



**THỰC HỌC – THỰC NGHIỆP**

# **LẬP TRÌNH JAVASCRIPT**

**PRE-INTERMEDIATE JAVASCRIPT**

# HỆ THỐNG BÀI CŨ

---

- Event target property
- onChange & onBlur
- Key event
- Drag & Drop elements
- Form Submission.
- Pre-intermediate javascript: function & arguments object, js hoisting, strict mode,...

# MỤC TIÊU BÀI HỌC

---

- Upload Files
- Geolocation
- Media on page
- Giới thiệu Libraries & Frameworks



# PHẦN I

---

## PRE-INTERMEDIATE JAVASCRIPT

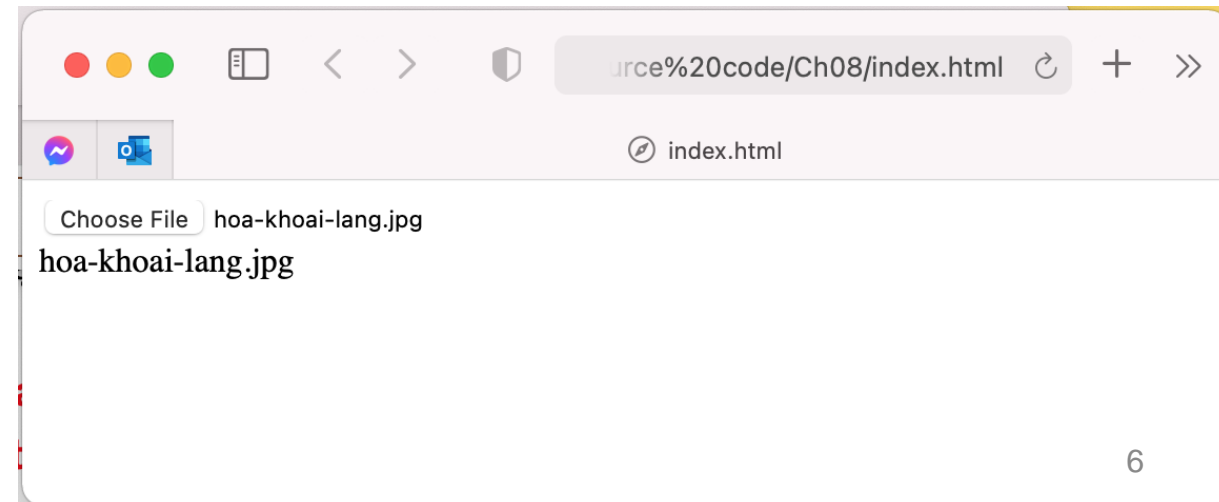
# Uploading files

---

- Chỉ ra tập tin cần upload bằng cách sử dụng input có type=file
- Sau khi chọn tập tin cần upload, javascript sẽ được kích hoạt để lấy được danh sách các tập tin.
- Các tập tin được biểu diễn dưới dạng đối tượng
- Các thuộc tính của đối tượng tập tin
  - name: tên tập tin
  - size: kích thước tập tin
  - type: loại tập tin

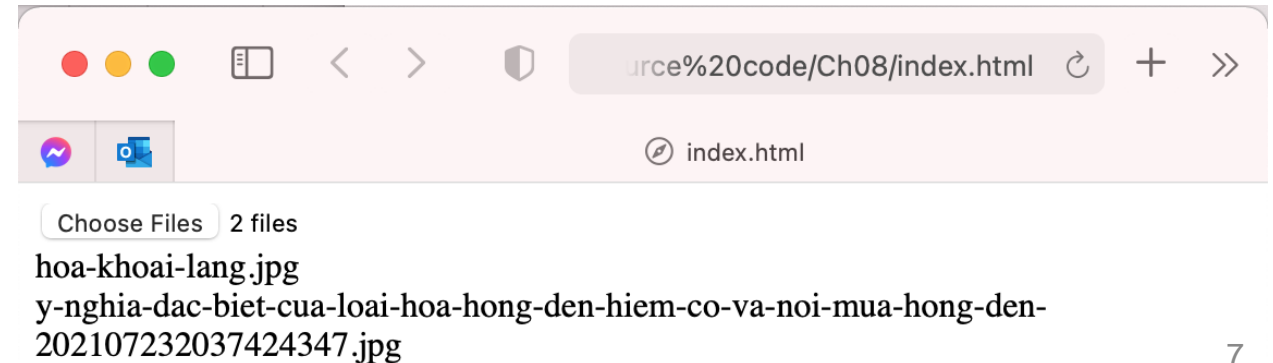
# Demo uploading files – 1 tập tin

```
<!DOCTYPE html>
<html>
  <body>
    <input type="file" onchange="uploadFile(this.files)" />
    <div id="message"></div>
    <script>
      let message = document.getElementById("message");
      function uploadFile(files) {
        console.log(files[0]);
        message.innerText = files[0].name;
      }
    </script>
  </body>
</html>
```



# Demo uploading files – nhiều tập tin

```
<!DOCTYPE html>
<html>
  <body>
    <input type="file" multiple onchange="uploadFile(this.files)" />
    <div id="message"></div>
    <script>
      let message = document.getElementById("message");
      function uploadFile(files) {
        for (let i = 0; i < files.length; i++) {
          message.innerHTML += files[i].name + "<br>";
        }
      }
    </script>
  </body>
</html>
```



# Geolocation

---

- HTML Geolocation API

- được sử dụng để xác định vị trí của người dùng
- Hiển thị các địa điểm được yêu thích gần người dùng
- Chỉ đường cho người dùng (GPS)
- Cập nhật, lưu trữ thông tin về vị trí người dùng
- .....

- Cú pháp

`navigator.geolocation.getCurrentPosition(success, error);`

- Trong đó phương thức `getCurrentPosition()` trả về vị trí của người dùng
- `getCurrentPosition()` trả về kết quả là một đối tượng



# Geolocation

---

- Chi tiết các thuộc tính trả về

Thuộc tính	Giá trị trả về
<code>coords.latitude</code>	Vĩ độ ở dạng số thập phân (giá trị này luôn được trả về)
<code>coords.longitude</code>	Kinh độ ở dạng số thập phân (giá trị này luôn được trả về)
<code>coords.accuracy</code>	Độ chính xác của vị trí
...	< tham khảo thêm các thuộc tính khác> <a href="https://www.w3schools.com/html/html5_geolocation.asp">https://www.w3schools.com/html/html5_geolocation.asp</a>

# Demo geolocation

<p>Click the button to get your coordinates.</p>

<button onclick="getLocation()">Try It</button>

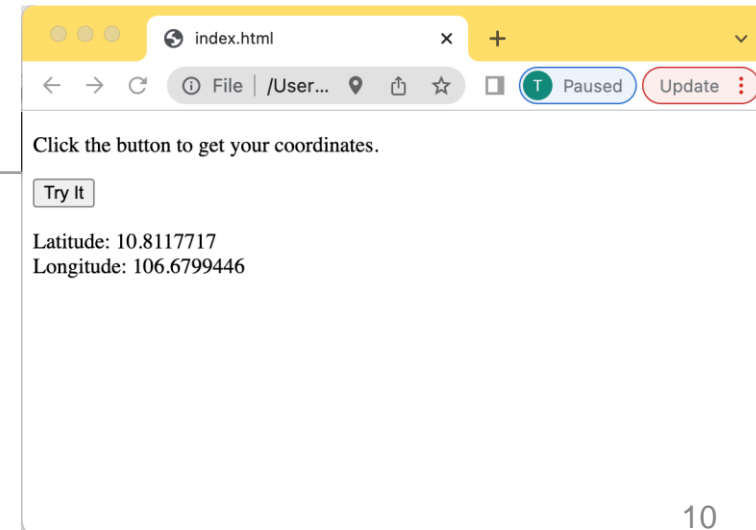
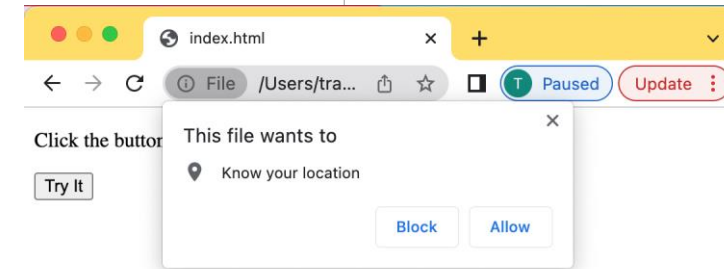
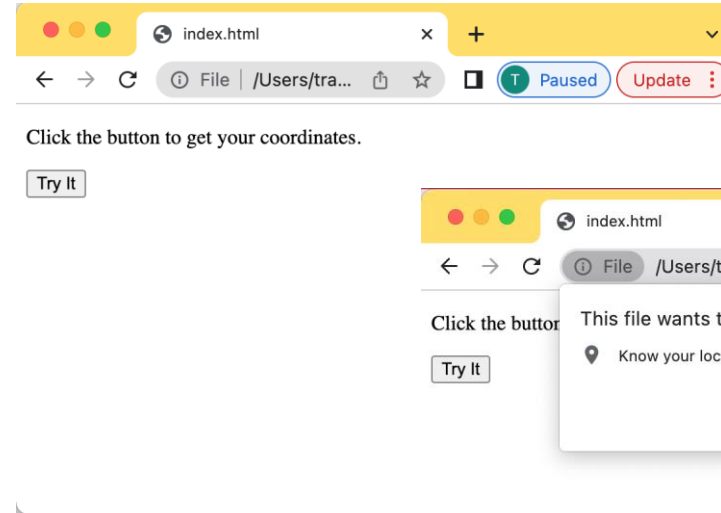
<p id="demo"></p>

<script>

var x = document.getElementById("demo");

```
function getLocation() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(showPosition);
  } else {
    x.innerHTML = "Geolocation is not supported by this browser.";
  }
}
```

```
function showPosition(position) {
  x.innerHTML = "Latitude: " + position.coords.latitude +
    "<br>Longitude: " + position.coords.longitude;
}
</script>
```



# Demo geolocation

---

- Giải thích ví dụ
  - Kiểm tra xem **Geolocation** có được trình duyệt hỗ trợ không
  - Nếu trình duyệt hỗ trợ, chạy phương thức **getCurrentPosition()**.  
Nếu không, hiển thị thông báo cho người dùng.
  - Nếu hàm **getCurrentPosition()** chạy thành công, thì nó trả về cho người dùng tọa độ vị trí thông qua hàm truyền tham số (showPosition)
  - Hàm **showPosition()** in ra là kinh độ và vĩ độ của người dùng.

# Media on page

---

- **Ôn tập:** hiển thị audio với HTML

```
<!DOCTYPE html>
<html>
  <body>
    <audio controls>
      <source src="sound.ogg" type="audio/ogg">
      <source src="sound.mp3" type="audio/mpeg">
    </audio>
  </body>
</html>
```

# Media on page

---

- **Ôn tập:** hiển thị video (video, youtube) với HTML

```
<video width="1024" height="576" controls>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogv" type="video/ogg">  
</video>
```

```
<iframe  
  width="1024"  
  height="576"  
  src="https://www.youtube.com/watch?v=P9c-gQiie0M">  
</iframe>
```

# HTML Audio/Video DOM

- HTML Audio/Video DOM cung cấp các phương thức, thuộc tính và sự kiện cho `<audio>` và `<video>` element
- Methods

Phương thức	Mô tả
Load()	Reload lại audio/video element
Play()	Bắt đầu play audio/video
Pause()	Dừng play audio/video hiện tại

# HTML Audio/Video DOM

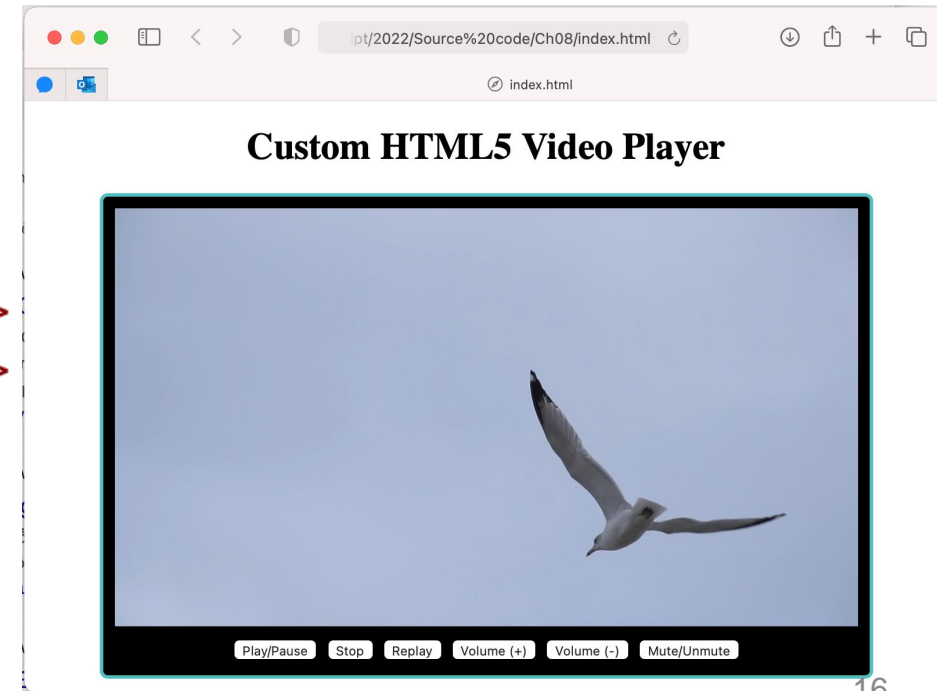
---

- Thuộc tính

Thuộc tính	Mô tả
autoplay	audio/video có tự động play sau khi load hay không (set/return)
controls	Audio/video có hiển thị các controls (set/return)
duration	Trả về kích thước (length) của audio/video hiện hành
loop	Sau khi kết thúc bài có audio/video có lặp lại hay không?
muted	Audio/video có tắt tiếng hay không?
paused	Audio/video có đang tạm dừng hay không
volume	Âm thanh của audio/video

# Demo HTML Audio/Video DOM

```
<div id="container">
  <h1>Custom HTML5 Video Player</h1>
  <div id="player-container">
    <video id="myplayer" width="640" height="360">
      <source src="SeagullsMediumAltMaster.mp4" type="video/mp4">
      Your browser does not support HTML5 Video.
    </video>
    <div id="controls">
      <button onclick="playPause()">Play/Pause</button>
      <button onclick="stopVideo()">Stop</button>
      <button onclick="replayVideo()">Replay</button>
      <button onclick="increaseVolume()">Volume (+)</button>
      <button onclick="decreaseVolume()">Volume (-)</button>
      <button onclick="muteVolume()">Mute/Unmute</button>
    </div>
  </div>
</div>
```





# Demo HTML Audio/Video DOM

```
var player = document.getElementById("myplayer");

function playPause() {
    if (player.paused) { player.play(); }
    else { player.pause(); }
}

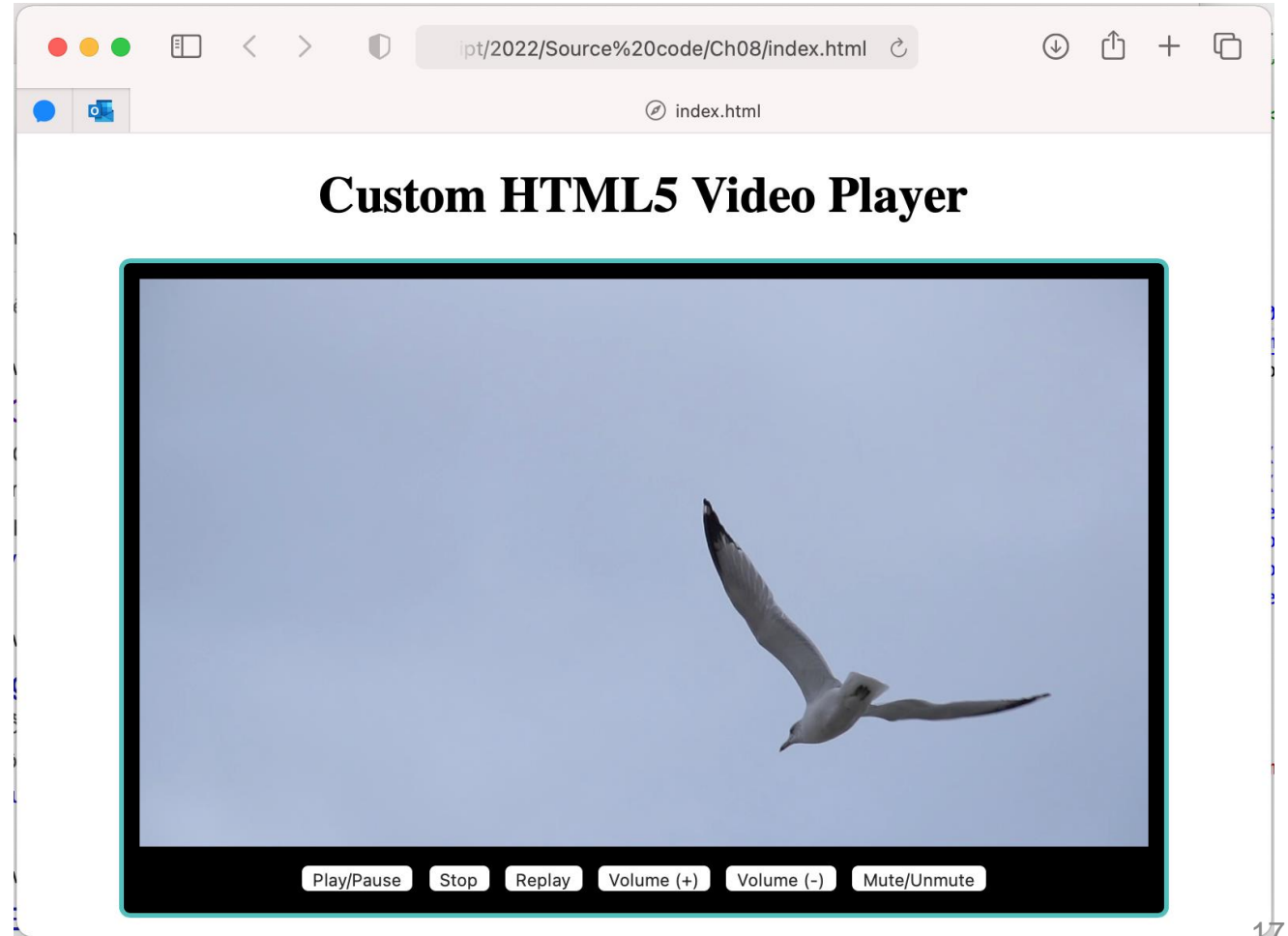
function stopVideo() {
    player.pause();
    if (player.currentTime) { player.currentTime = 0; }
}

function replayVideo() {
    player.currentTime = 0;
    player.play();
}

function increaseVolume() {
    if (player.volume < 1)
        player.volume = parseFloat(player.volume + 0.1).toFixed(1);
}

function decreaseVolume() {
    if (player.volume > 0)
        player.volume = parseFloat(player.volume - 0.1).toFixed(1);
}

function muteVolume() {
    if (player.muted) { player.muted = false; }
    else { player.muted = true; }
}
```





# PHẦN II

---

## LIBRARIES & FRAMEWORKS

# Libraries & framework

---

- Libraries:
  - về cơ bản là các module javascript được lập trình sẵn, có thể sử dụng để tăng tốc quá trình phát triển của mình.
  - Thường làm một việc cụ thể
- Framework: một gói thư viện cung cấp tất cả giải pháp trong một
- Libraries & framework luôn được cập nhật để sửa lỗi cho các vấn đề bảo mật.

# Libraries & framework

---

- Giới thiệu một số thư viện



UNDERSCORE.JS

---

# Libraries & framework

---

- Giới thiệu một số thư viện: **Jquery**
  - Là thư viện nổi tiếng của javascript
  - Thao tác với HTML/DOM
  - Thao tác CSS
  - Các phương thức của HTML event
  - Hiệu ứng và các animation
  - AJAX
  - Tiện ích



# Libraries & framework

---

- Giới thiệu một số thư viện: **D3**
  - Viết tắt của Data Driven Documents
  - Trực quan hoá dữ liệu tương tác trong trình duyệt web bằng cách sử dụng SVG, HTML và CSS (thư viện đồ hoạ



# Libraries & framework



- Giới thiệu framewok:
  - Framework phát triển dựa trên javascript để tạo các ứng dụng web
  - Thường dùng để phát triển frontend thông qua các API để gọi data, sử dụng mô hình MVC rất mạnh mẽ.
  - Mã nguồn AngularJS tự động fix các trình duyệt khác nhau.
  - Mã nguồn mở, hoàn toàn miễn phí
- Nhược điểm
  - Người dùng có thể vô hiệu hoá javascript vì angularjs được phát triển từ javascript.

# Libraries & framework

---

- Node.js là một thư viện của javascript, hơn thế nữa node.js là cả một platform với 4 sức mạnh
  - Ngôn ngữ phổ biến
  - Tốc độ nhanh (nhờ V8)
  - Paradigm mới (non blocking I/O)
  - Nhiều thư viện, dễ cài (nhờ npm)

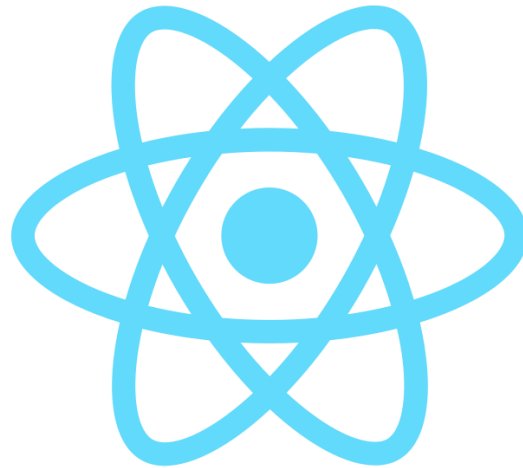




# Libraries & framework

---

- Một số framework khác



# TỔNG KẾT BÀI HỌC

---

- Upload Files
- Geolocation
- Media on page (audio, video)
- Giới thiệu Libraries & Frameworks
  - JQuery / D3 / underscore.js
  - Angularjs
  - Vue.js
  - Reactjs
  - Angular

*Thank  
you!*