

# Internet và Công nghệ Web

## JQuery





# Nội dung

1. Giới thiệu JQuery
2. Cú pháp
3. Thao tác với phần tử DOM
4. Xử lý sự kiện - Event Handling
5. Hiệu ứng - Effect
6. Animation
7. Ajax
8. Bài tập



# Giới thiệu

- jQuery là một thư viện JavaScript, dễ sử dụng.
- Đơn giản: xử lý HTML document, sự kiện, animation, tương tác Ajax.
- **Animation** là tạo chuyển động hoặc thay đổi đặc điểm phần tử HTML mà không **tải lại trang**.
- **Animation** bao gồm thay đổi kích thước, vị trí, màu sắc, độ trong suốt, hoặc thuộc tính khác của phần tử để tạo hiệu ứng động.
- **AJAX** (Asynchronous JavaScript and XML) tạo trang web tương tác mà không **tải lại trang**.
- **AJAX** giao tiếp với máy chủ và trao đổi dữ liệu bất đồng bộ => gửi và nhận thông tin từ máy chủ mà không cần chuyển trang hoặc làm mới toàn bộ nội dung trang.



# Cú pháp

## Tích hợp

```
<!DOCTYPE html>
<html>
<head>
  <title>My Website</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
  <!-- Nội dung trang web -->

  <!-- JS sử dụng jQuery -->
  <script>
    $(document).ready(function () {
      // Code jQuery của bạn ở đây
    });
  </script>
</body>
</html>
```



# Cú pháp

## *Phương pháp chung*

- Dùng selector: tìm và chọn các phần tử DOM dựa trên các điều kiện.
- **\$()** để bắt đầu một selector.
- Trang tra cứu: <https://api.jquery.com/>



# Cú pháp

## *Chọn phần tử DOM bằng selector*

Cách chọn	Cú pháp	Kết quả
Phần tử theo tên <b>thẻ</b>	<code>\$("p")</code>	Chọn tất cả các thẻ <code>&lt;p&gt;</code>
Phần tử theo <b>id</b>	<code>\$("#myId")</code>	Chọn thẻ có id là "myid"
Phần tử theo <b>class</b>	<code>\$(".myClass")</code>	Chọn tất cả các phần tử có class là "myclass"
Kết hợp tên <b>thẻ</b> và tên <b>class</b>	<code>\$("p.myClass")</code>	Chọn các thẻ <code>&lt;p&gt;</code> có class là "myclass"



# Cú pháp

## *Chọn phần tử DOM bằng selector*

Cách chọn	Cú pháp	Kết quả
Thuộc tính cụ thể	<code>\$("[data-attribute]")</code>	Chọn các phần tử có thuộc tính bất kỳ
Thuộc tính cụ thể	<code>\$("[data-attribute='value']")</code>	Chọn phần tử có thuộc tính là "value"
Phần tử con của một phần tử	<code>\$("#parentEle &gt; *")</code>	Chọn tất cả các phần tử con của phần tử có id "parentEle"
Phần tử con của một phần tử	<code>\$("#parentEle &gt; .childClass")</code>	Chọn các phần tử con có class là "childClass" của phần tử có id là "parentEle"



# Cú pháp

## Chọn phần tử DOM bằng selector

Cách chọn	Cú pháp	Kết quả
Kết hợp nhiều selector	<code>\$("p, .myClass")</code>	Chọn tất cả các thẻ <code>&lt;p&gt;</code> và các phần tử có class là "myClass"
Phần tử thỏa cả hai điều kiện (AND)	<code>\$("p.myClass:first")</code>	Chọn thẻ <code>&lt;p&gt;</code> đầu tiên có class là "myClass"
Phần tử thỏa ít nhất một trong các điều kiện (OR)	<code>\$("p.myClass, .otherClass")</code>	Chọn các thẻ <code>&lt;p&gt;</code> có class "myClass" hoặc các phần tử có class "otherClass"





# Cú pháp

## *Chọn phần tử DOM bằng selector*

Cách chọn	Cú pháp	Kết quả
Sử dụng .find() để tìm các phần tử con trong phần tử đã được chọn	<code>\$("#parentEle").find("p")</code>	Tìm các thẻ <p> trong phần tử có id là "parentEle"
Sử dụng .filter() để lọc các phần tử dựa trên điều kiện	<code>\$("p").filter(".myClass")</code>	Lọc ra các thẻ <p> có class là "myClass"



# Sự kiện

## *Giới thiệu*

- Sự kiện là phần quan trọng thể hiện khả năng tương tác trang web.
- jQuery cung cấp các sự kiện để phản ứng và xử lý các tương tác người dùng và web.
- jQuery hỗ trợ cú pháp **\$(document)** xử lý sự kiện trực tiếp vào HTML.

```
$(document).tên_sự_kiện(function () {  
  
    // Code thực hiện  
    // xử lý  
  
});
```



# Sự kiện ready

- **ready**: xảy ra khi DOM đã được tạo xong, tất cả các phần tử HTML đã được tải vào bộ nhớ trình duyệt => truy cập và tương tác

```
$(document).ready(function () {  
    // Code thực hiện  
    // Khi DOM sẵn sàng, tải xong  
});
```



# Sự kiện click

- **click**: xảy ra khi một phần tử được click chuột.

```
$(document).on("click", ".my-element", function () {  
    // Xử lý khi phần tử class="my-element" được click  
});
```

```
<!DOCTYPE html>
<html>
<head>
  <title>jQuery Click Event Example</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
</head>
<body>

  <button class="my-element">Click Thực hiện</button>

  <script>
    $(document).on("click", ".my-element", function () {
      alert("Button đã được click");
    });
  </script>

</body>
</html>
```



# Sự kiện change

- **change**: xảy ra khi giá trị của phần tử **input** hoặc **select** thay đổi

```
$(document).on("change", "#my-input", function () {  
    // Xử lý khi thay đổi giá trị của phần tử có id="my-input"  
});
```

```
<!DOCTYPE html>
<html>
<head>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">
  </script>
</head>
<body>
  <label for="id-input">Input:</label>
  <input type="text" id="id-input" placeholder="Nhập thông tin...">

  <div id="output"></div>

  <script>
    $(document).on("change", "#id-input", function () {
      var value = $(this).val(); //Lấy giá trị hiện tại của phần tử HTML
      $('#output').text('Output: ' + value);
    });
  </script>
</body>
</html>
```



# Sự kiện submit

- **submit**: xảy ra khi form gửi

```
$(document).on("submit", "#my-form", function () {  
    // Xử lý khi biểu mẫu có id="my-form" được gửi  
});
```





# Sự kiện keydown/keyup

- **keydown/keyup**: xảy ra khi phím được nhấn xuống hoặc thả ra

```
$(document).on("keydown", function (event) {  
    // Xử lý khi phím được nhấn  
});
```



# Sự kiện scroll

- **scroll**: xảy ra khi cuộn lên hoặc cuộn xuống

```
$(document).on("scroll", function () {  
    // Xử lý khi được cuộn lên hoặc xuống  
});
```



# Sự kiện mouseenter - mouseleave

- **mouseenter - mouseleave**: xảy ra khi con trỏ chuột tiến vào hoặc ra khỏi phần tử

```
$(document).on("mouseenter", ".my-element", function () {  
    // Xử lý khi chuột vào phần tử class="my-element"  
});
```



## Sự kiện focus - blur

- **focus - blur**: xảy ra khi phần tử được focus hoặc blur

```
$(document).on("focus", "#my-input", function () {  
    // Xử lý khi phần tử id="my-input" được focus  
});
```

```
$(document).ready(function () {  
    $("#click-button").click(function () {  
        alert("Đã click vào Button");  
    });  
  
    $("#change-input").change(function () {  
        alert("Đã thay đổi giá trị");  
    });  
  
    $("#submit-form").submit(function (e) {  
        e.preventDefault();  
        alert("Đã submit");  
    });  
  
    $("#keypress-box").keydown(function () {  
        console.log("Nhấn phím");  
    }).keyup(function () {  
        console.log("Thả phím");  
    });  
});
```

```
$("#mo-leave-box").mouseenter(function () {  
    console.log("Mouse đã vào box.");  
}).mouseleave(function () {  
    console.log("Mouse ra khỏi box.");  
});  
  
$("#focus-blur-input").focus(function () {  
    console.log("Đang focus");  
}).blur(function () {  
    console.log("Không còn focus");  
});  
  
$("#scroll-box").scroll(function () {  
    console.log("Đã scroll trong box");  
});  
});
```



# Hiệu ứng - Effect

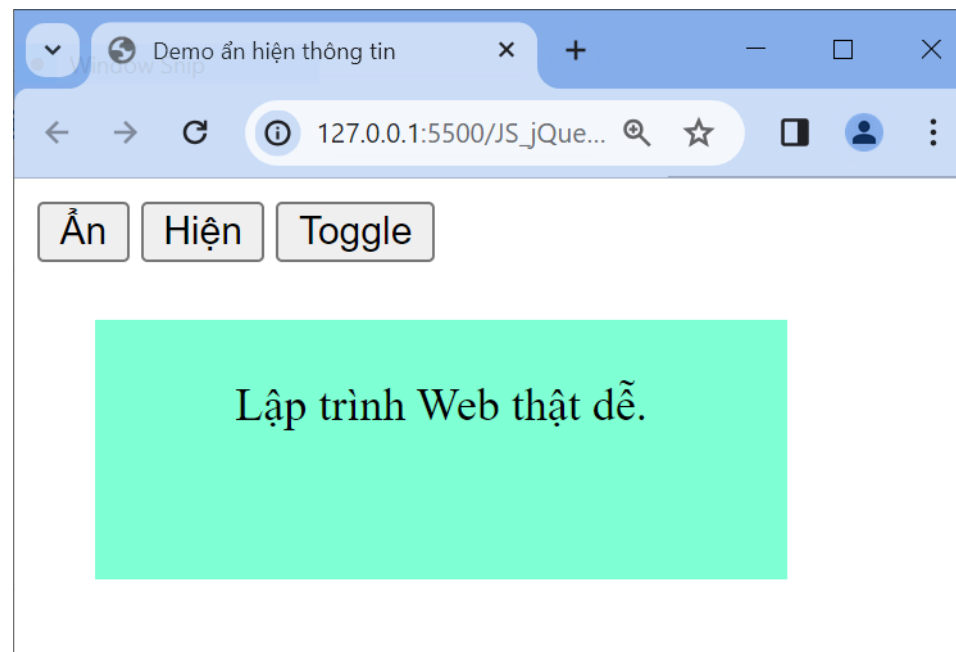
## *Giới thiệu*

- jQuery cung cấp hiệu ứng:
  - Ẩn/hiện - hide/show
  - Ẩn hiện theo hiệu ứng mờ dần - fade in/out
  - Trượt lên, trượt xuống - slide up/down
  - ...

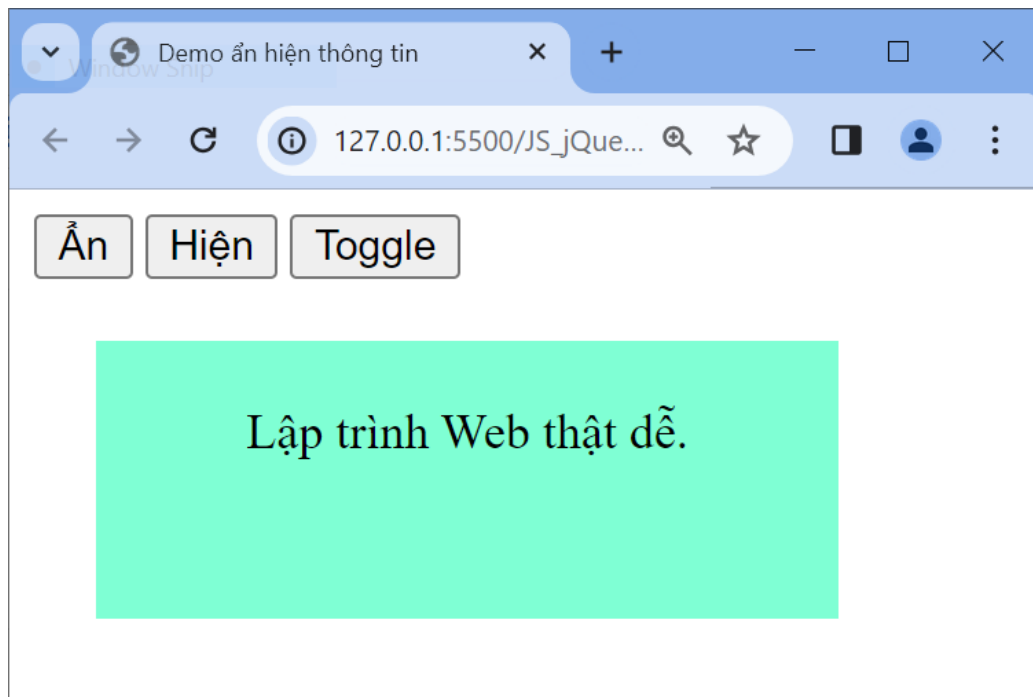


# Hiệu ứng hide-show

Ẩn một phần tử	<code>\$("#idBox").hide()</code>
Hiển thị một phần tử	<code>\$("#idBox").show()</code>
Chuyển đổi giữa trạng thái <b>ẩn</b> và <b>hiển</b> thị của một phần tử	<code>\$("#idBox").toggle()</code>



```
<style>
    .box {
        width: 200px;
        height: 50px;
        background-color: aquamarine;
        text-align: center;
        padding: 20px;
        margin: 20px;
    }
</style>
```



```
<button id="hideButton">Ẩn</button>
<button id="showButton">Hiện</button>
<button id="toggleButton">Toggle</button>
```

```
<div id="idBox" class="box">
    Lập trình Web thật dễ.
</div>
```

```
<script>
    $(document).ready(function () {
        $("#hideButton").click(function () {
            $("#idBox").hide();
        });

        $("#showButton").click(function () {
            $("#idBox").show();
        });

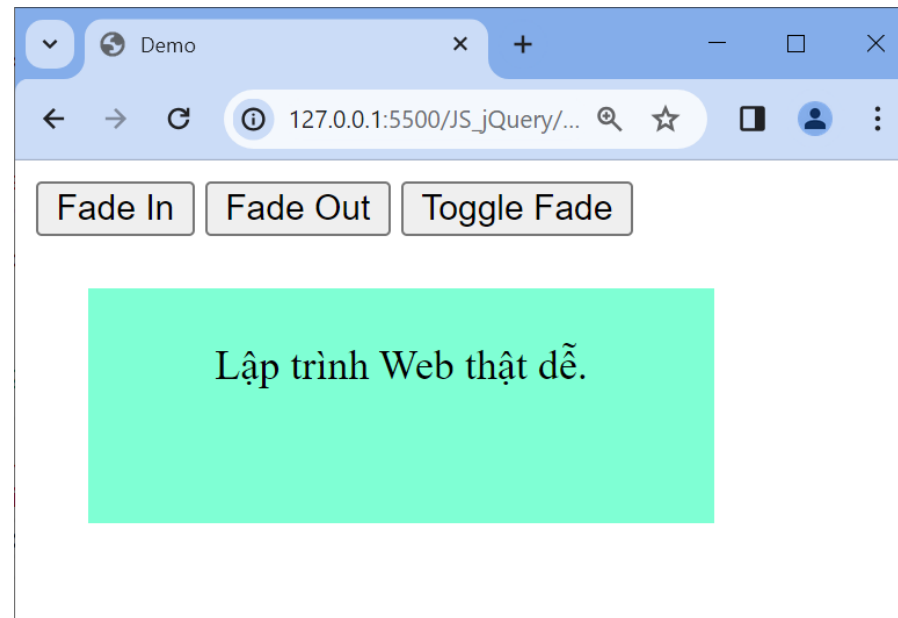
        $("#toggleButton").click(function () {
            $("#idBox").toggle();
        });
    });
</script>
```



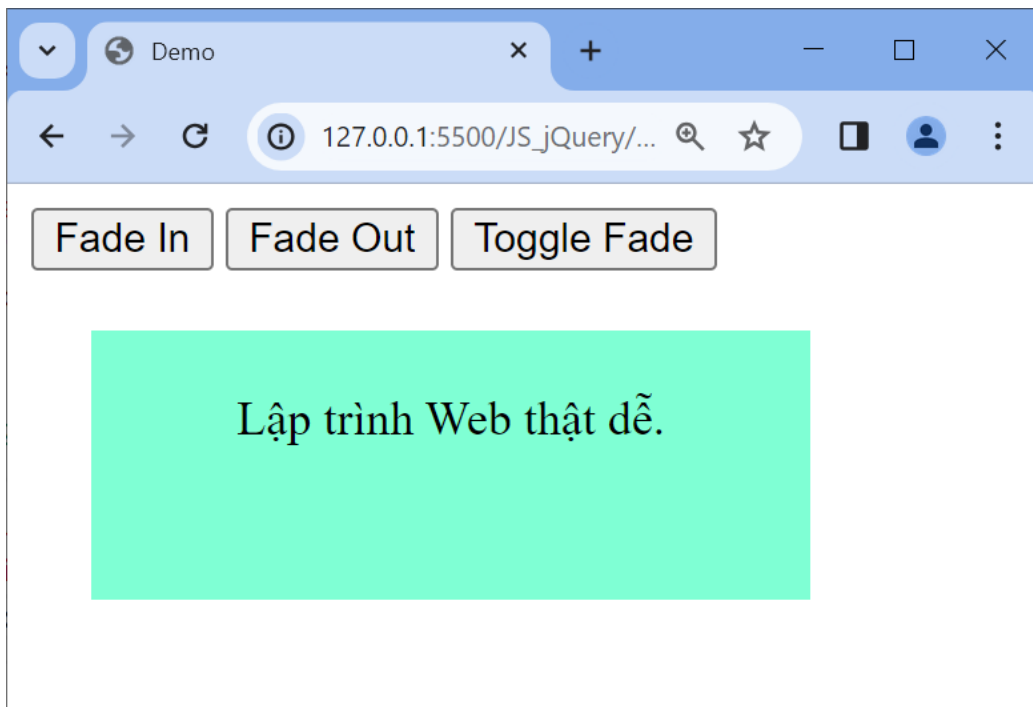


# Hiệu ứng fade

Phần tử hiển thị với hiệu ứng mờ	<code>\$("#idBox").fadeIn()</code>
Phần tử ẩn với hiệu ứng mờ	<code>\$("#idBox").fadeOut()</code>
Chuyển đổi giữa trạng thái hiển thị và ẩn hiệu ứng mờ	<code>\$("#idBox").fadeToggle()</code>



```
<style>
    .box {
        width: 200px;
        height: 50px;
        background-color: aquamarine;
        text-align: center;
        padding: 20px;
        margin: 20px;
    }
</style>
```



```
<button id="fadeInButton">Fade In</button>
<button id="fadeOutButton">Fade Out</button>
<button id="toggleFadeButton">Toggle Fade</button>

<div id="idBox" class="box">
    Lập trình Web thật dễ.
</div>
```

```
<script>
    $(document).ready(function () {
        $("#fadeInButton").click(function () {
            $("#idBox").fadeIn();
        });

        $("#fadeOutButton").click(function () {
            $("#idBox").fadeOut();
        });

        $("#toggleFadeButton").click(function () {
            $("#idBox").fadeToggle();
        });
    });
</script>
```

```
<script>
    $(document).ready(function () {
        $("#fadeInButton").click(function () {
            $("#idBox").fadeIn(3000);
        });

        $("#fadeOutButton").click(function () {
            $("#idBox").fadeOut(3000);
        });

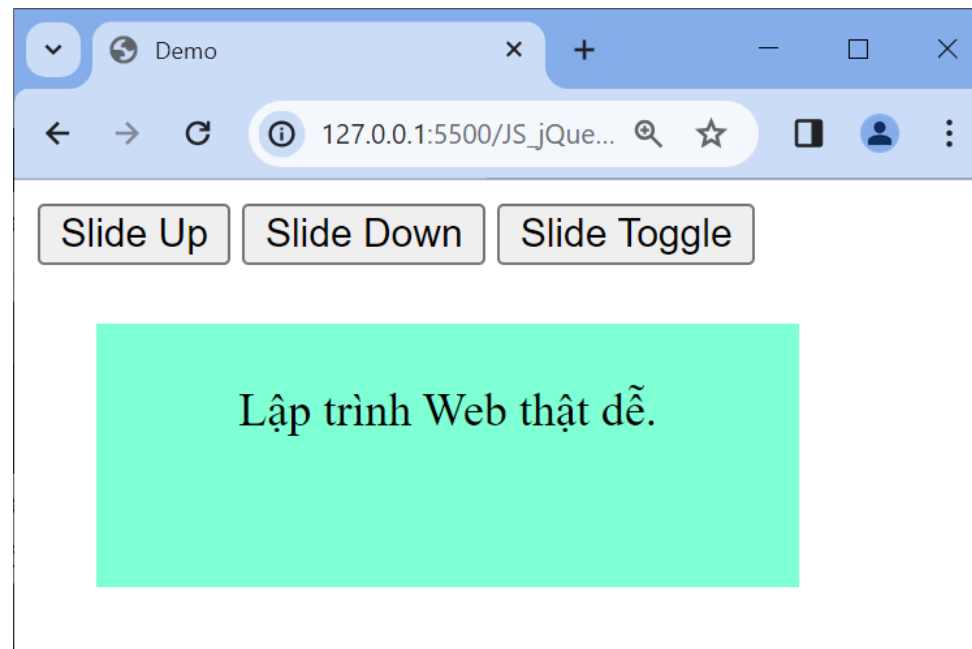
        $("#toggleFadeButton").click(function () {
            $("#idBox").fadeToggle(3000, function(){

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



# Hiệu ứng slide

Phần tử trượt lên và biến mất	<code>\$("#idBox").slideUp()</code>
Phần tử trượt xuống và hiển thị	<code>\$("#idBox").slideDown()</code>
Chuyển đổi giữa trạng thái trượt lên và trượt xuống của phần tử	<code>\$("#idBox").slideToggle()</code>



```
<style>
    .box {
        width: 200px;
        height: 50px;
        background-color: aquamarine;
        text-align: center;
        padding: 20px;
        margin: 20px;
    }
</style>
```

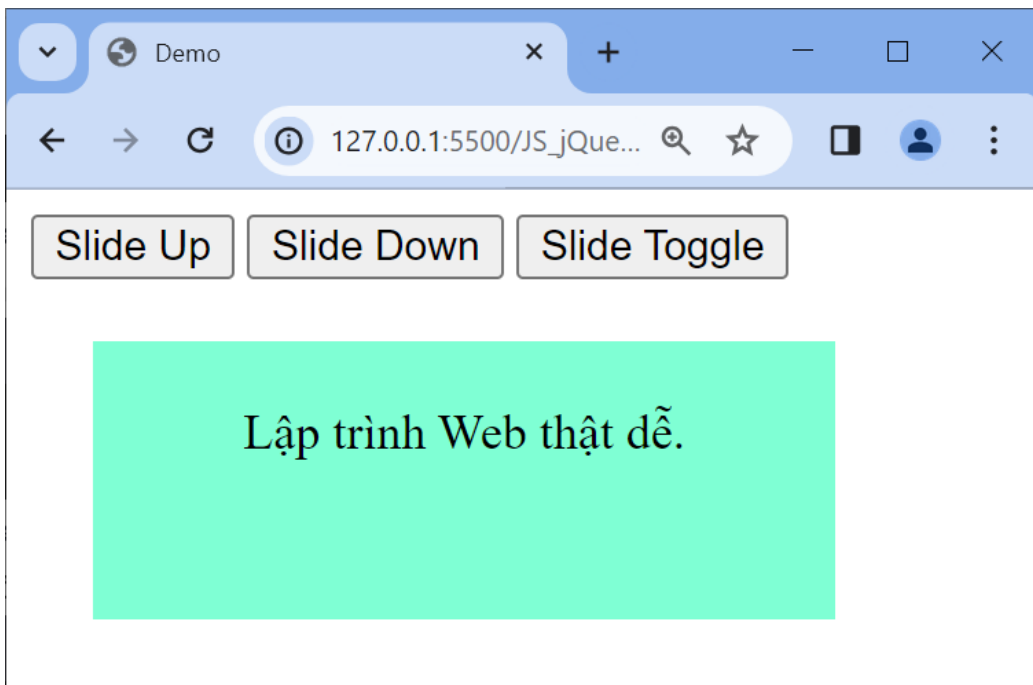
```
<button id="slideUpButton">Slide Up</button>
<button id="slideDownButton">Slide Down</button>
<button id="slideToggleButton">Slide Toggle</button>

<div id="idBox" class="box">
    Lập trình Web thật dễ.
</div>
```

```
<script>
    $(document).ready(function () {
        $("#slideUpButton").click(function () {
            $("#idBox").slideUp();
        });

        $("#slideDownButton").click(function () {
            $("#idBox").slideDown();
        });

        $("#slideToggleButton").click(function () {
            $("#idBox").slideToggle();
        });
    });
</script>
```





# Animation

## Cú pháp

```
$(selector).animate(properties, duration, easing, complete);
```

Trong đó:

- **selector**: Phần tử áp dụng animation.
- **properties**: Thuộc tính CSS và giá trị muốn thay đổi => kiểu đối tượng JSON theo dạng key-value.
- **duration**: Thời gian thực hiện animation (mili giây).
- **easing** (tùy chọn): Loại chuyển động: "**swing**", "linear", "easeInOut"...
- **complete** (tùy chọn): callback hàm sau khi animation hoàn thành.

```

<style>
    .box {
        width: 200px;
        height: 50px;
        background-color: aquamarine;
        text-align: center;
        padding: 20px;
        margin: 20px;
        position: relative;
    }
</style>

```

```

<button id="idButton">
    Start Animation
</button>
<div id="idBox" class="box">
    Học lập trình Web thật dễ.
</div>

```

```

<script>
    $(document).ready(function () {
        var animatedElement = $("#idBox");
        function startAnimation() {
            animatedElement.animate(
                {
                    left: '+=200px', // Sang phải
                    top: '+=50px',   // Xuống dưới
                    opacity: 0.5, // Giảm độ trong
                    width: '200px',
                    height: '50px'
                },
                3000,
                "linear"
            );
        }

        $("#idButton").click(startAnimation);
    });
</script>

```



# Ajax

## *Giới thiệu*

- **AJAX** (Asynchronous JavaScript and XML) tạo trang web tương tác mà không **tải lại trang**.
- **AJAX** giao tiếp với máy chủ và trao đổi dữ liệu bất đồng bộ => gửi và nhận thông tin từ máy chủ mà không cần chuyển trang hoặc làm mới toàn bộ nội dung trang.





## Cú pháp

```
$.ajax({  
    url: "URL_API",           // Đường dẫn tới API hoặc tập tin  
    method: "METHOD",        // Phương thức HTTP (GET, POST, PUT, DELETE...)  
    data: DATA,              // Dữ liệu gửi đi (nếu có)  
    dataType: "DATA_TYPE",    // Kiểu dữ liệu mong đợi (json, xml, html...)  
    success: function (data) {  
        // Xử lý dữ liệu khi yêu cầu thành công  
    },  
    error: function () {  
        // Xử lý lỗi khi yêu cầu thất bại  
    }  
});
```



# Bài tập

- Thiết kế menu
- Hướng dẫn trên lớp



Thank you



# Thảo luận

