<section>
<div class="article">	<article>
<div id="footer">	<footer>

Từ HTML4 sang HTML5

➤ DOCTYPE

- HTML4: <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" >
- HTML5: <!DOCTYPE html>

➤ ENCODING

- HTML4: <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
- HTML5: <meta charset="utf-8">

Sematic Elements (thẻ ngữ nghĩa)

➤ <header>

- Định nghĩa phần header cho một section hoặc trang web

```
<header>
  <div class="header__content">
    <a href="" class="header__logo">...
    </a>
    <nav>...
    </nav>
  </div>
</header>
```

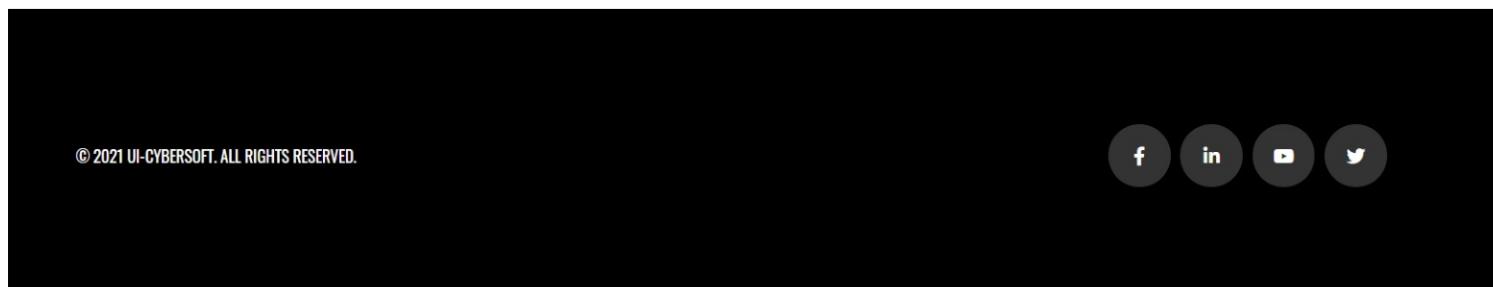


Sematic Elements (thẻ ngữ nghĩa)

➤ <footer>

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

```
<footer>
  <div class="footer__content">
    <p>© 2021 UI-CYBERSOFT. ALL RIGHTS RESERVED.</p>
    <div>...
      </div>
    </div>
  </footer>
```



Sematic Elements (thẻ ngữ nghĩa)

➤ <nav>

- Định nghĩa thanh navigation



```
<nav>
  <ul>
    <li class="active"><a href="#">HOME</a></li>
    <li><a href="#">SERVICE</a></li>
    <li><a href="#">showcase</a></li>
    <li><a href="#">blog</a></li>
    <li><a href="#">ABOUT US</a></li>
    <li><a href="#">CONTACT</a></li>
  </ul>
</nav>
```



Sematic Elements (thẻ ngữ nghĩa)

➤ <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>
```

Sematic Elements (thẻ ngữ nghĩa)

➤ <figure> và <figcaption>

- figure : chứa các nội dung liên quan đến hình ảnh, sơ đồ
- Tag <figure> thường dùng kèm với <figcaption> để thêm caption cho hình.

```
<figure>
  
  <figcaption>CYBERSOFT</figcaption>
</figure>
```



CYBERSOFT

Sematic Elements (thẻ ngữ nghĩa)

➤ <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 class="browser">
  ...<h2>Google Chrome</h2>
  ...<p>Google Chrome is a web browser developed by
    Google</p>
  ...</article>
  ...<article class="browser">
  ...<h2>Mozilla Firefox</h2>
  ...<p>Mozilla Firefox is an open-source web
    browser developed by Mozilla.</p>
  ...</article>
```

Google Chrome

Google Chrome is a web browser developed by Google

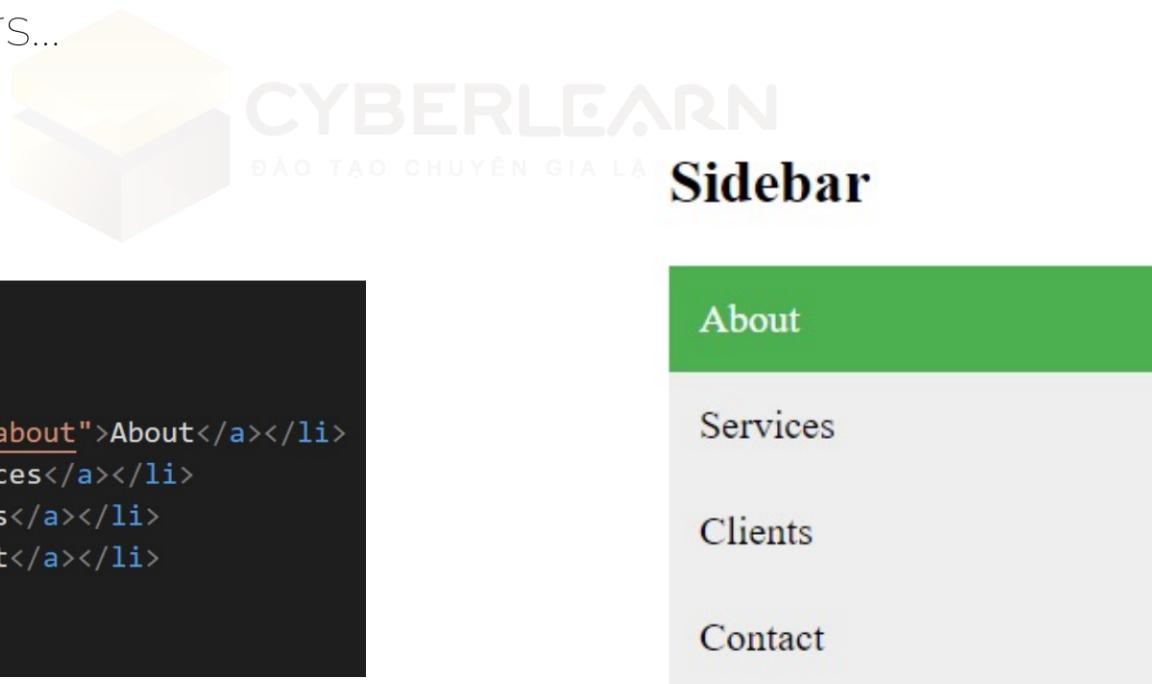
Mozilla Firefox

Mozilla Firefox is an open-source web browser developed by Mozilla.

Sematic Elements (thẻ ngữ nghĩa)

➤ <aside>

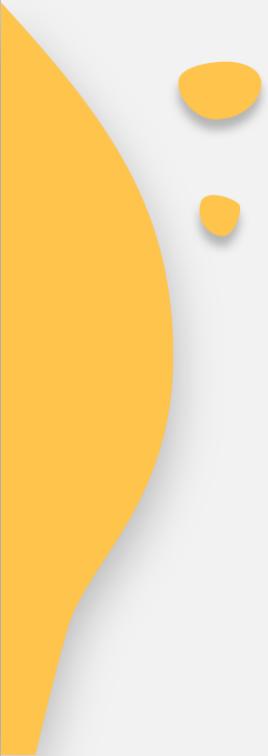
- Chứa một phần nội dung mà liên quan gián tiếp tới phần còn lại của trang như sidebars...



The screenshot shows a website header with the text "CYBERLEARN ĐÀO TẠO CHUYÊN GIA LÀ" and a large "Sidebar" heading. Below the header is a sidebar menu with four items: "About", "Services", "Clients", and "Contact". The "About" item is highlighted with a green background.

```
<aside class=" sidenav">
    ....<h2>Sidebar</h2>
    ....<ul>
        ....<li class="active"><a href="#about">About</a></li>
        ....<li><a href="#services">Services</a></li>
        ....<li><a href="#clients">Clients</a></li>
        ....<li><a href="#contact">Contact</a></li>
    ....</ul>
</aside>
```

Multimedia Elements



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>
```

Multimedia Elements

➤ <video>

- Thêm video cho trang web với các thuộc tính
 - Controls: Hiển thị thanh điều khiển video
 - Preload: load trước video
 - Loop: lặp video
 - Muted: tắt âm thanh của video khi load trang web
 - Autoplay: tự chạy video khi load trang web (một số trình duyệt bắt buộc phải dùng chung với muted)

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

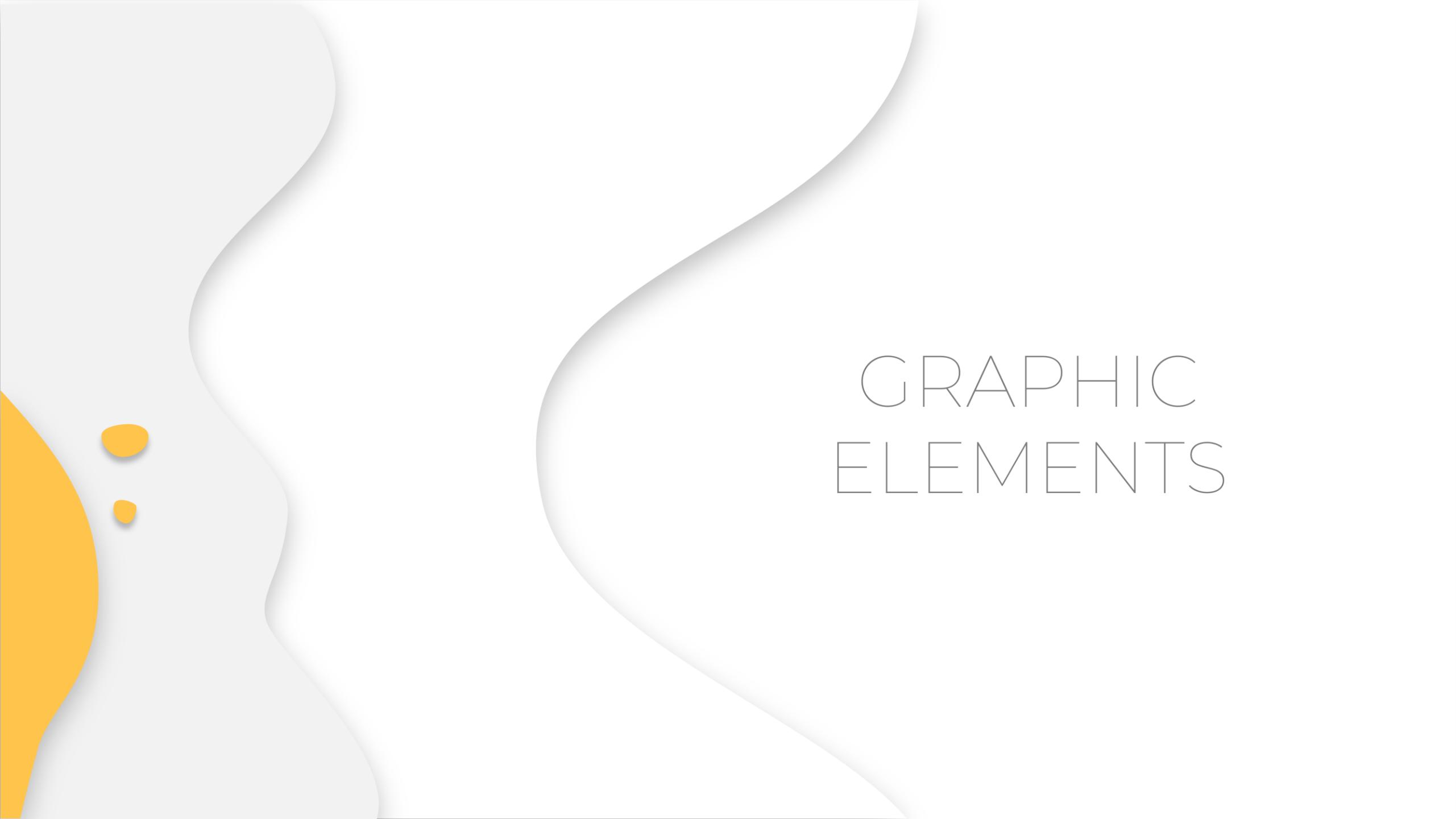
Multimedia Elements

➤ < 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>
```





GRAPHIC ELEMENTS

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>
```

GRAPHIC ELEMENTS

- <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>
```

HTML 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

HTML FORM

➤ 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

➤ 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/POST

➤ 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 Adress bar (khoảng 3.000 ký tự)
- Không dùng 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/POST

➤ 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

Thông tin người dùng nhập hiển thị
trên url

First name:

Mickey

Last name:

Mouse

Submit

PHƯƠNG THỨC GET/POST

➤ 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 GET/POST

➤ 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>
```

ⓘ file:///D:/action_page.php

Thông tin người dùng 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

Các thành phần của form

➤ <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 nhưng vẫn tab được và focus vào thẻ
 - Size: kích thước của khung nhập liệu
 - Maxlength: chiều dài tối đa cho phép nhập liệu
 - Tabindex: Thứ tự focus khi nhấn nút tab
 - Value: giá trị trong ô input
 - Disabled: Không cho nhập dữ liệu và cũng không thể tab tới thẻ
- https://www.w3schools.com/tags/tag_input.asp

Các thành phần của form

- <input>

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

Tên đăng nhập:

Các thành phần của form

➤ <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>
```

First name:

Mickey

Last name:

Mouse

Submit

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

Các thành phần của form

➤ <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

Các thành phần của form

➤ 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

Các thành phần của form

➤ 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

Các thành phần của form

- <input type="button">

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

Đăng nhập

Các thành phần của form

➤ <label>

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

Các thành phần của form

➤ 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

Các thành phần của form

➤ 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

Các thành phần của form

➤ 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

Các thành phần của form

- 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

Các thành phần của form

➤ <textarea>

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

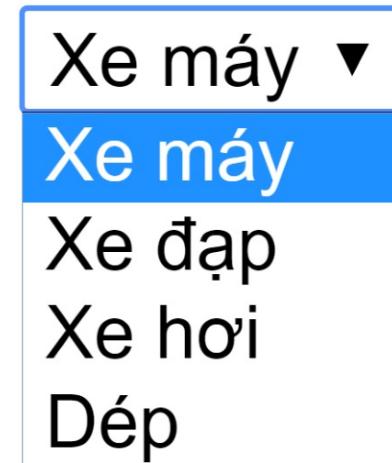
```
<form method="post" action="trangchu.php">
    <textarea cols="40" rows="10"></textarea>
</form>
```

Các thành phần của 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>
```

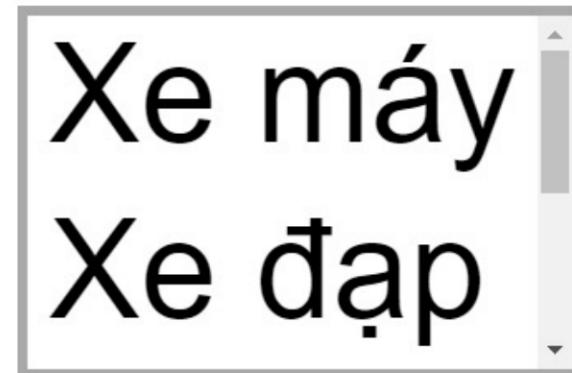


Các thành phần của form

➤ 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>
```

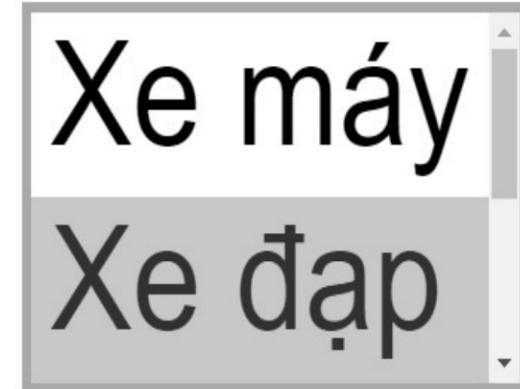


Các thành phần của form

➤ COMBOBOX

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

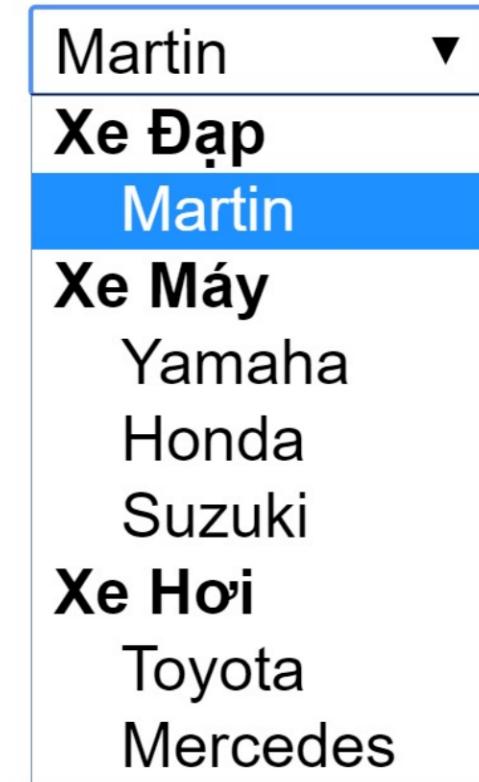
```
<form method="post" action="trangchu.php">
    <select multiple size="2">
        <option value="xeMay">Xe máy</option>
        <option value="xeDap" selected>Xe đạp</option>
        <option value="xeHoi">Xe hơi</option>
        <option value="dep">Dép</option>
    </select>
</form>
```



Các thành phần của 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>
```



Các thành phần của 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

Các thành phần của form

- 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>
```

Các thành phần của 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>
```

Các thành phần của form

➤ MIN & MAX

- Set giá trị trị nhỏ nhất và giá trị lớn nhất 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>
```

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 được sử dụng:

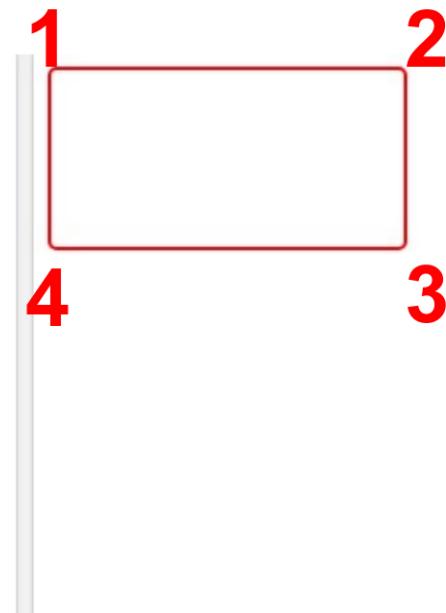
- Border-radius
- Background-size
- Linear-gradient
- Text-shadow
- Box-shadow
- Before,after
- @font-face
- CSS variables

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

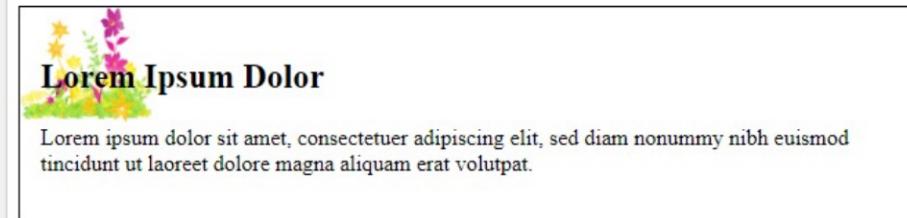
➤ background-size :

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

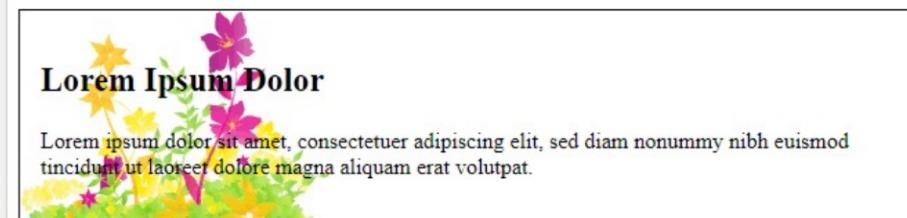
```
<!DOCTYPE html>
<html>
<head>
<style>
#example1 {
    border: 1px solid black;
    background:url(img_flwr.gif);
    background-size: 100px 80px;
    background-repeat: no-repeat;
    padding:15px;
}

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

Điều chỉnh kích thước background:



Mặc định



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

➤ 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

Các thuộc tính của 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.
- Cú pháp: background: linear-gradient(direction, color-stop1, color-stop2, ...);

```
<!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
```

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.

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

Các thuộc tính của CSS3

➤ Linear Gradients

```
<!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.

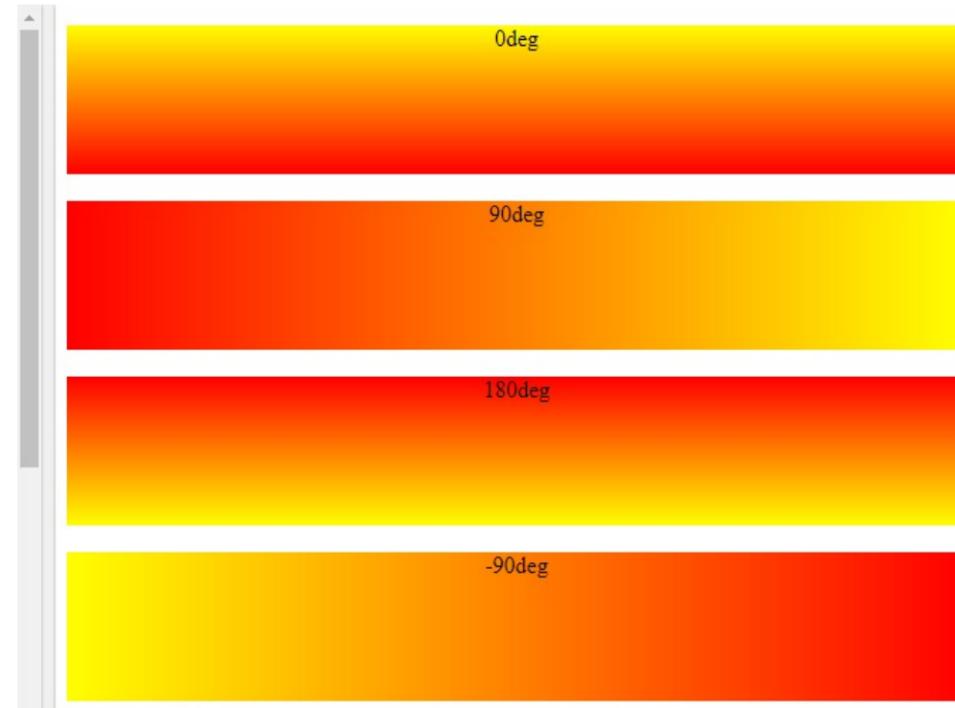
Linear Gradient : Trải theo đường chéo

Các thuộc tính của CSS3

➤ Linear Gradients

```
<!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) */
```



Linear Gradient : sử dụng degree

Các thuộc tính của CSS3

➤ Linear Gradients

```
<!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.

Linear Gradient : kết hợp nhiều màu trǎi

Các thuộc tính của 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>
```

Text-shadow effect!

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

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

Các thuộc tính của 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!

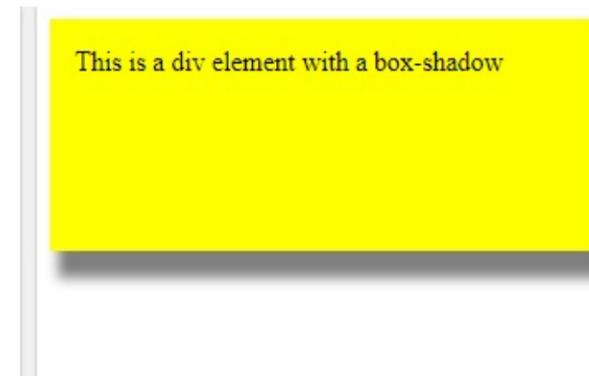
Note: Internet Explorer 9 and earlier versions, do not support the text-shadow property.

Các thuộc tính của CSS3

➤ Box Shadow

- Thuộc tính giúp làm nổi một khối các phần tử
- Box-shadow:[val1] [val2] [val3] [val4];

```
<!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

Các thuộc tính của 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);
}

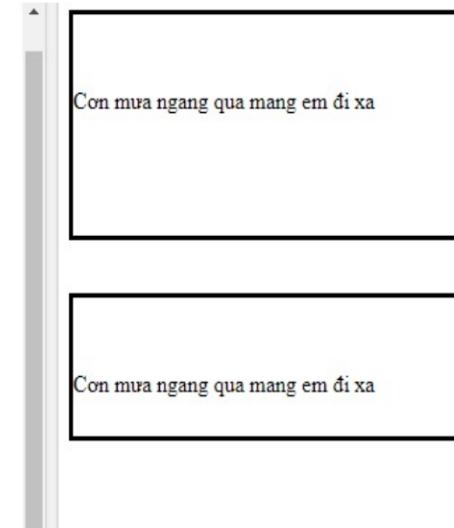
* {
    font-family: myFirstFont;
}
</style>
```

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 :
 - $\text{width} + \text{padding} + \text{border} = \text{chiều rộng của phần tử}$
 - $\text{height} + \text{padding} + \text{border} = \text{chiều cao của phần tử}$
- Cách tính 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>
```



Các thuộc tính của CSS3

➤ Before và After

- Cho phép chèn nội dung vào trước hoặc sau phần tử HTML.
- Thường được dùng để tạo các symbol font (font awesome, material)



Các thuộc tính của CSS3

➤ 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>
```



Các thuộc tính của CSS3

➤ 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)

EARN
A LẬP TRÌNH

CyberSoft Academy



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;



CSS3 Transition

➤ Transition Delay

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

➤ 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 – 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 sự 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;  
}
```



CSS3 Transform

CSS3 Transform

- CSS3 cung cấp cho ta một số thuộc tính để tạo hiệu ứng, làm cho trang web thêm sinh động
- 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-



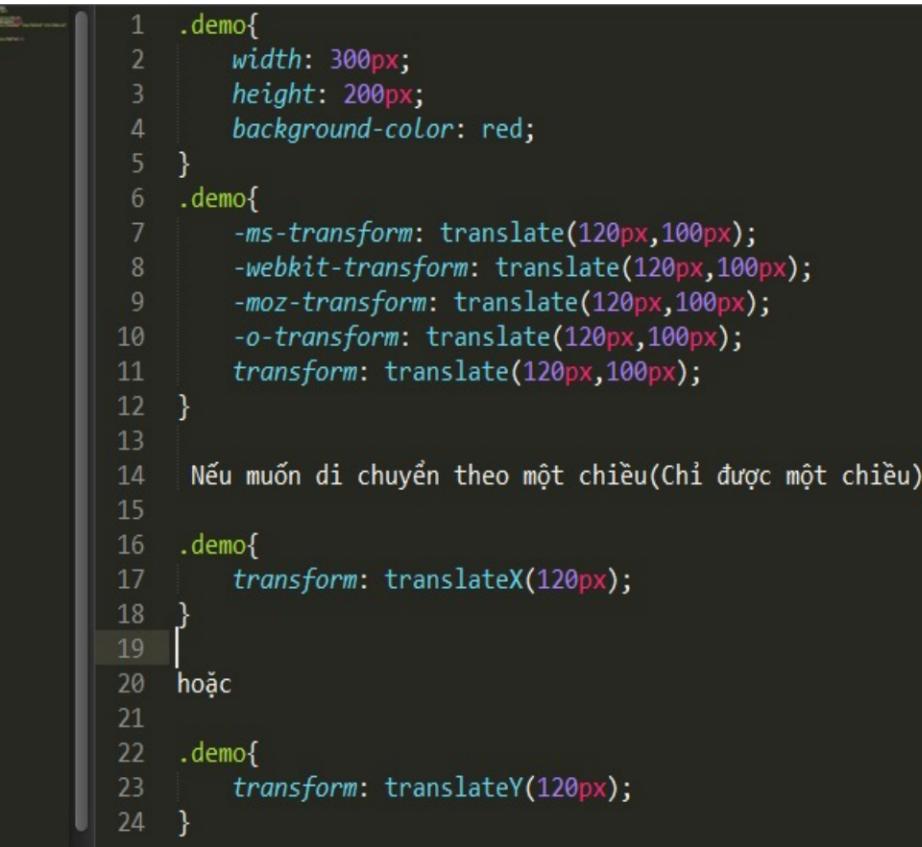
CSS3 Transform

- Các hàm sử dụng kèm với transform
 - Translate()
 - Rotate()
 - Scale()
 - SkewX()
 - SkewY()
 - Matrix()

CSS3 Transform

➤ 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   />
8 </head>
9 <body>
10  <div class="demo"></div>
11 </body>
12 </html>
```



The screenshot shows a code editor with two panes. The left pane displays the HTML code above. The right pane shows the corresponding CSS code:

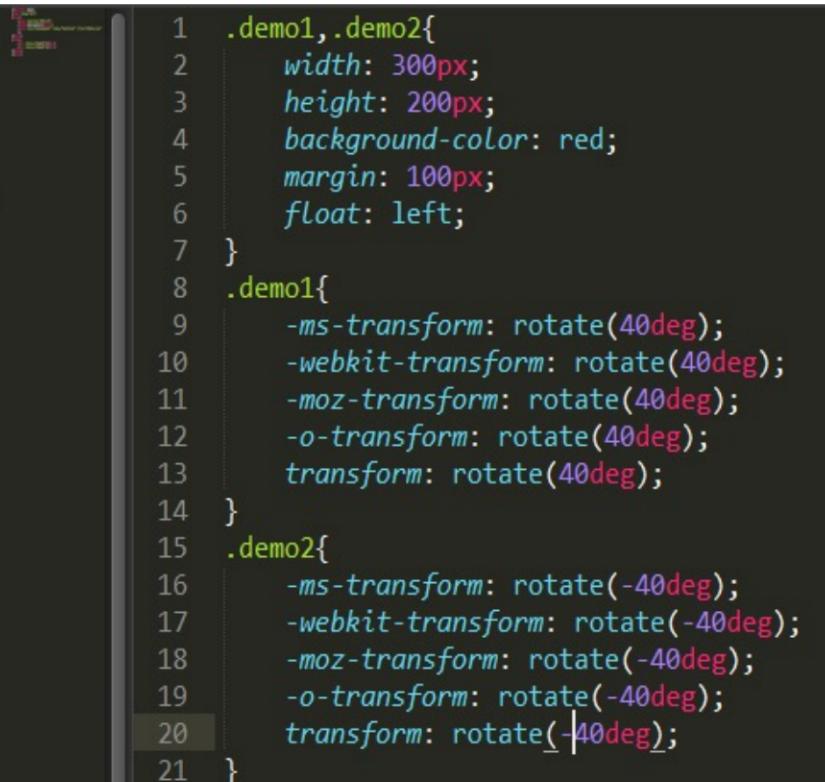
```
1 .demo{
2   width: 300px;
3   height: 200px;
4   background-color: red;
5 }
6 .demo{
7   -ms-transform: translate(120px,100px);
8   -webkit-transform: translate(120px,100px);
9   -moz-transform: translate(120px,100px);
10  -o-transform: translate(120px,100px);
11  transform: translate(120px,100px);
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 .demo{
21   transform: translateY(120px);
22 }
```

A vertical scrollbar is visible between the two panes.

CSS3 Transform

➤ 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>
```



The screenshot shows a code editor with two columns of CSS code. The left column contains the HTML structure with two div elements, one with class "demo1" and one with class "demo2". The right column contains the corresponding CSS styles. Both classes have a width of 300px, a height of 200px, a red background color, and a 10px margin. The "demo1" class uses the rotate(40deg) transform, while the "demo2" class uses the rotate(-40deg) transform. The CSS code includes vendor-specific prefixes (-ms-, -webkit-, -moz-, -o-) followed by the standard transform property.

```
1 .demo1,.demo2{
2   width: 300px;
3   height: 200px;
4   background-color: red;
5   margin: 10px;
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 Transform

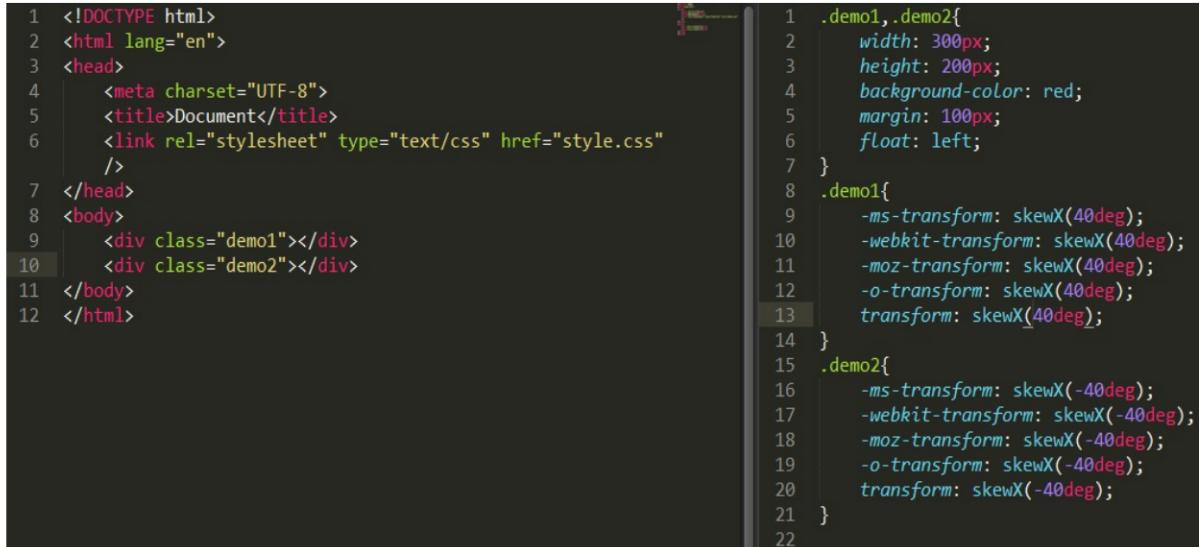
➤ Scale

```
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 Transform

➤ Skew

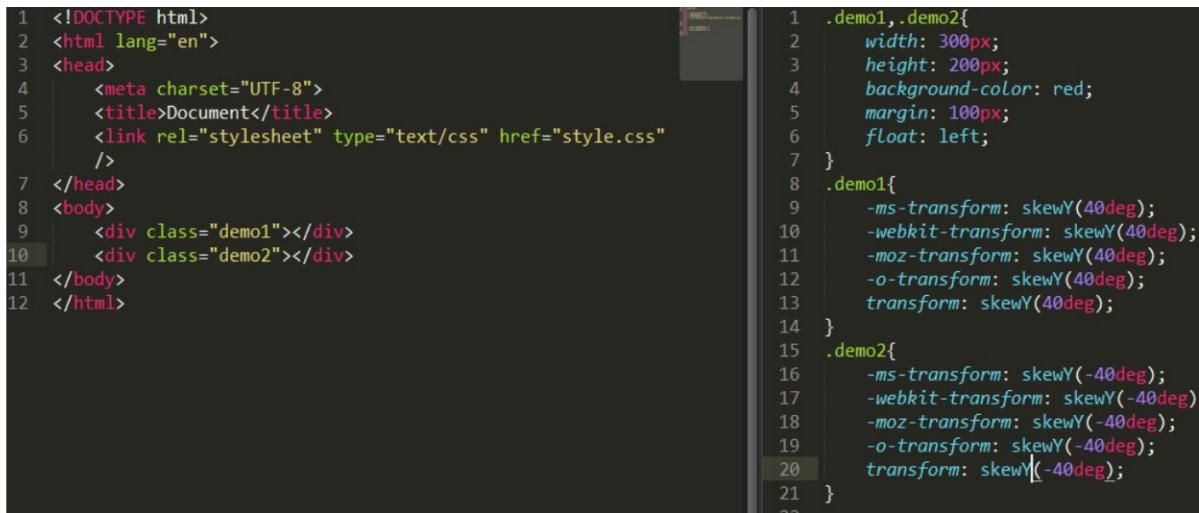


The image shows a code editor with two panes. The left pane displays the following HTML code:

```
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"></div>
10  <div class="demo2"></div>
11 </body>
12 </html>
```

The right pane displays the corresponding CSS code:

```
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
```



The image shows a code editor with two panes. The left pane displays the same HTML code as the previous screenshot:

```
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"></div>
10  <div class="demo2"></div>
11 </body>
12 </html>
```

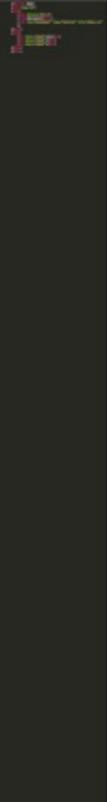
The right pane displays the corresponding CSS code:

```
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 Transform

➤ 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 }
```



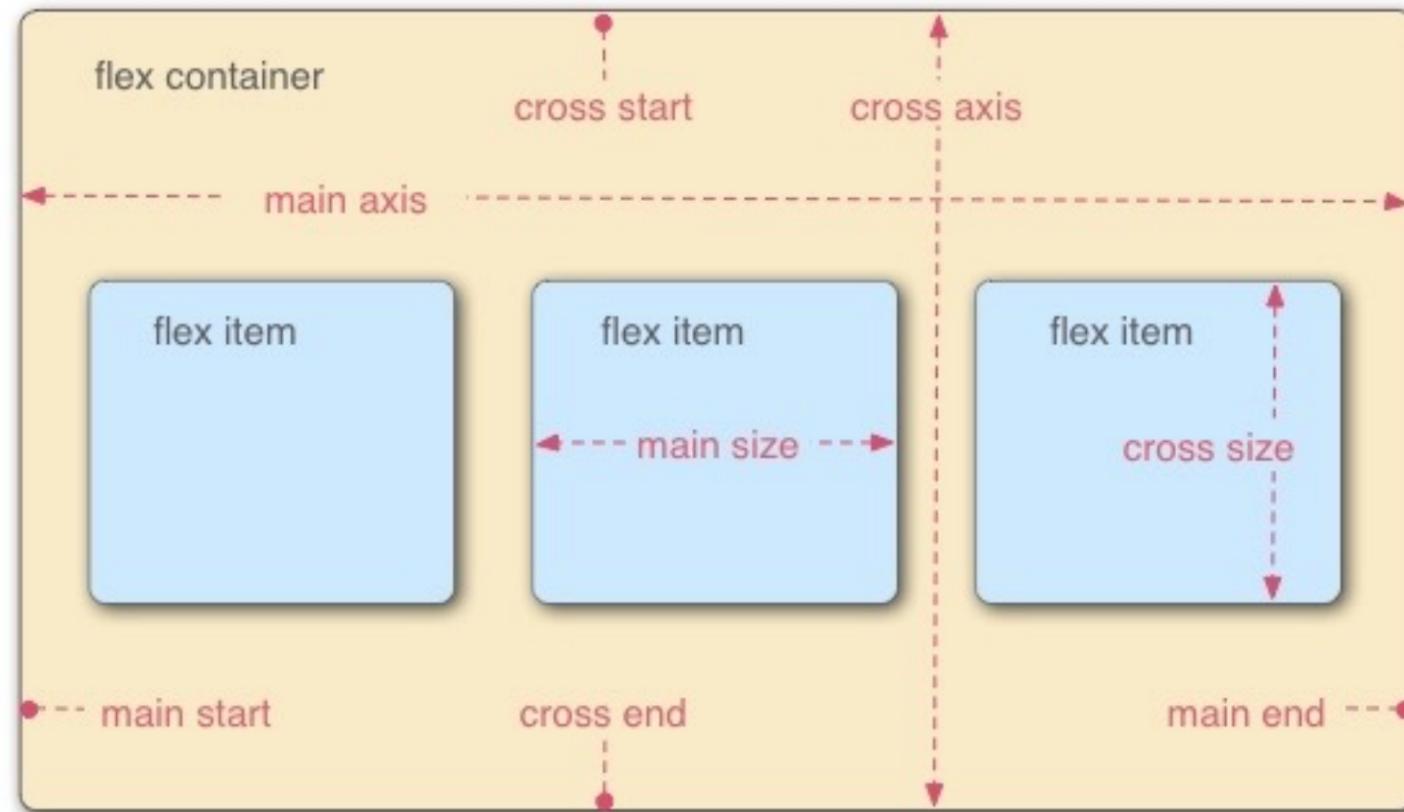
Flexbox



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

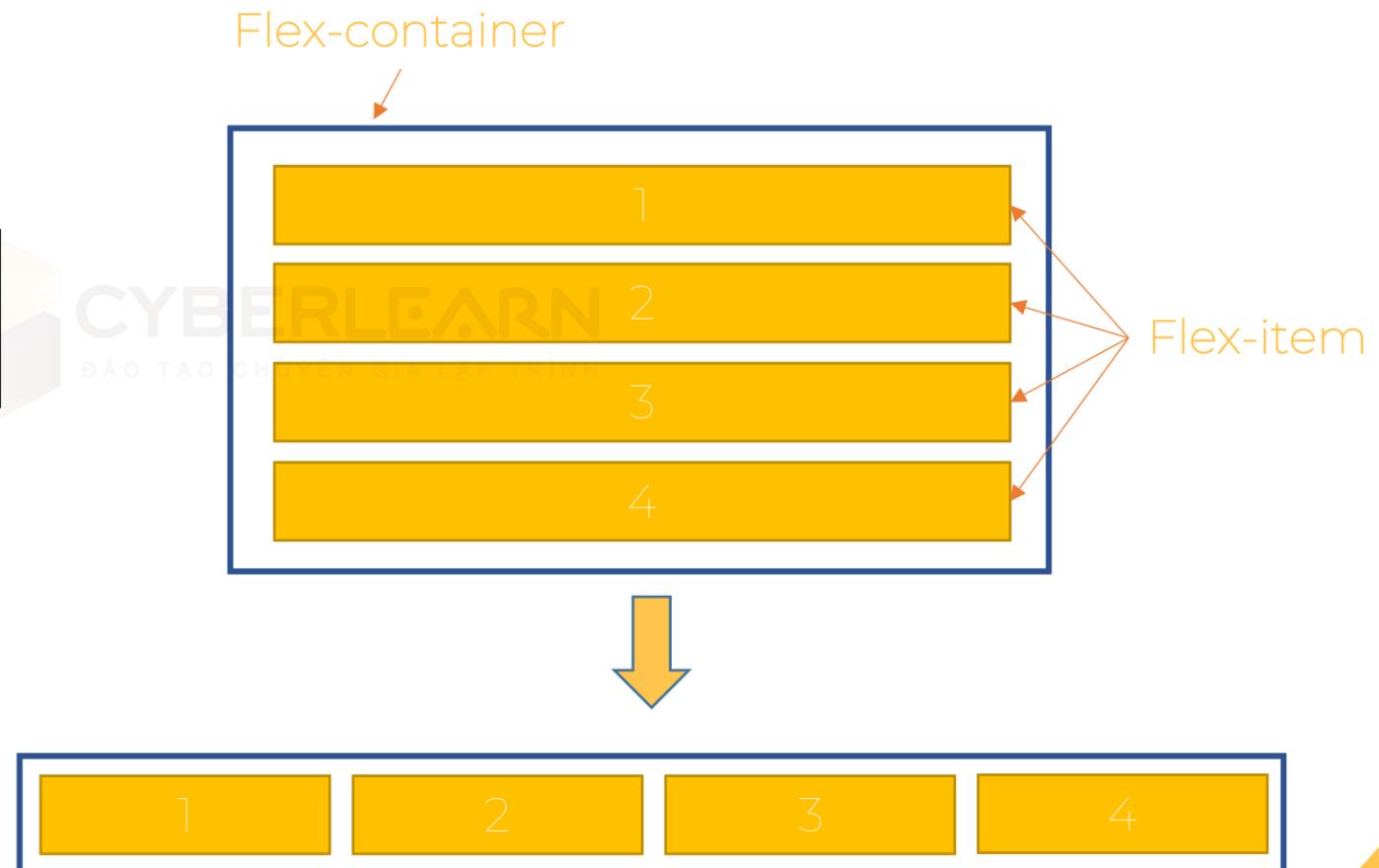
Các thành phần trong flexbox



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;|  
}
```

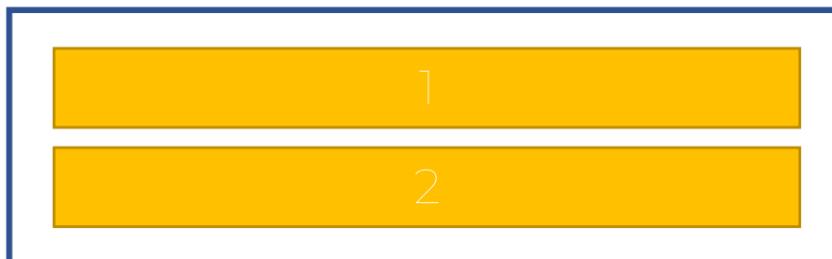
row



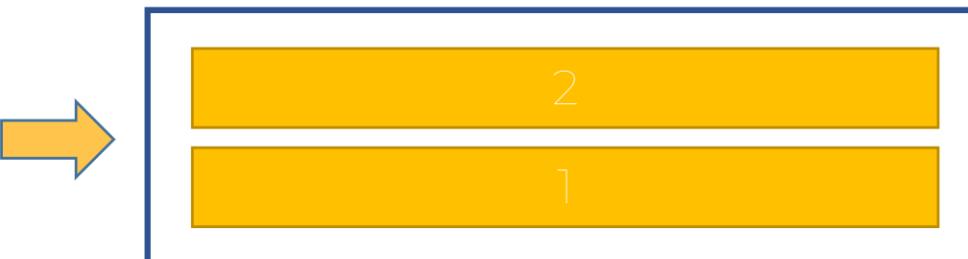
row-reverse



column



column-reverse



Các thuộc tính của flexbox

➤ flex-wrap :

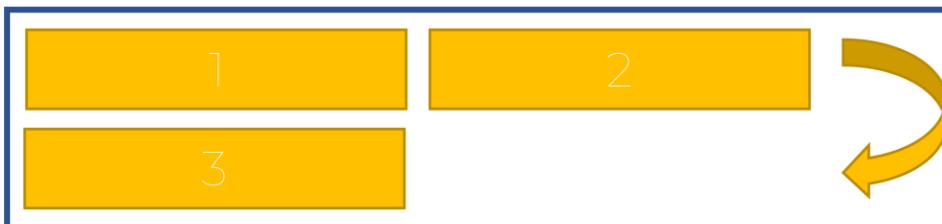
Cho phép các phần tử được xuống hàng khi không đủ chỗ chứa

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

nowarp



warp



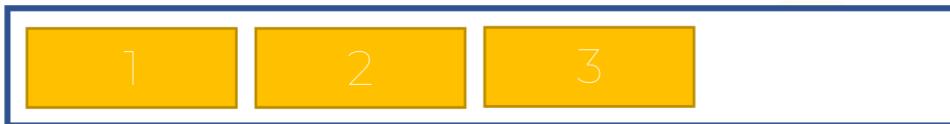
Các thuộc tính của flexbox

➤ 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



center



Space-between



Flex-end



Space-around



Space-evenly

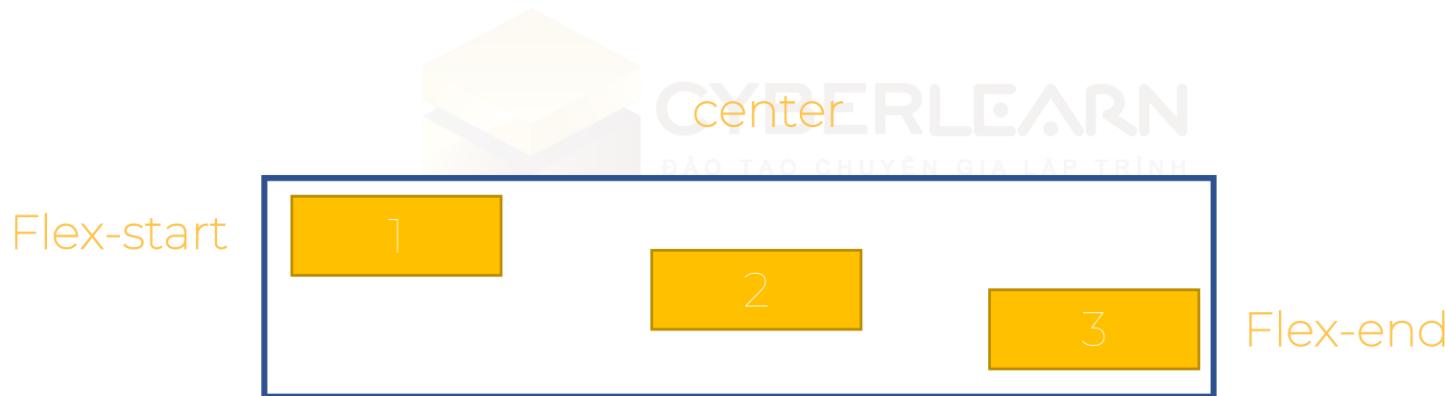


Các thuộc tính của flexbox

➤ 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

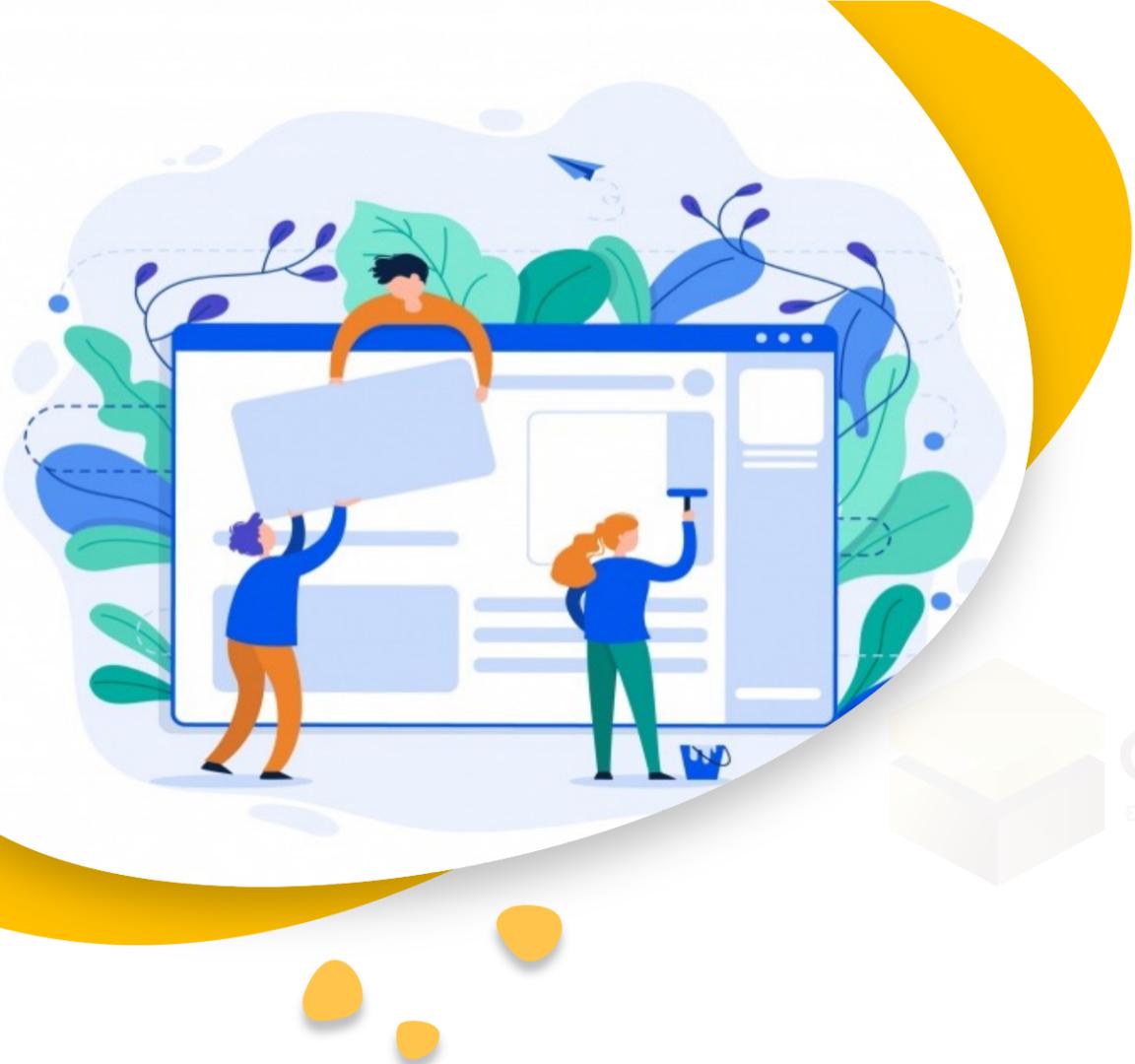
➤ flex-grow:

- 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



The background features a minimalist design with large, white, organic, cloud-like shapes that overlap each other. In the bottom left corner, there is a solid yellow circle containing three smaller, semi-transparent yellow circles of varying sizes.

Grid

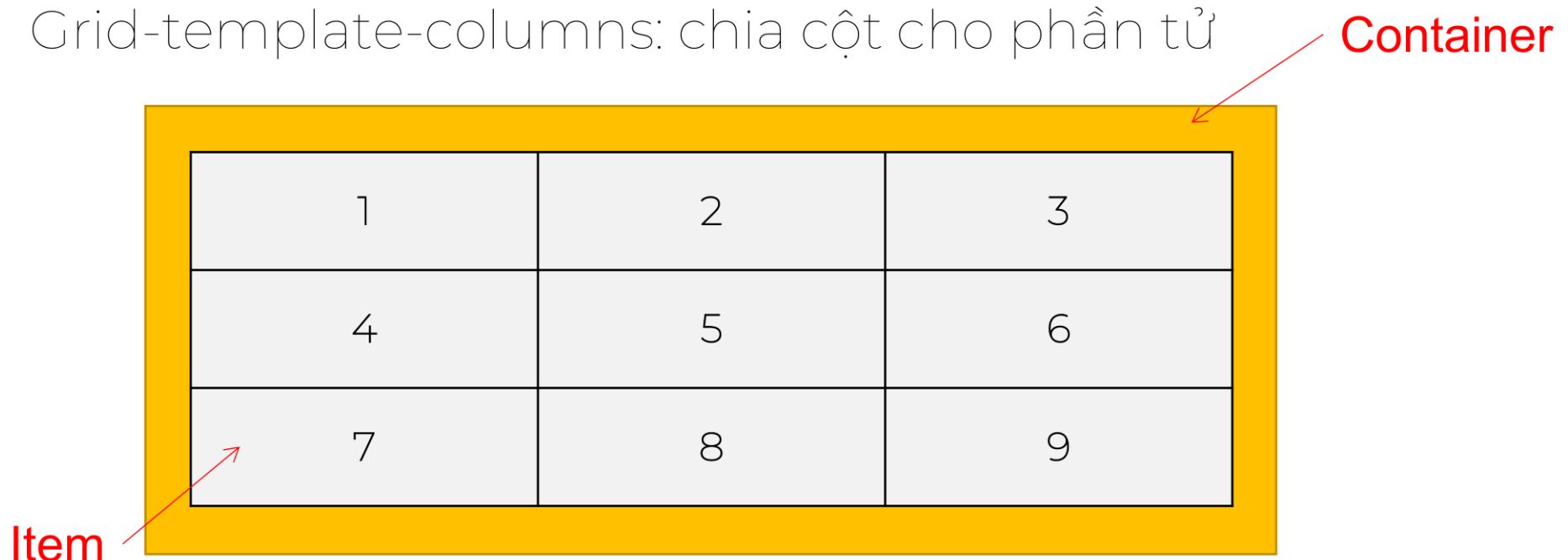


Grid là gì?

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

Các thành phần trong Grid

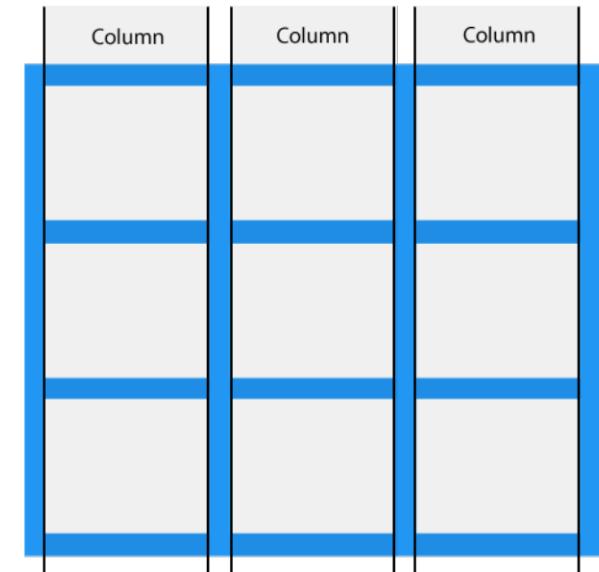
- Gồm 2 thành phần chính :
 - Grid Container
 - Grid Item
- Muốn sử dụng css Grid phải có các thuộc tính :
 - Display: grid
 - Grid-template-columns: chia cột cho phần tử



Các thuộc tính của Grid

➤ Grid-template-columns

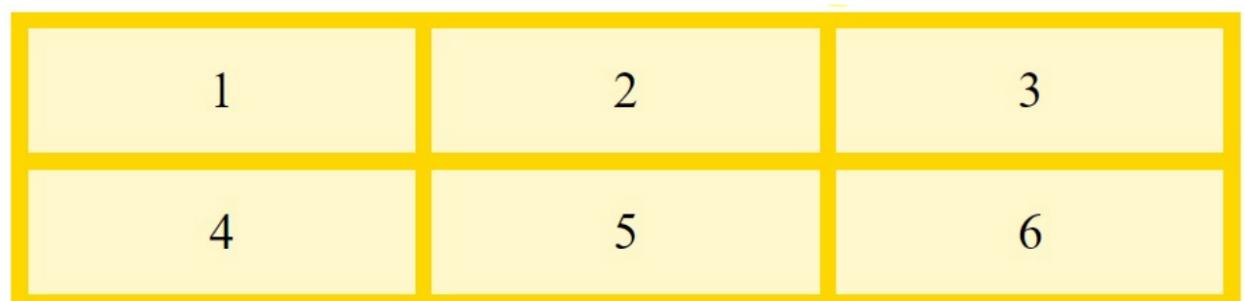
- Dùng để chia cột cho phần tử
- Cú pháp:
 - Grid-template-columns:[col1] [col2] [col3]...;
 - Grid-template-columns: repeat(số cột, kích thước);
- Đơn vị kích thước ưu tiên dùng “fr”
- Ví dụ: grid-template-columns: 1fr 1fr 2fr
=> chia phần tử ra thành 4 phần bằng nhau, cột 1 và 2 chiếm một phần, cột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



Các thuộc tính của Grid

➤ Grid-template-columns

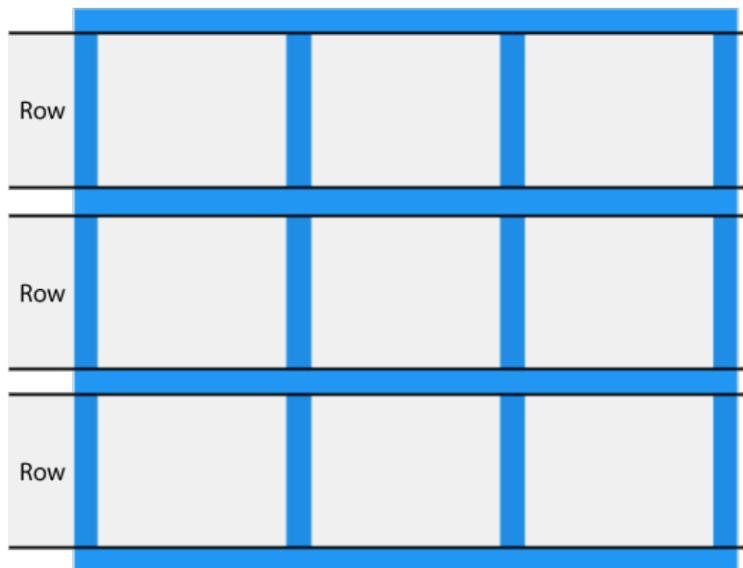
```
.grid-container {  
    display: grid;  
    grid-template-columns: auto auto auto;  
    grid-gap: 10px;  
    background-color: #gold;  
    padding: 10px;  
}  
  
.grid-container > div {  
    background-color: #rgba(255, 255, 255, 0.8);  
    text-align: center;  
    padding: 20px 0;  
   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



Các thuộc tính của Grid

➤ Grid-template-rows

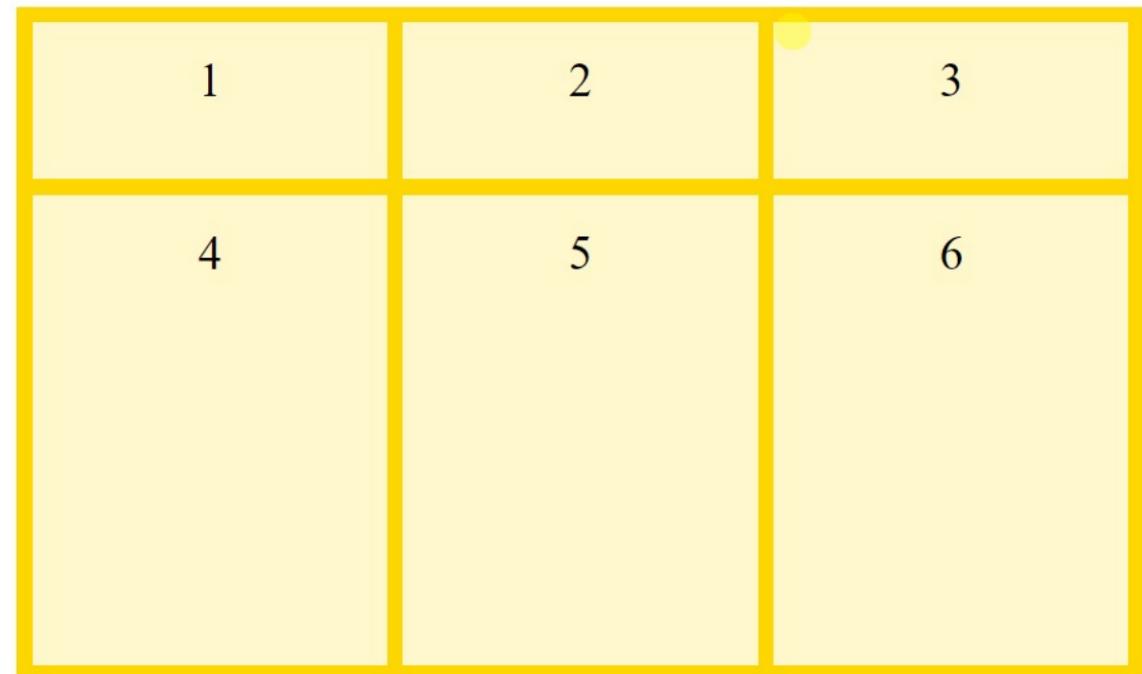
- Cú pháp:
 - Grid-template-rows:[row1] [row2] [row3]...;
 - Chỉ nên xét khi muốn chỉnh height cho các rows.



Các thuộc tính của Grid

➤ Grid-template-rows

```
.grid-container {  
    display: grid;  
    grid-template-columns: auto auto auto;  
    grid-template-rows: 100px 300px;  
    grid-gap: 10px;  
    background-color: #gold;  
    padding: 10px;  
}  
  
.grid-container > div {  
    background-color: #rgba(255, 255, 255, 0.8);  
    text-align: center;  
    padding: 20px 0;  
    font-size: 30px;  
}
```

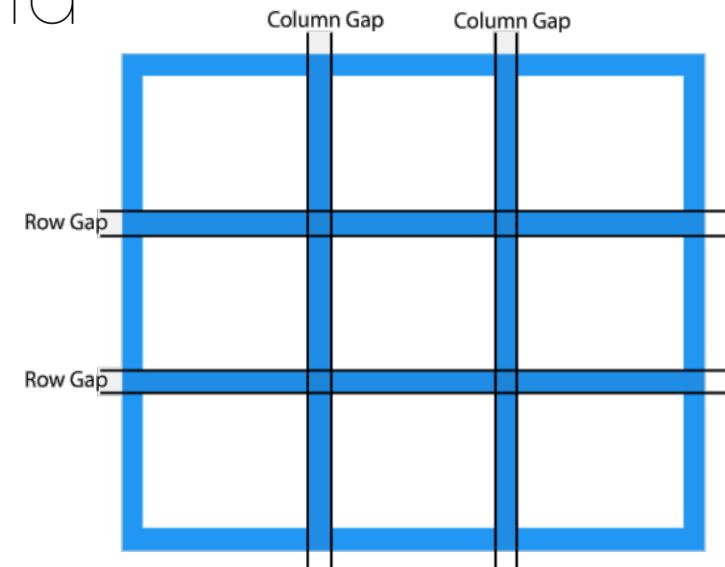
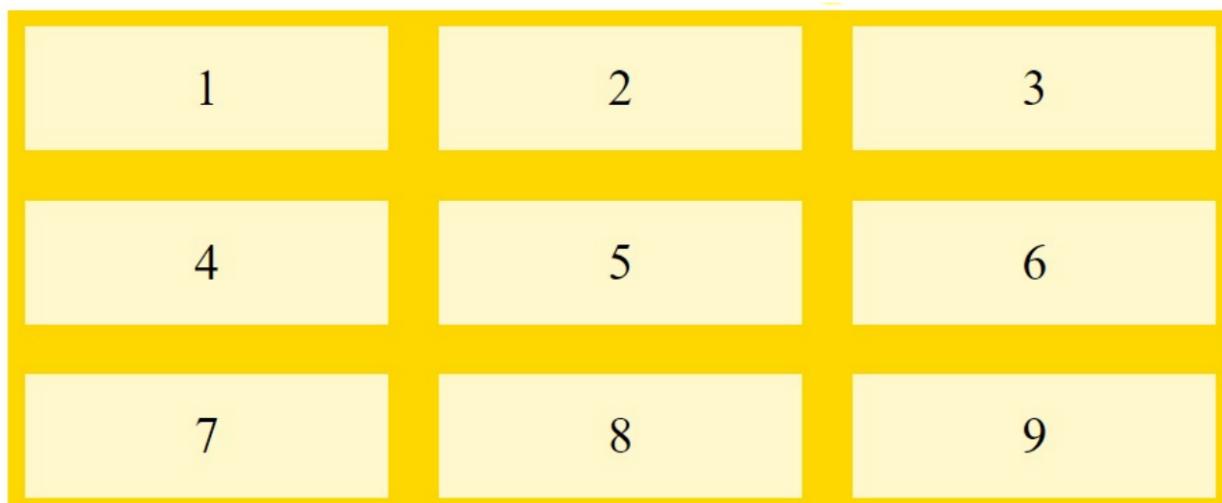


Các thuộc tính của Grid

➤ Grid Gaps

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

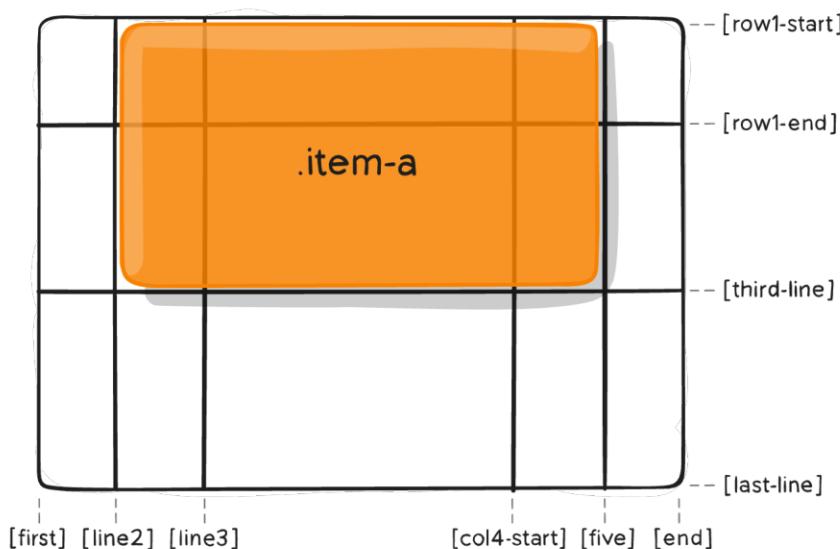
```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  grid-gap: 30px;  
  background-color: #gold;  
  padding: 10px;  
}  
  
.grid-container > div {  
  background-color: #rgba(255, 255, 255, 0.8);  
  text-align: center;  
  padding: 20px 0;  
  font-size: 30px; 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



Các thuộc tính của Grid

➤ Grid-column & Grid-row:

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

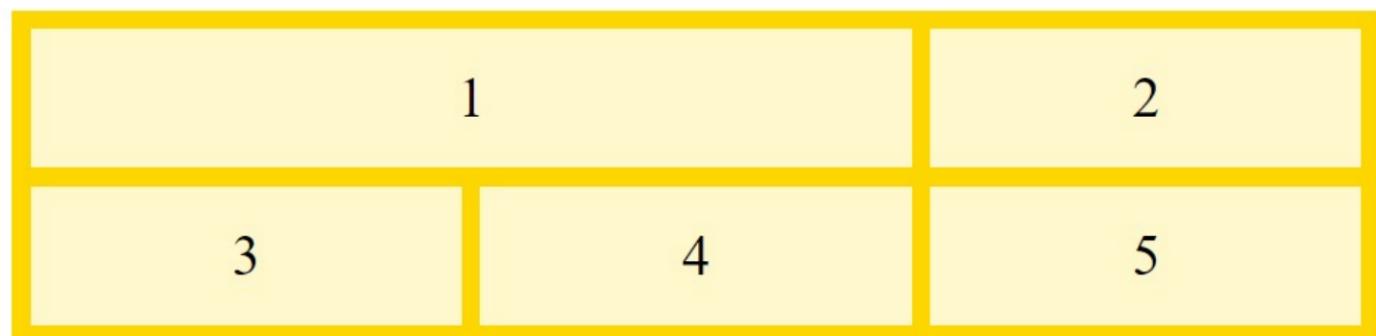


Các thuộc tính của Grid

➤ Grid-column:

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

```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  grid-gap: 10px;  
  background-color: #gold;  
  padding: 10px;  
}  
  
.grid-container > div {  
  background-color: #rgba(255, 255, 255, 0.8);  
  text-align: center;  
  padding: 20px 0;  
  font-size: 30px;  
}  
  
.item1 {  
  grid-column: 1 / 3;  
}
```

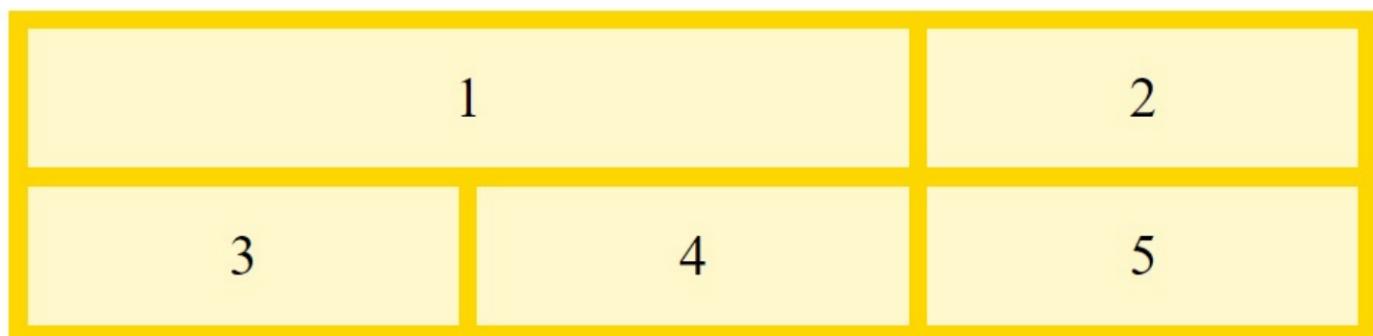


Các thuộc tính của Grid

➤ Grid-column:

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

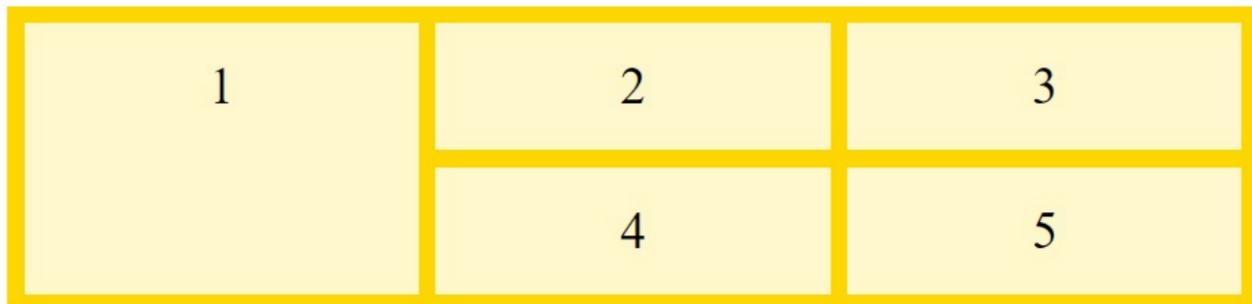
```
.grid-container {  
    display: grid;  
    grid-template-columns: auto auto auto;  
    grid-gap: 10px;  
    background-color: #gold;  
    padding: 10px;  
}  
  
.grid-container > div {  
    background-color: #rgba(255, 255, 255, 0.8);  
    text-align: center;  
    padding: 20px 0;  
    font-size: 30px;  
}  
  
.item1 {  
    grid-column: 1 / span 2;  
}
```



Các thuộc tính của Grid

➤ Grid-row:

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



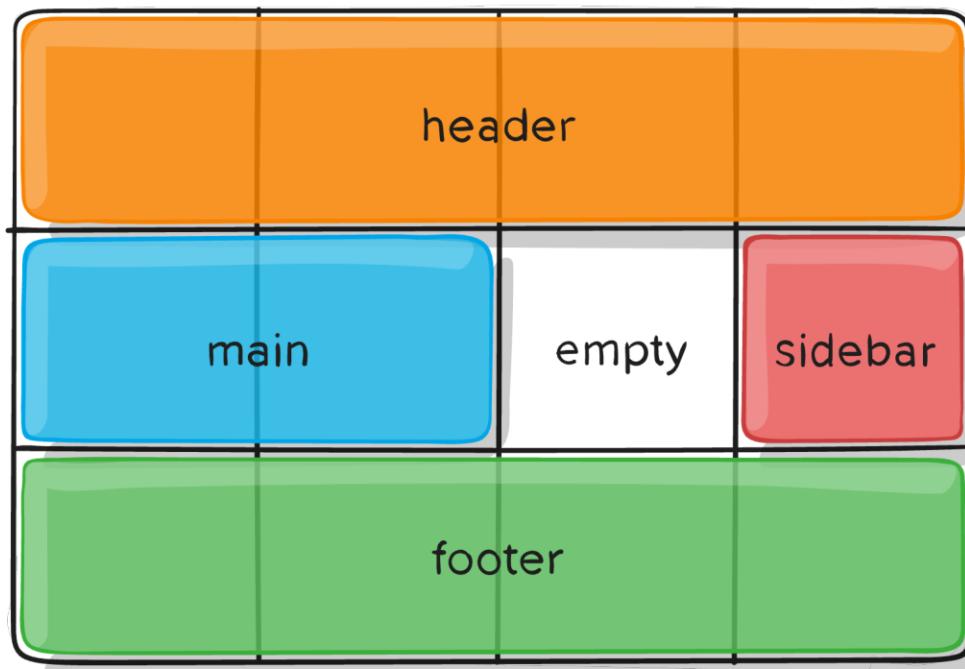
```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  grid-gap: 10px;  
  background-color: #gold;  
  padding: 10px;  
}  
  
.grid-container > div {  
  background-color: #rgba(255, 255, 255, 0.8);  
  text-align: center;  
  padding: 20px 0;  
  font-size: 30px;  
}  
  
.item1 {  
  grid-row: 1 / span 2;  
}
```

Các thuộc tính của Grid

➤ Naming Grid Item:

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

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



So sánh Float, Flexbox, Grid

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

So sánh Float, Flexbox, Grid

➤ Browser Support cho flex:

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

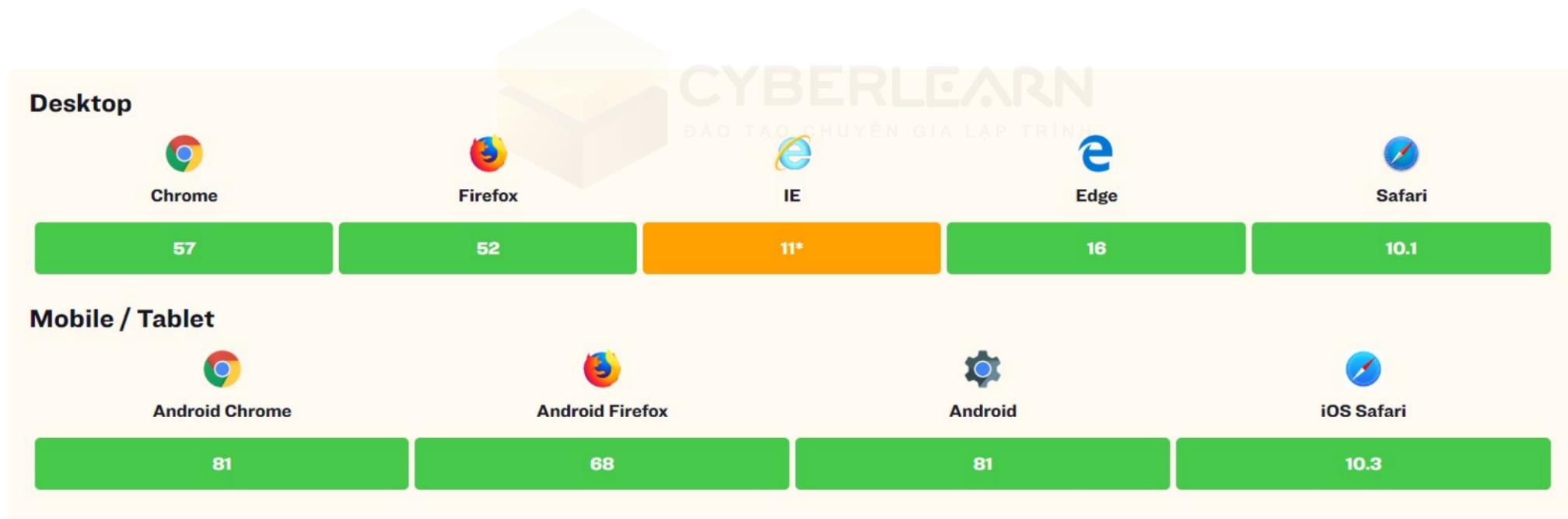


Chrome	Safari	Firefox	Opera	IE	Edge	Android	iOS
20- (old) 21+ (new)	3.1+ (old) 6.1+ (new)	2-21 (old) 22+ (new)	12.1+ (new)	10 (tweener) 11+ (new)	17+ (new)	2.1+ (old) 4.4+ (new)	3.2+ (old) 7.1+ (new)

So sánh Float, Flexbox, Grid

➤ Browser Support cho grid:

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





Animation

CSS3 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

CSS3 Animation

- Tên hiệu ứng được đặt thoải mái.
- Phải có duration.
- Có thể chỉnh theo from...to hoặc theo %

```
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 }
```

```
.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

➤ Animation delay

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

CSS3 Animation

➤ 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

➤ 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;

CSS3 Animation

➤ Animation direction

- Hướng chuyển động của animation
- Cú pháp: animation-direction: [Types] ;
- Ví dụ :
 - animation-direction : normal;
 - animation-direction : reverse;
 - animation-direction : alternate;
 - animation-direction : alternate-reverse;

Loại	Mô tả
normal	Giá trị mặc định, chuyển động theo hướng được định nghĩa trong keyframe
reverse	Chuyển động theo chiều ngược với hướng được định nghĩa trong keyframe
alternate	Chuyển động theo hướng cùng chiều keyframe sau đó đi chiều ngược lại và dừng ở trạng thái đầu tiên
alternate-reverse	Chuyển động theo hướng ngược chiều keyframe sau đó đi cùng chiều rồi dừng lại ở trạng thái cuối cùng của keyframe

CSS3 Animation

➤ Animation fill-mode

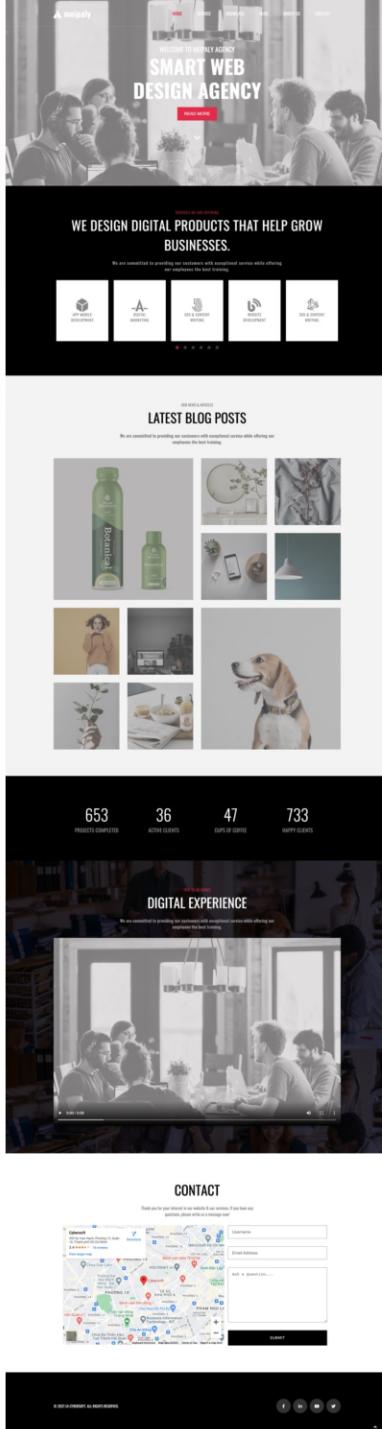
- Định nghĩa css của thẻ trước hoặc sau khi animation chạy hoặc cả hai
- Cú pháp: animation-fill-mode: [Types] ;
- Ví dụ :
 - animation-fill-mode : none;
 - animation-fill-mode : forwards;
 - animation-fill-mode : backwards;
 - animation-fill-mode : both;

Loại	Mô tả
none	Giá trị mặc định, chuyển động theo css được định nghĩa trong keyframe
forwards	Thẻ sẽ duy trì css của keyframe cuối cùng sau khi animation kết thúc
backwards	Thẻ sẽ lấy css của keyframe đầu tiên sau đó mới bắt đầu animation
both	Thẻ sẽ lấy css của keyframe đầu tiên sau đó mới bắt đầu animation và duy trì css của keyframe cuối cùng sau khi animation kết thúc

02

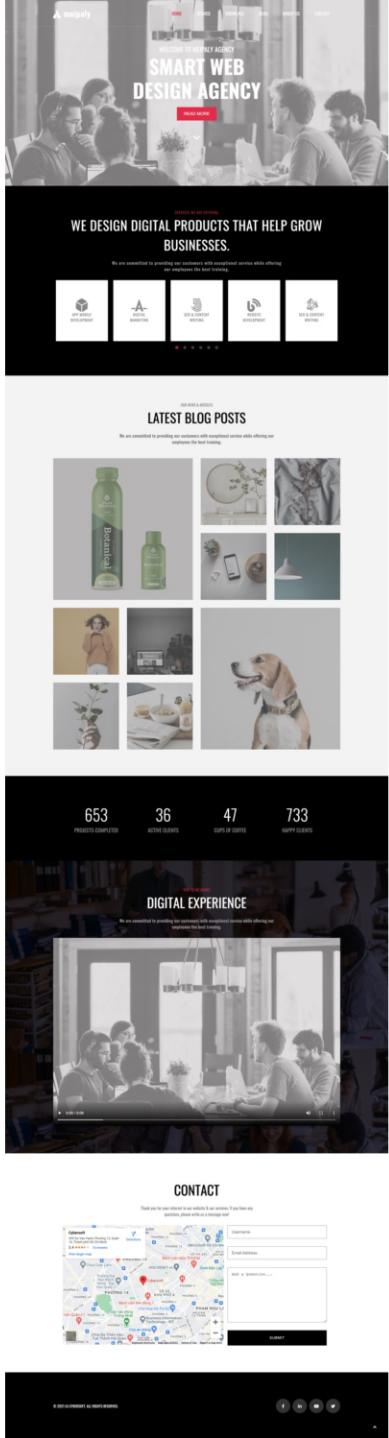


MEIPALY LAYOUT



Xây dựng Meipaly Layout

- 01 Tổ chức thư mục
- 02 Phân chia bố cục layout
- 03 Tạo cấu trúc trang
- 04 Xử lý phần Header
- 05 Xử lý phần Carousel
- 06 Xử lý phần Service
- 07 Xử lý phần Blog
- 08 Xử lý phần Number
- 09 Xử lý phần Experience
- 10 Xử lý phần Contact
- 11 Xử lý phần Footer

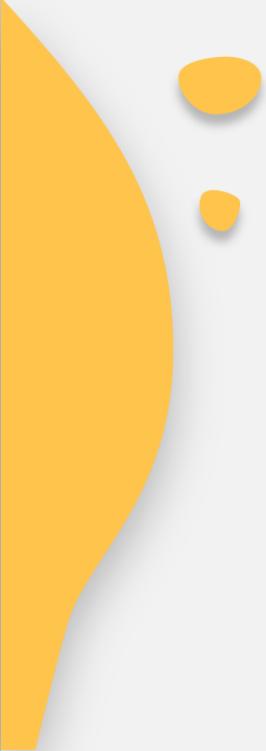


Xây dựng Meipaly Layout

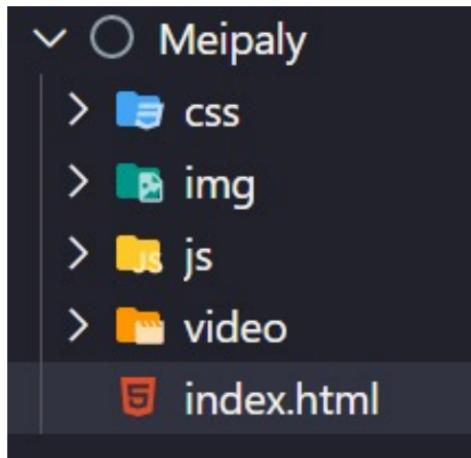
Nhấn vào đây để tải hình



Tổ chức thư mục



Tổ chức thư mục



- Folder dự án sẽ chứa:
 - index.html: Trang chủ của website
 - css: chứa các file css
 - img: chứa hình
 - js : chứa file javascript
 - video: chứa file video
- Do layout này sử dụng external css nên cần thêm đường dẫn tới file index.css vào thẻ head

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Meipaly</title>

        <!-- MAIN CSS -->
        <link rel="stylesheet" href=".//css/index.css">
</head>
```



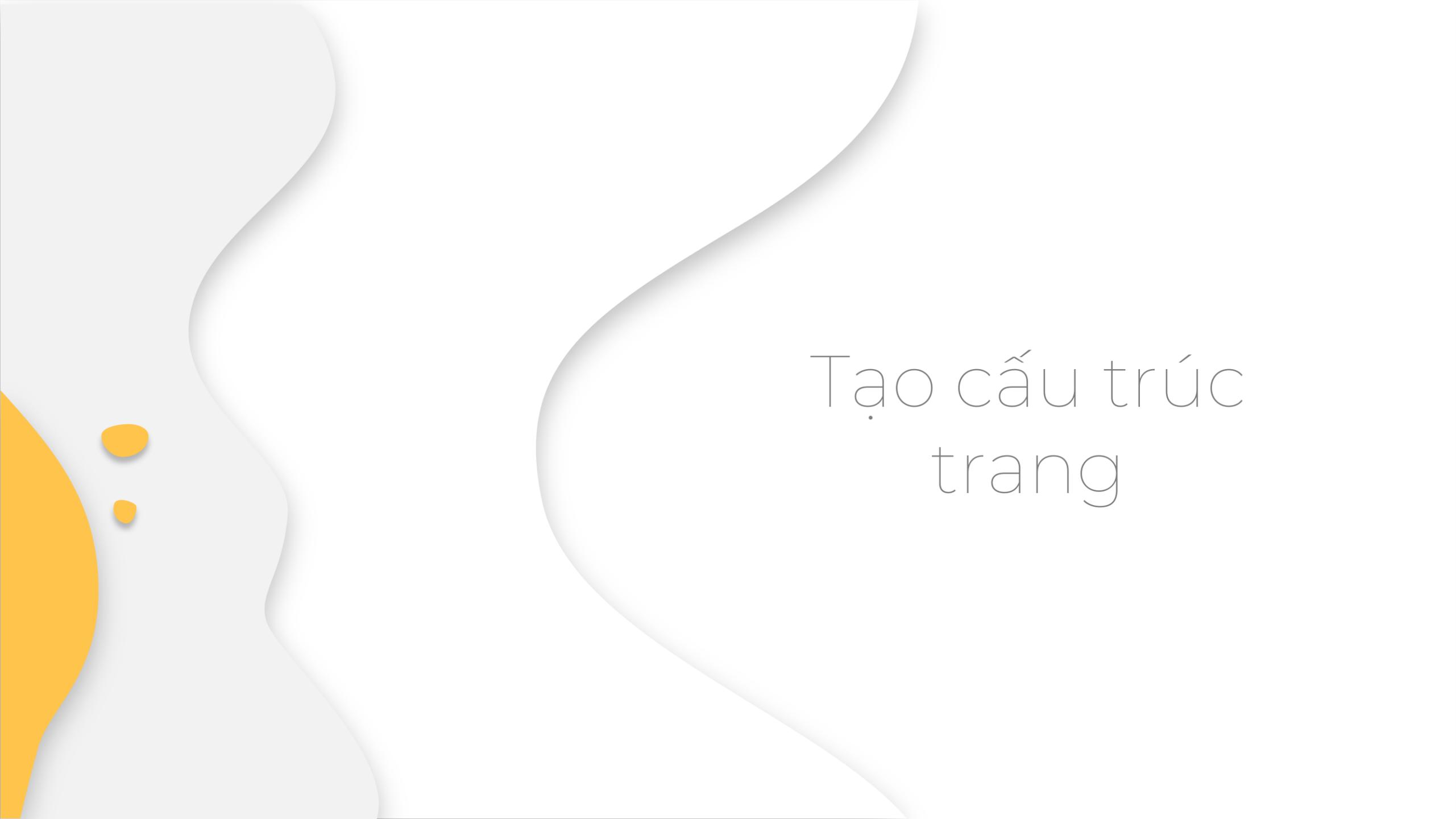
Phân chia bố
cục layout

Phân chia bố cục layout

```
<header>...
</header>
<section class="carousel">...
</section>
<section class="service">...
</section>
<section class="blog">...
</section>
<section class="number">...
</section>
<section class="experience"> ...
</section>
<section class="contact">...
</section>
<footer>...
</footer>
```

Layout có 8 phần chính:

- 2 phần chính luôn xuất hiện ở các website là header và footer
- Các phần chính ở giữa chúng ta chia theo nội dung của từng phần. Các phần này tạo bằng thẻ section và đặt tên class theo nội dung của phần đó để dễ quản lý và thêm css.



Tạo cấu trúc
trang

Tạo cấu trúc trang

- Cài đặt thư viện Font Awesome
- Cài đặt thư viện font chữ cho website
- Các file thư viện phải nằm trên file index.css
- Khai báo các thiết lập CSS chung cho cả website



```
/* GLOBAL */
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

html,
body {
    font-family: "Oswald", sans-serif;
    color: #white;
}
```

```
<!-- GOOGLE FONT -->
<link rel="preconnect" href="https://fonts.gstatic.com">
<link href="https://fonts.googleapis.com/css2?family=Oswald:wght@300;400;700&display=swap" rel="stylesheet">

<!-- FONT AWESOME -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css"
      integrity="sha512-iBBXm8fW90+nuLcSKlbmrPcLa0OT92x01BIzZ+ywDWZCvqsWgccV3gFoRBv0z+8dLJgyAHIhR35VZc2oM/gI1w=="
      crossorigin="anonymous" />
<!-- MAIN CSS -->
<link rel="stylesheet" href=".//css/index.css">
```



Xử lý phần
Header

Xử lý phần Header



```
<header>
  <div class="header__content">
    <a href="" class="header__logo">
      <i class="fab fa-atlassian"></i>
      <span>meipaly</span>
    </a>
    <nav>
      <ul>
        <li class="active"><a href="#">HOME</a></li>
        <li><a href="#">SERVICE</a></li>
        <li><a href="#">showcase</a></li>
        <li><a href="#">blog</a></li>
        <li><a href="#">ABOUT US</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 a chứa chữ logo, nav chứa phần navigation bar

Xử lý phần Header

```
.header__content ul {  
    list-style: none;  
}  
  
.header__content li {  
    display: inline;  
    padding: 0 26px;  
}  
  
.header__content a {  
    color: white;  
    text-decoration: none;  
}  
  
.header__content li a {  
    text-transform: uppercase;  
    font-size: 17px;  
    transition: all 0.5s;  
}  
  
.header__content li a:hover {  
    color: #e62b4a;  
}  
  
.header__content li.active a {  
    color: #e62b4a;  
}
```

- 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
 - Xử lý các mục (thẻ li) của navbar nằm trên 1 hàng
 - Thêm khoảng cách

Dùng transition để khi di chuyển chuột vào thẻ a, việc thay đổi màu của thẻ a sẽ xảy ra chậm

Sử dụng :hover để tạo hiệu ứng thay đổi màu khi di chuyển chuột vào thẻ a

Xử lý phần Header



➤ 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,  
body {  
    font-family: "Oswald", sans-serif;  
    color: #white;  
    /* font-size: 18px; */  
    /* Nếu không khai báo font-size, mặc định 1rem=16px */  
}
```

```
.header__Logo {  
    font-size: 2rem;  
    font-weight: 700;  
}
```

Xử lý phần Header



- Xử lý phần logo và navbar nằm trên 1 hàng:

```
.header__content {  
    width: 75%;  
    margin: 0 auto;  
  
    display: flex;  
    justify-content: space-between;  
    align-items: center;  
}
```

Dùng width < 100% và margin auto
giúp căn nội dung ở giữa

Áp dụng kiến thức của Flexbox

Xử lý phần Header

➤ Code xử lý đầy đủ của Header

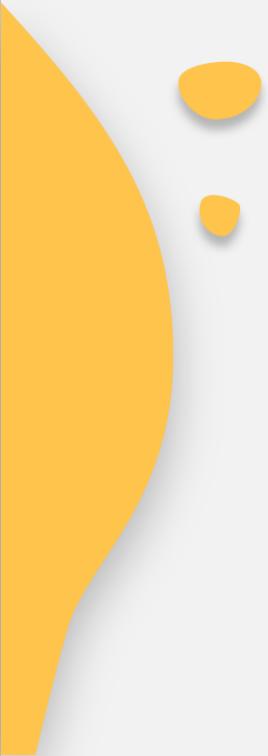
```
✓ header {  
    background-color: transparent;  
    border-bottom: 1px solid □rgba(255, 255, 255, 0.2);  
    padding: 26px 0;  
    position: absolute;  
    width: 100%;  
    z-index: 10;  
}  
✓ .header__content {  
    width: 75%;  
    margin: 0 auto;  
  
    display: flex;  
    justify-content: space-between;  
    align-items: center;  
}  
✓ .header__Logo {  
    font-size: 2rem;  
    font-weight: 700;  
}  
  
✓ .header__content ul {  
    list-style: none;  
}
```

Xử lý để header nằm đè lên carousel:

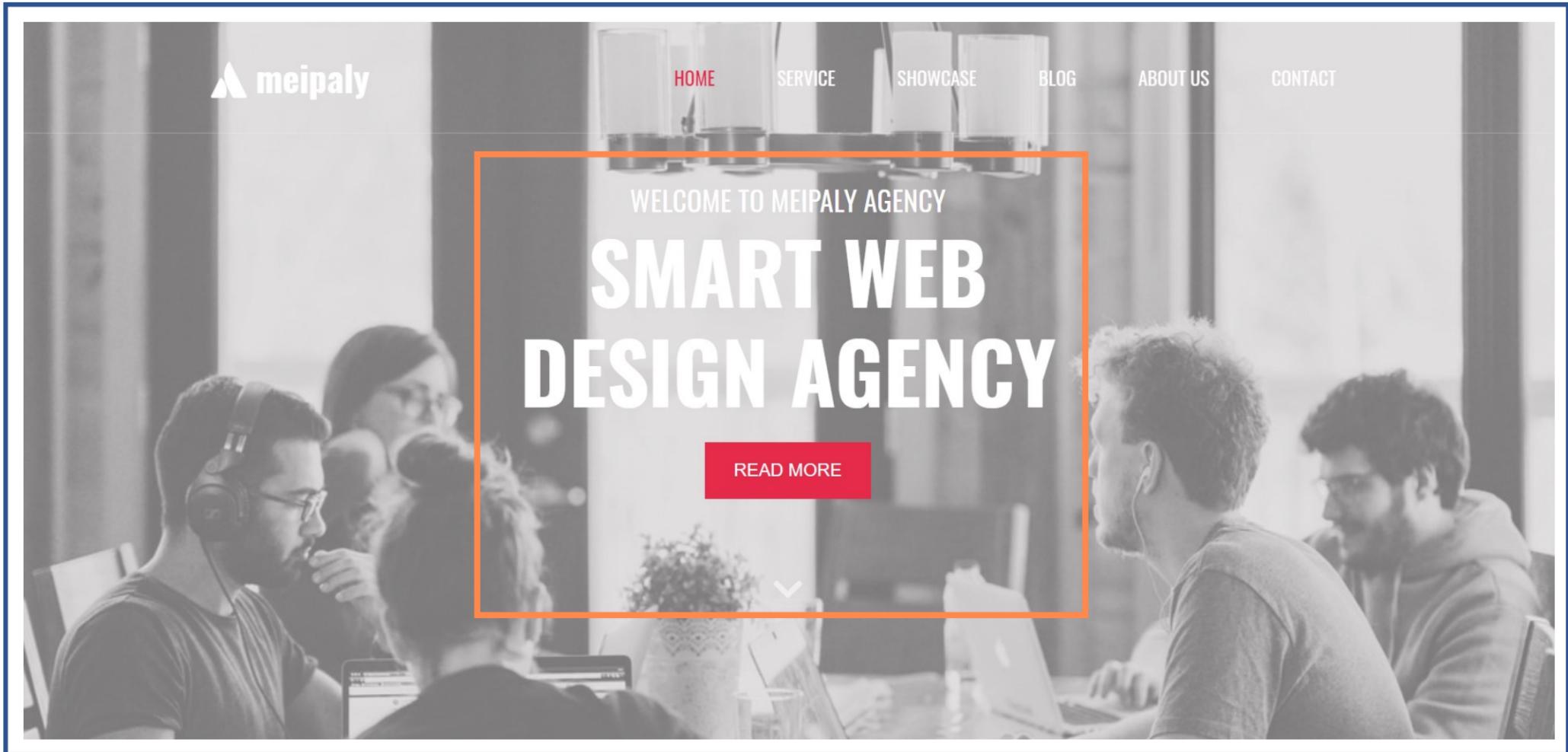
- Khi thêm absolute thẻ sẽ bị mất kích thước nên width:100% để xét lại kích thước.
- z-index: để header không bị carousel che

```
.header__content li {  
    display: inline;  
    padding: 0 26px;  
}  
.header__content a {  
    color: □white;  
    text-decoration: none;  
}  
.header__content li a {  
    text-transform: uppercase;  
    font-size: 17px;  
    transition: all 0.5s;  
}  
.header__content li a:hover {  
    color: □#e62b4a;  
}  
.header__content li.active a {  
    color: □#e62b4a;  
}
```

Xử lý phần Carousel



Xử lý phần Carousel



Xử lý phần Carousel

```
<section class="carousel">
  <div class="carousel__content">
    <p>WELCOME TO MEIPALY AGENCY</p>
    <h1>SMART WEB<br>DESIGN AGENCY</h1>
    <button>READ MORE</button>
    <div class="carousel__scroll">
      <i class="fa fa-chevron-down"></i>
    </div>
  </div>
</section>
```

YBERLARON

TAO CHUYÊN GIA LẬP TRÌNH

➤ Xây dựng HTML:

- Phần Carousel sẽ có 1 thẻ con trực tiếp.
- Bên trong div con carousel__content có các thẻ con là p,h1,button và carousel__scroll chứa icon mũi tên

Xử lý phần Carousel

➤ Xử lý CSS

```
/* CAROUSEL */  
carousel {  
background-image: url(..../img/  
home_slider.jpg);  
background-size: cover;  
background-position: center;  
height: 650px;  
  
display: flex;  
justify-content: center;  
align-items: center;  
}  
}
```

Áp dụng lý thuyết background của
css để thêm hình nền.
Xét height để tăng kích thước khung
hình nền.

Khi thẻ cha có display:flex chỉ chứa 1
thẻ con cấp 1 và kết hợp thêm justify-
content, align-items center sẽ giúp
căn thẻ con đó ở giữa thẻ cha

Xử lý phần Carousel

```
.carousel__content {  
    text-align: center;  
    color: white;  
}  
  
.carousel__content p {  
    font-size: 25px;  
}  
.carousel__content h1 {  
    font-size: 75px;  
    line-height: 90px;  
}  
  
.carousel__content button {  
    font-size: 16px;  
    margin-top: 20px;  
    padding: 15px 25px;  
    background-color: #e62b4a;  
    color: white;  
    border: 1px solid #e62b4a;  
    cursor: pointer;  
    transition: all 0.5s;  
}  
  
.carousel__content button:hover {  
    background-color: #af2239;  
}
```

Xét màu chữ chung cho cả phần nội dung
Chỉnh căn chữ ở giữa

Chỉnh riêng font-size và khoảng cách
giữa các dòng chữ cho thẻ p, h1

Thiết kế button Read More

Xử lý phần Carousel

```
.carousel__scroll i{  
    margin-top: 50px;  
    font-size: 30px;  
    animation: arrowDown 1s linear 0s infinite;  
}
```

Sử dụng animation để tạo chuyển động cho icon arrow

```
@keyframes arrowDown{  
    from{  
        opacity: 1;  
        transform: translateY(0);  
    }  
  
    to{  
        opacity: 0;  
        transform: translateY(40px);  
    }  
}
```

BERLEARN
KHO CHUYÊN GIA LẬP TRÌNH



Xử lý phần
Service

Xử lý phần Service

SERVICES WE ARE OFFERING

WE DESIGN DIGITAL PRODUCTS THAT HELP GROW BUSINESSES.

We are committed to providing our customers with exceptional service while offering
our employees the best training.

SEO & CONTENT
WRITING

WEBSITE
DEVELOPMENT

SEO & CONTENT
WRITING

APP MOBILE
DEVELOPMENT

APP MOBILE
DEVELOPMENT

Xử lý phần Service

```
<section class="service">
    <div class="service__content">
        <div class="title">...
        </div>
        <div class="service__slider">
        </div>
    </div>
</section>
```

➤ Xây dựng HTML:

- Phần Services sẽ có 1 thẻ con trực tiếp.
- Bên trong div con service__content có các 2 thẻ con: title (phần chữ tiêu đề), service__slider

Xử lý phần Service

➤ Xây dựng phần title:

- Các phần chính đều có thiết kế tiêu đề giống nhau nên tạo 1 class title để sử dụng chung

```
<section class="service">
  <div class="service_content">
    <div class="title">
      <p>SERVICES WE ARE OFFERING</p>
      <h2>WE DESIGN DIGITAL PRODUCTS THAT
      HELP GROW BUSINESSES.</h2>
      <p>We are committed to providing our
      customers with exceptional service
      while offering
        <br>
        our employees the best training.
      </p>
    </div>
    <div class="service_slider">...
    </div>
  </div>
</section>
```

```
html,
body {
  font-family: "Oswald", sans-serif;
  color: #white;
  /* font-size: 18px; */
  /* Nếu không khai báo font-size, mặc
  định là 18px */
}

.title{
  text-align: center;
}
.title p:first-of-type {
  font-size: 14px;
  color: #e62b4a;
  font-weight: 300;
}
.title h2 {
  font-size: 50px;
  font-weight: 400;
}

.title p:last-of-type {
  font-size: 16px;
  color: #888888;
  font-weight: 700;
  margin-top: 20px;
}
```

Gọi thẻ p thứ nhất
của class title

Gọi thẻ p cuối cùng
của class title

Xử lý phần Service

➤ Xây dựng phần slider:

- Sử dụng thư viện để tạo hiệu ứng chuyển động cho slider:
<http://kenwheeler.github.io/slick/>
- Cài đặt thư viện:
 - Link css thêm trong thẻ <head> và để phía trên file index.css

```
<!-- SLICK -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/
slick-carousel/1.8.1/slick.min.css"
      integrity="sha512-yHknP1/AwR+yx26cB1y0cjvQUMvEa2PFzt1c9LlS4pRQ5NOTZFbhBig
      +X9G9eYW/8m0/40XNx8pxJ6z57x0dw=="
      crossorigin="anonymous" />
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/
slick-carousel/1.8.1/slick-theme.min.css"
      integrity="sha512-17EgCFERpgZKcm0j0fEq1YCJuAyWdz9KUtv1EjVuaOz8pDnh/
      0nZxmU6BBXwaaxqoi9PQXnRWqlcDB027hgv9A=="
      crossorigin="anonymous" />

<!-- MAIN CSS -->
<link rel="stylesheet" href=".css/index.css">
```

Xử lý phần Service

- Link javascript để cuối file html trước thẻ đóng của thẻ </body>

```
<script src="https://code.jquery.com/jquery-3.6.0.min.js"
       integrity="sha256-/xUj+3OJU5yExlq6GSYGShk7tPXikynS7ogEvDej/m4=" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/slick-carousel/1.8.1/slick.min.js"
       integrity="sha512-XtmMtDEcNz2j7ekrtHv0VR4iwwaD6o/FUJe6+Zq+HgcCsk3kj4uSQQR8weQ2QVj1o0Pk6PwYLohm206ZzNfubg=="
       crossorigin="anonymous"></script>
<script>
    // SLICK
    $('.slick-content').slick({
        infinite: true,
        slidesToShow: 5,
        autoplay: true,
        autoplaySpeed: 4000,
        speed: 1100,
        dots: true

    });
</script>
</body>
```

Xử lý phần Service

- Xây dựng html cho slider:

```
<section class="service">
    <div class="service__content">
        <div class="title">...
        </div>
        <div class="service__slider">
            <div class="slick-content">
                <div class="item">
                    <i class="fa fa-dice-d6"></i>
                    <p>
                        APP MOBILE<br>DEVELOPMENT
                    </p>
                </div>
                <div class="item">
                    <i class="fab fa-autoprefixer"></i>
                    <p>
                        DIGITAL<br> MARKETING
                    </p>
                </div>
                <div class="item">
                    <i class="fab fa-500px"></i>
                    <p>
                        SEO &amp; CONTENT <br>WRITING
                    </p>
                </div>
            </div>
        </div>
    </div>
</section>
```

```
<div class="item">
    <i class="fa fa-blog"></i>
    <p>
        WEBSITE<br>DEVELOPMENT
    </p>
</div>
<div class="item">
    <i class="fab fa-battle-net"></i>
    <p>
        SEO &amp; CONTENT <br>WRITING
    </p>
</div>
<div class="item">
    <i class="fa fa-memory"></i>
    <p>
        APP MOBILE<br>DEVELOPMENT
    </p>
</div>
</div>
</div>
</section>
```

Xử lý phần Service

- Xây dựng css cho slider:

```
/* SERVICE */  
.service {  
    text-align: center;  
    padding: 90px 0;  
    background-color: black;  
}  
.service_content {  
    width: 75%;  
    margin: 0 auto;  
    letter-spacing: 1px;  
}  
.service_slider{  
    margin-top: 20px;  
}
```

Căn giữa nội dung bằng flex.
Nếu thẻ cha có nhiều
hơn 1 thẻ con thì
dùng flex-direction
để tránh làm các thẻ
con bị kéo lên 1 hàng

```
.service_slider .item {  
    width: 214px;  
    height: 230px;  
    /* border: 2px solid #81807d; */  
    background-color: white;  
    color: #848482;  
    text-align: center;  
    margin: 10px;  
  
    display: flex;  
    flex-direction: column;  
    justify-content: center;  
    align-items: center;  
  
    position: relative;  
}  
  
.service_slider .item i {  
    font-size: 50px;  
    color: #888888;  
}  
.service_slider .item p {  
    font-size: 14px;  
}
```

Xử lý phần Service

- Xây dựng css cho slider:

```
.service_slider .item::after{  
    content: '';  
    height: 0;  
    width: 5px;  
    background-color: #e62b4a;  
    display: block;  
  
    position: absolute;  
    top: 0;  
    right: 0;  
    transition: all .2s linear;  
}  
  
.service_slider .item:hover::after{  
    height: 100%;  
}
```

Sử dụng After để tạo đường line màu đỏ xuất hiện khi di chuyển chuột vào phần slider

Chỉnh sửa css phần dấu tròn dots của thư viện slick

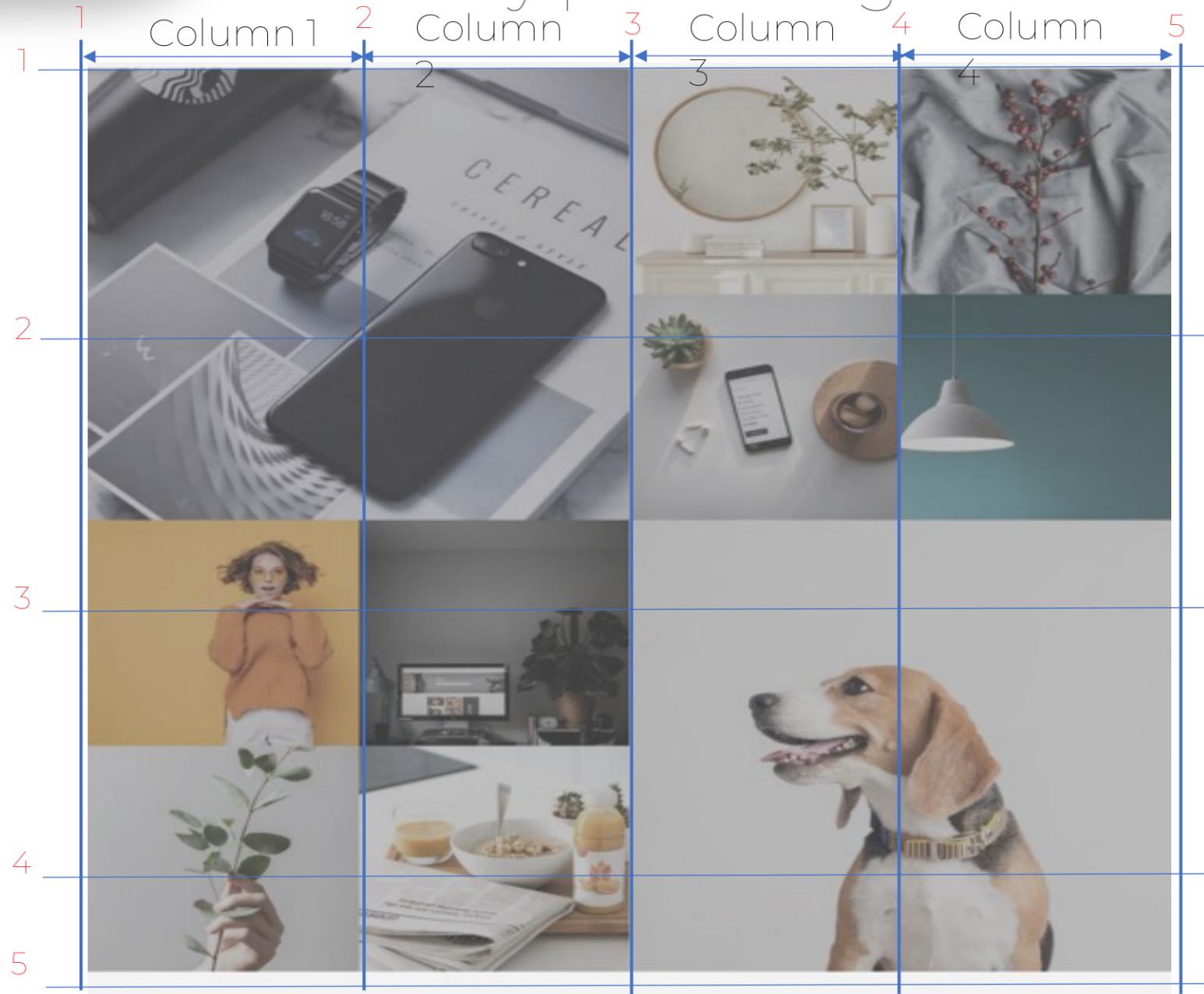
```
.slick-content .slick-dots li button:before{  
    color: #ffffff;  
    font-size: 14px;  
}  
  
.slick-content .slick-dots li.slick-active button:before{  
    color: #e62b4a;  
}
```

So sánh inline, block và inline-block

Display: Inline	Display: Block	Display: Inline-Block
<ul style="list-style-type: none">Các item sẽ nằm trên cùng một dòng, ví dụ như .Nếu các items vượt quá độ dài của dòng thì item sẽ xuống dòng mới	<ul style="list-style-type: none">Các item có kiểu luôn được xuống dòng và chiếm toàn bộ width của thẻ cha nếu width không được xét	<ul style="list-style-type: none">Các items sẽ được xếp cùng nhau trên một dòng giống kiểu inline
<ul style="list-style-type: none">Các item không thể xét width và height.	<ul style="list-style-type: none">Xét được width, height	<ul style="list-style-type: none">Xét được width, height giống của block
<ul style="list-style-type: none">Chỉ có thể điều chỉnh margin và padding left and right (top và bottom thì không thể).	<ul style="list-style-type: none">Xét được margin, padding đầy đủ 4 hướng (top, bottom, right, left).	<ul style="list-style-type: none">Xét được margin, padding đầy đủ 4 hướng (top, bottom, right, left).

Xử lý phần Blog

Xử lý phần Blog



Xử lý phần Blog

```
<section class="blog">
  <div class="title title--black">
    <p>OUR NEWS & ARTICLES</p>
    <h2>LATEST BLOG POSTS</h2>
    <p>We are committed to providing our
      customers with exceptional service while
      offering our
      <br>
      employees the best training.
    </p>
  </div>

  <div class="blog__content">...
  </div>
</section>
```

➤ Xây dựng HTML:

- Phần Blog sẽ có 2 thẻ con trực tiếp: title (phần chữ tiêu đề), blog_content
- Phần title xây dựng tương tự các phần trước

Xử lý phần Blog

➤ Xây dựng CSS cho Blog :

Áp dụng kiến thức Grid:
Thêm display:grid ở thẻ cha trực tiếp của
các thẻ con đang cần chỉnh bố cục

```
.blog__content{  
    display: grid;  
    grid-template-columns: repeat(4,1fr);  
    gap: 30px;  
  
    width:75%;  
    margin: 50px auto 0;  
}
```

```
<div class="blog__content">  
    <div class="blog__item blog__metro1">  
          
  
    </div>
```

Khai báo số lượng column của grid, kích thước
của mỗi column
gap: giúp thêm khoảng cách giữa các thẻ con
của grid

Xử lý phần Blog

- Xây dựng CSS cho Blog :

```
<div class="blog_content">  
  <div class="blog_item blog_metro1">...</div>  
  <div class="blog_item">...</div>  
  <div class="blog_item blog_metro10">...</div>  
</div>
```

Những thẻ con cần
gồm hàng hoặc cột sẽ
được thêm class riêng

```
.blog_metro1{  
  /* grid-column: 1 / 3; */  
  grid-column: 1 / span 2;  
  grid-row: 1 / span 2;  
}  
.blog_metro10{  
  grid-column: 3 / span 2;  
  grid-row: 3 / span 2;  
}
```

Áp dụng kiến thức grid
để gồm hàng và cột

Xử lý phần Blog

- Xử lý blog_item cho Blog:

```
.blog_item img{  
    width: 100%;  
    height: 100%;  
    filter: grayscale(50%) contrast(50%);  
    object-fit: cover;  
  
    transition: all .5s;  
}
```

Giúp chuyển ảnh màu sang
ảnh màu tối hơn

Tác dụng tương tự như
background-size:cover, giúp
hình vừa với thẻ cha và không
bị bóp méo khi xét hình width,
height:100%

Xử lý phần Blog

- Xây dựng HTML cho phần blog_overlay:

```
<div class="blog_content">
    <div class="blog_item blog_metro1">
        
        <div class="blog_overlay">
            <div class="blog_icons">
                <a href="../img/grid-metro-1.jpg" data-fancybox="gallery" data-caption="Caption #1">
                    <i class="fa fa-search-plus"></i>
                </a>
            </div>
            <div class="blog_text">
                <p>LOGO</p>
                <h3>FLIPPIN BIRD</h3>
            </div>
        </div>
    </div>
</div>
```

Xử lý phần Blog

- Xây dựng css phần blog_item:

```
.blog_overlay{  
    background-color: #rgba(230, 43, 74, 0.6);  
    position: absolute;  
    top: 0;  
    left: 0;  
    width: 100%;  
    height: 100%;  
    padding: 25px;  
    opacity: 0;  
    transition: all .5s;  
}  
  
.blog_item:hover .blog_overlay{  
    opacity: 1;  
}
```

Dùng kỹ thuật position để overlay nằm đè lên phần hình (nhớ thêm position relative cho thẻ cha)

```
.blog_item{  
    position: relative;  
    overflow: hidden;  
}
```

Xử lý hiệu ứng ẩn hiện khi hover (di chuyển chuột)

Xử lý phần Blog

- Xây dựng css phần blog_item:

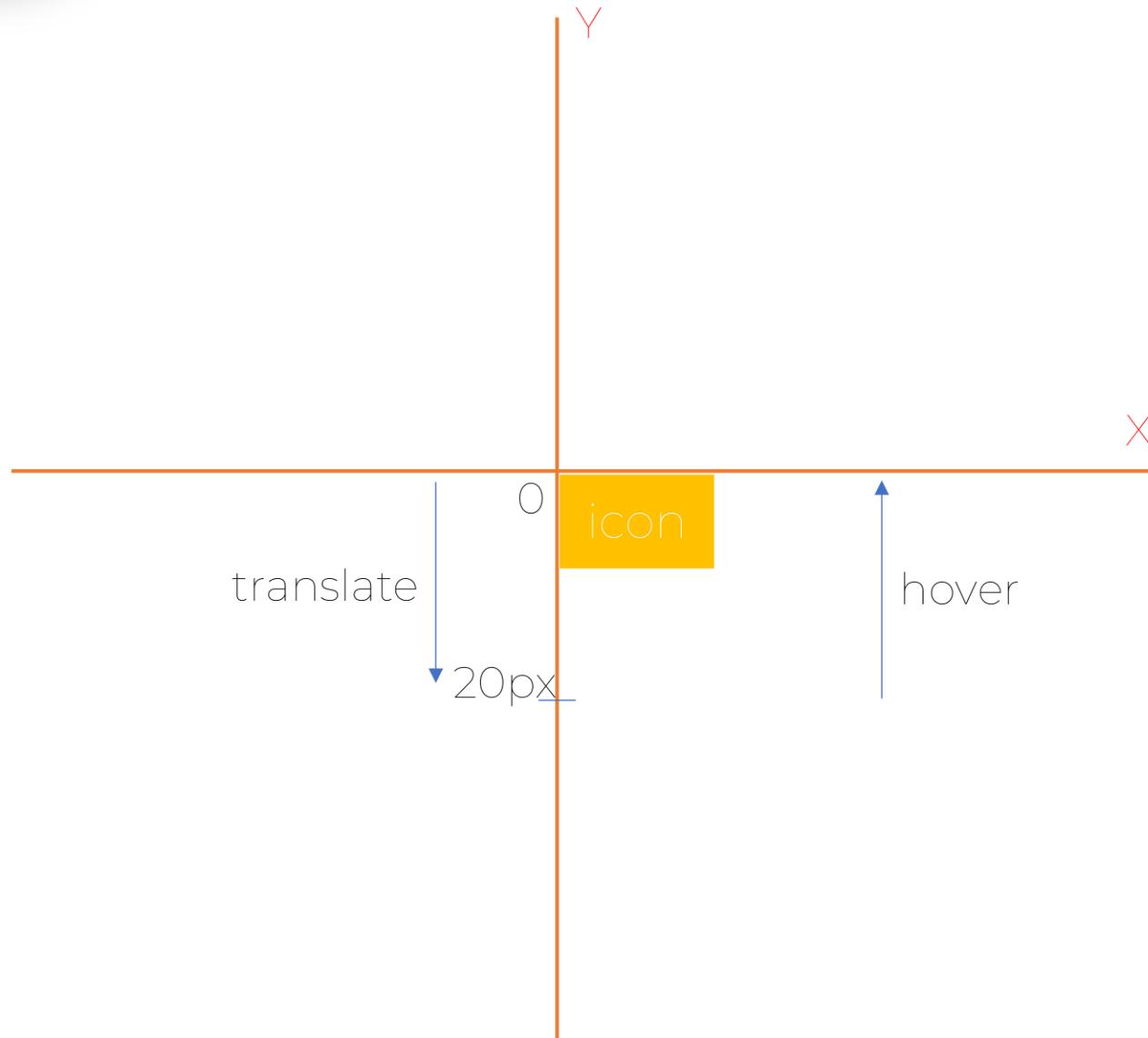
```
.blog_icons{  
    position: absolute;  
    top: 25px;  
    right: 25px;  
}  
  
.blog_icons i{  
    width: 35px;  
    height: 35px;  
    line-height: 35px;  
    text-align: center;  
    background-color: #ffffff;  
    font-size: 14px;  
    color: #000000;  
    margin-bottom: 10px;  
    transform: translateY(20px);  
    transition: all .8s;  
}  
  
.blog_item:hover .blog_icons i{  
    transform: translateY(0);  
}  
  
.blog_icons i:hover{  
    background-color: #000000;  
    color: #ffffff;  
}
```

Dùng kỹ thuật position để phần icon nằm bên góc trên bên phải



Áp dụng transform để tạo hiệu ứng phần icon di chuyển từ dưới lên khi hover

Xử lý phần Blog



Xử lý phần Showcase

- Xây dựng css phần text của blog_item:

```
.blog_text{  
    position: absolute;  
    left: 25px;  
    bottom: 25px;  
    transform: translateY(20px);  
    transition: all .8s;  
}  
  
.blog_text p{  
    color: #D0C7C8;  
}  
  
.blog_item:hover .blog_text{  
    transform: translateY(0);  
}
```

Dùng kỹ thuật position để phần text nằm bên góc dưới bên trái



Áp dụng transform để tạo hiệu ứng phần text di chuyển từ dưới lên khi hover

Xử lý phần Blog

➤ Xây dựng phần hiệu ứng popup cho Blog:

- Sử dụng thư viện:

<https://fancyapps.com/fancybox/3/>

Cài đặt thư viện:

- Link css thêm trong thẻ <head> và để phía trên file index.css
- Link javascript để cuối file html trước thẻ đóng của thẻ </body>

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/fancybox/3.5.7/jquery.fancybox.min.css"
      integrity="sha512-H9jrZiiopUdsLpg94A333EfumgUBpO9MdbxStdeITo+KEIMaNfHNvwyjjDJb+ERPaRS6DpyR1KbvPUasNItRyw=="
      crossorigin="anonymous" />
<!-- MAIN CSS --&gt;
&lt;link rel="stylesheet" href="./css/index.css"&gt;
&lt;/head&gt;</pre>
```

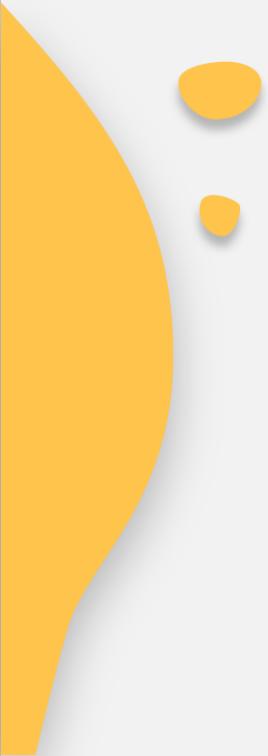
```
<script src="https://cdnjs.cloudflare.com/ajax/libs/fancybox/3.5.7/jquery.fancybox.min.js"
      integrity="sha512-uURL+ZXMBrF4AwGaWmEetzrd+J5/8NRkWAvJx5sbPSSuOb0bZLqf+tOzniOb000BjHa/dD7gub9oCGMLPQHtQA==" 
      crossorigin="anonymous"></script>
<script>
    // FANCYBOX
    $('[data-fancybox="gallery"]').fancybox({
        // Options will go here
        loop: true
    });
</script>
```

Xử lý phần Blog

➤ Xây dựng phần hiệu ứng popup cho Blog:

- Thêm các cài đặt của thư viện vào thẻ a chứa đường url đến phần hình cần hiện lên popup

```
<div class="blog_icons">
    <a href="../img/grid-metro-1.jpg" data-fancybox="gallery" data-caption="Caption #1">
        <i class="fa fa-search-plus"></i>
    </a>
</div>
```



Xử lý phần
Number

Xử lý phần Number

8,705

PROJECTS COMPLETED

480

ACTIVE CLIENTS

625

CUPS OF COFFEE

9,774

HAPPY CLIENTS

```
<section class="number">
  <div class="number__content">
    <div class="number__item">
      <p class="counter">8,705</p>
      <p>PROJECTS COMPLETED</p>
    </div>
    <div class="number__item">
      <p class="counter">480</p>
      <p>ACTIVE CLIENTS</p>
    </div>
    <div class="number__item">
      <p class="counter">625</p>
      <p>CUPS OF COFFEE</p>
    </div>
    <div class="number__item">
      <p class="counter">9,774</p>
      <p>HAPPY CLIENTS</p>
    </div>
  </div>
</section>
```

```
/* NUMBER */
.number{
  background-color: #fff;
  padding: 100px 0 220px;
}

.number__content{
  width: 70%;
  margin: 0 auto;
  display: flex;
  justify-content: space-between;
  text-align: center;
}

.number__item{
  width: 25%;
  border-right: 3px solid #888888;
}

.number__item:last-of-type{
  border-right: none;
}

.number__item p:first-of-type{
  font-size: 60px;
  font-weight: 300;
}

.number__item p:last-of-type{
  font-size: 20px;
  color: #888888;
}
```

Xử lý phần Number

➤ Thêm hiệu ứng counting :

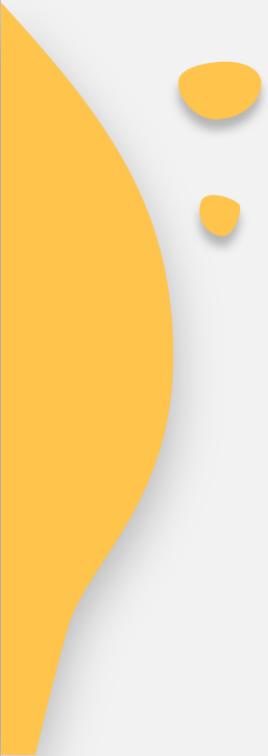
- Sử dụng thư viện:

<https://www.jqueryscript.net/other/Counting-Up-To-Numerical-Values-On-Scroll-jQuery-Countup-js.html>

- Cài đặt thư viện:

Link javascript để cuối file html trước thẻ đóng của thẻ </body>

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/waypoints/4.0.1/jquery.waypoints.js"
       integrity="sha512-ZKNVEa7gi0Dz4Rq9jXcySgcPiK+5f01CqW+ZoKLLKr9VMXuCsw3RjWiv8ZpIOa0hx079np7Ec8DDWALM0bD0aQ=="
       crossorigin="anonymous"></script>
<script src=".js/jquery.countup.js"></script>
<script>
    // COUNT UP
    $('.counter').countUp();
</script>
```



Xử lý phần
Experience

Xử lý phần Experience

```
<section class="experience">
  <div class="title">
    <p>HOW DO WE WORKS</p>
    <h2>DIGITAL EXPERIENCE</h2>
    <p>We are committed to providing our customers with exceptional service
      while offering our
      <br>
      employees the best training.
    </p>
  </div>
  <div class="experience_content">
    <video controls poster="./img/home_slider.jpg">
      <source src="./video/video.mp4" type="video/mp4">
      Your browser doesn't support HTML5 video tag.
    </video>
  </div>
</section>
```

- Xây dựng HTML:
- Phần Experience sẽ có 2 thẻ con trực tiếp: title, experience_content
 - Sử dụng thẻ video để hiển thị video lên trình duyệt

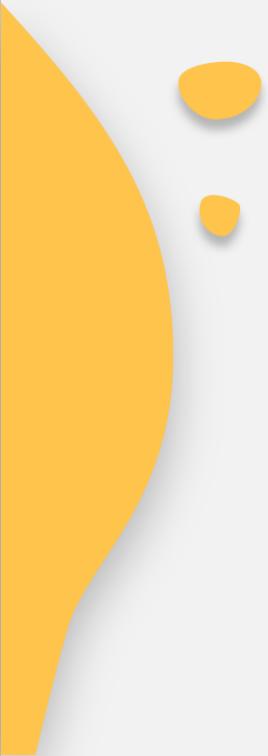
Xử lý phần Experience

```
/* EXPERIENCE */  
.experience{  
    /* background-color: #F3F3F3; */  
    background-image: url(..../img/bussiness_img_1.jpg);  
    background-size: cover;  
    background-attachment: fixed;  
    padding: 100px 0;  
}  
  
.experience__content{  
    width:75%;  
    margin:30px auto 0;  
}  
  
.experience__content video{  
    width:100%;  
    object-fit: cover;  
    border: none;  
    box-shadow: 0 2px 10px 1px □rgb(0,0,0,0.5);  
    outline: none;  
}
```

Background-attachment:fixed giúp hình nền cố định vào trang web và tạo cảm giác phần nội dung đang trượt trên hình nền

Tạo phần bóng mờ xung quanh video

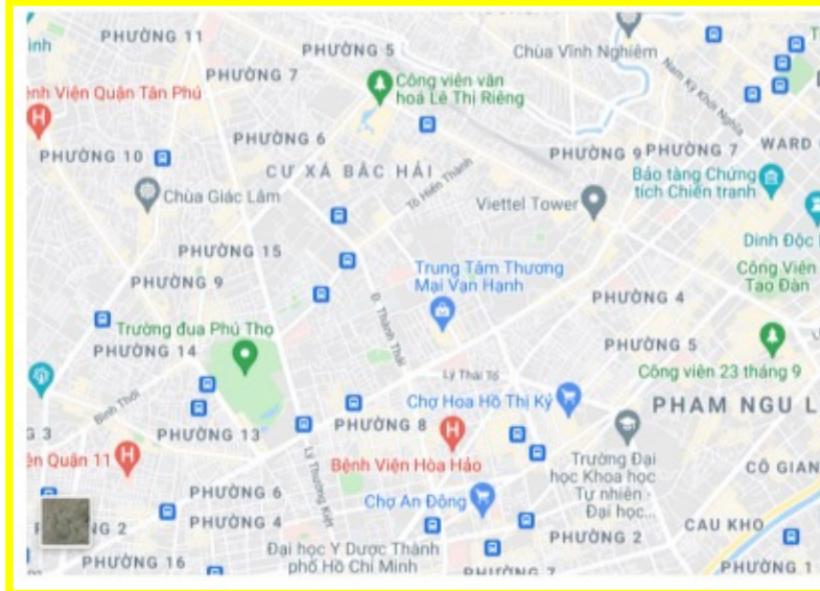
Xử lý phần Contact



Xử lý phần Contact

CONTACT

Thank you for your interest in our website & our services. If you have any questions, please write us a message now!



A map of District 11, Ho Chi Minh City, Vietnam. The map shows the district's boundaries and 16 wards (PHƯỜNG 1 through PHƯỜNG 16). Key landmarks labeled include: Công viên văn hóa Lê Thị Riêng, Chùa Vĩnh Nghiêm, Bảo tàng Chính kịch Chiến tranh, Dinh Độc Lập, Công Viên Tao Đàn, Trung Tâm Thương Mại Vạn Hạnh, Chợ Hoa Hồ Thị Kỷ, Chợ An Đông, Bệnh Viện Hoà Hảo, Đại học Y Dược Thành phố Hồ Chí Minh, and Viettel Tower. A yellow box highlights the map area.

Username:

Email Address:

Ask a Question...

Xử lý phần Contact

```
<section class="contact">
  <div class="title title--black">

    <h2>CONTACT</h2>
    <p>Thank you for your interest in our website & our services. If you have any
       questions, please write
       us a message now!
    </p>
  </div>
  <div class="contact__content">
    <div class="contact__map">
      <iframe
        src="https://www.google.com/maps/embed?pb=!1m14!1m8!1m3!1d15677.
        980968126514!2d106.6686637!3d10.7733302!3m2!1i1024!2i768!4f13.
        1!3m3!1m2!1s0x0%3A0xb490b64157cb9dc1!2sCybersoft!5e0!3m2!1sen!2s!4v161
        8408657872!5m2!1sen!2s"
        width="600" height="450" style="border:0;" allowfullscreen=""
        loading="lazy"></iframe>
    </div>
    <div class="contact__form">
      <form action="">
        <input type="text" placeholder="Username">
        <input type="email" placeholder="Email Address">
        <textarea placeholder="Ask a Question..." name="" id="" cols="30"
          rows="10"></textarea>
        <button class="btnMei">SUBMIT</button>
      </form>
    </div>
  </div>
</section>
```

➤ Xây dựng HTML:

- Phần Contact sẽ có 2 thẻ con trực tiếp: title, contact_content
- Sử dụng thẻ iframe để hiển thị map lên trình duyệt
- Sử dụng các thẻ control của form để tạo form liên hệ

Xử lý phần Contact

```
/* CONTACT */  
.contact{  
    padding: 100px 0;  
}  
  
.contact__content{  
    width: 70%;  
    margin: 30px auto 0;  
    display: flex;  
    align-items: center;  
}  
  
.contact__map{  
    width: 60%;  
}  
  
.contact__map iframe{  
    width: 100%;  
}  
  
.contact__form{  
    width: 40%;  
}  
  
.contact__form form{  
    display: flex;  
    flex-direction: column;  
    padding: 0 15px;  
}  
  
.contact__form input, .contact__form textarea{  
    font-size: 15px;  
    padding: 1em;  
    border: 1px solid #888888;  
    border-radius: 2px;  
    margin-bottom: 30px;  
}
```

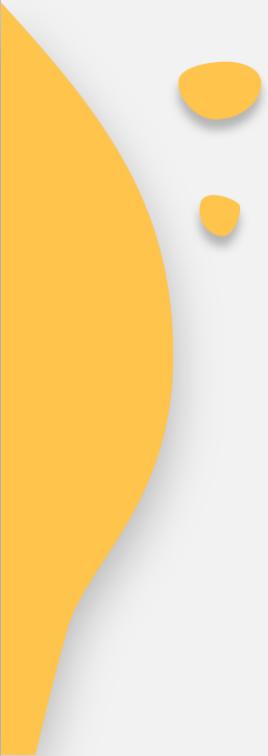
Dùng flex để map và form nằm cùng 1 hàng

Dùng flex-direction để các thẻ input nằm theo cột dọc

2 bộ selector có chung code css có thể viết rút gọn chung 1 hàng

Đơn vị em sẽ phụ thuộc font-size được thêm ở thẻ hiện tại

Xử lý phần Footer



Xử lý phần Footer



Xử lý phần Footer

```
<footer>
  <div class="footer_content">
    <p>© 2021 UI-CYBERSOFT. ALL RIGHTS RESERVED.</p>
    <div>
      <a href="#">
        <i class="fab fa-facebook-f"></i>
      </a>
      <a href="#">
        <i class="fab fa-linkedin-in"></i>
      </a>
      <a href="#">
        <i class="fab fa-youtube"></i>
      </a>
      <a href="#">
        <i class="fab fa-twitter"></i>
      </a>
    </div>
  </div>
</footer>
```

➤ Xây dựng HTML:

- Phần Footer sẽ có 1 thẻ con trực tiếp: footer_content
- footer_content sẽ chứa 2 thẻ con: p, div chứa các icons

Xử lý phần Footer

```
footer{  
    background-color: #000;  
}  
.footer__content{  
    width: 75%;  
    margin: 0 auto;  
    display: flex;  
    justify-content: space-between;  
    align-items: center;  
    padding: 100px 0;  
}  
  
.footer__content p{  
    font-size: 14px;  
}  
  
.footer__content a{  
    width: 54px;  
    height: 54px;  
    display: inline-block;  
    border-radius: 50%;  
    line-height: 54px;  
    text-align: center;  
    background-color: rgba(255, 255, 255, 0.2);  
    font-size: 16px;  
    margin-left: 5px;  
    text-decoration: none;  
    color: #ffffff;  
  
    transition: all .5s;  
}  
.footer__content a:hover{  
    background-color: #e62b4a;  
}
```

Dùng flex để phân chia
copyright và các icons nằm
cùng 1 hàng
Space-between giúp căn 2
thẻ con sát 2 bên trái, phải

Thẻ a là thẻ inline nên
không xét được kích
thước. Sử dụng inline-
block để xét được width,
height cho thẻ a và đồng
thời vẫn giữ các thẻ a
nằm cùng 1 hàng

Xử lý phần Footer

➤ Xây dựng nút back to top:

- Sử dụng thư viện:

<https://codyhouse.co/gem/back-to-top/>

Cài đặt thư viện:

- Link javascript để cuối file html trước thẻ đóng của thẻ </body>

```
<script src=".js/util.js"></script>
<script src=".js/backToTop.js"></script>
```

➤ Xây dựng HTML:

```
<a href="#" class="backToTop cd-top text-replace js-cd-top">
|   <i class="fa fa-angle-up"></i>
</a>
```

Xử lý phần Footer

```
/* BACKTOTOP */
.backToTop{
    width: 60px;
    height: 60px;
    line-height: 60px;
    border-radius: 4px;
    text-align: center;
    background: rgba(0,0,0,0.25);
    font-weight: bold;
    color: #fff;
    position: fixed;
    bottom: 20px;
    right: 20px;
    font-size: 20px;

    visibility: hidden;
    opacity: 0;
}

.cd-top--is-visible {
    visibility: visible;
    opacity: 1;
}
```

Position:fixed giúp thẻ di chuyển theo khi người dùng scroll web

Xử lý giúp nút back to top ẩn hiện khi scroll



Thank You