



# MÔN THIẾT KẾ WEB

BỘ MÔN KỸ THUẬT VÀ MẠNG MÁY TÍNH



## Bài 5: Cơ bản về Javascript

Giới thiệu về Javascript

01

Cú pháp trong Javascript

02

Biến, kiểu dữ liệu, hàm

03

Các câu lệnh trong Javascript

04



Phần thứ nhất

# GIỚI THIỆU VỀ JAVASCRIPT



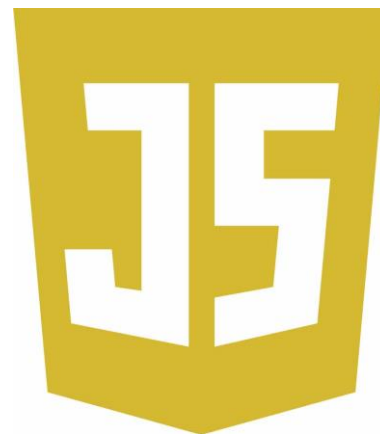
# 1



# Giới thiệu về Javascript

Javascript là gì?

- Ngôn ngữ thông dịch, mã nguồn của nó được nhúng hoặc tích hợp trực tiếp vào các tập tin HTML. Khi trang web được tải xong, trình duyệt sẽ thông dịch và thực hiện các mã lệnh này
- Ngôn ngữ được cung cấp hoàn toàn miễn phí



JavaScript™



# Giới thiệu về Javascript

Javascript có thể làm được gì?

- Giúp cho trang HTML trở nên sinh động hơn
- Thực hiện một công việc với sự kiện nào đó từ phía người dùng
- Có thể thay đổi giá trị thuộc tính của thẻ HTML
- Có thể thay đổi CSS của trang HTML
- Có thể ẩn các thành phần của trang HTML
- Kiểm tra tính chính xác của dữ liệu
- Phát hiện các loại trình duyệt khác nhau
- Tạo các tập tin cookie lưu trữ và truy xuất thông tin trên máy tính của người truy cập website
- ...





# Giới thiệu về Javascript

Sử dụng Javascript như thế nào?

- Các câu lệnh JavaScript phải được chèn trong thẻ `<script>` và `</script>` và có thể đặt trong phần head hoặc body của trang web hoặc để trong file riêng.



```
<!DOCTYPE html>
<html>

<head>
<script>
function myFunction() {
    document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</head>

<body>


<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

</body>
</html>
```

# Giới thiệu về Javascript

Javascript thực thi lệnh khi nào?

- Trường hợp **thứ nhất**: Ngay khi trang web được tải về trình duyệt của người sử dụng.
- Trường hợp **thứ hai**: Thực hiện khi nhận được một tác động nào đó như nhất nút, di chuyển chuột, ...



```
<!DOCTYPE html>
<html>
<body>

<h1>My First Web Page</h1>
<p>My first paragraph.</p>

<script>
document.write(5 + 6);
</script>

</body>
</html>
```

# Giới thiệu về Javascript

Javascript thực thi lệnh khi nào?

➤ Danh sách các sự kiện thông dụng của JavaScript

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page



# Giới thiệu về Javascript

Vị trí của Javascript trong trang HTML?

- Đặt trong cặp thẻ `<head>` của trang HTML
- Đặt trong cặp thẻ `<body>` của trang HTML
- Đặt trong tập tin `.js` sau đó nhúng tập tin này vào trong HTML

```
<!DOCTYPE html>
<html>
<body>

<script src="myScript.js"></script>

</body>
</html>
```

External file: myScript.js

```
function myFunction() {
    document.getElementById("demo").innerHTML = "Paragraph changed.";
}
```



Phần thứ hai

# CÚ PHÁP TRONG JAVASCRIPT



# 2



# Cú pháp trong Javascript

## Quy tắc sử dụng cú pháp lệnh

- Javascript phân biệt chữ hoa chữ thường
- Kết thúc câu lệnh của Javascript bằng dấu chấm phẩy “;”
- Một câu lệnh JavaScript có thể bao gồm:
  - ✓ **Giá trị**: có thể là hằng hoặc biến
  - ✓ **Toán tử (phép toán)**: Các phép toán số học (+, -, \*, /, %, ++, --); các phép toán logic (&&, ||, !); các phép toán so sánh (>, <, >=, <=, ==, !=); phép gán =
  - ✓ **Biểu thức**: Là sự kết hợp của hằng số, biến số, phép toán để tính toán ra một kết quả
  - ✓ **Từ khóa (keyword)**: là một từ hoặc một cụm từ được định nghĩa bởi ngôn ngữ Javascript
  - ✓ **Chú thích (comment)**: giải thích quá trình viết mã lệnh có thể dùng chú thích trên 1 dòng (dùng “//”) hoặc trên nhiều dòng (dùng cặp “/\*...\*/”)

### Cú pháp

```
variablename = (condition) ? value1:value2
```

```
var voteable = (age < 18) ? "Too young":"Old enough";
```

# Cú pháp trong Javascript

## Cú pháp lệnh cơ bản

- Hiển thị một dòng văn bản bằng câu lệnh `document.write`
- Ghi ra nội dung thẻ HTML sử dụng `innerHTML`
- Viết ra cảnh báo bằng cách sử dụng `window.alert()`

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Web Page</h1>
<p>My first paragraph.</p>

<script>
window.alert(5 + 6);
</script>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Web Page</h1>
<p>My First Paragraph</p>

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

<script>
document.getElementById("demo").innerHTML = 5 + 6;
</script>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Web Page</h1>
<p>My first paragraph.</p>

<script>
document.write(5 + 6);
</script>

</body>
</html>
```



Phần thứ ba

# BIẾN, KIỂU DỮ LIỆU, HÀM TRONG JAVASCRIPT

3



# Biến, kiểu dữ liệu, hàm trong Javascript

## Biến trong Javascript

- Được dùng để lưu trữ một giá trị nào đó có thể là một chuỗi, một con số, một mảng, một phép toán hoặc một đối tượng
- Cách khai báo trong Javascript theo cú pháp
  - ✓ `Var <tên_biến>;`
  - ✓ `<tên_biến>=giá_trị;`



```
var x;
```

```
x = 6;
```



# Biến, kiểu dữ liệu, hàm trong Javascript

## Kiểu dữ liệu trong Javascript

- Trong lập trình, kiểu dữ liệu là một khái niệm rất quan trọng. Các biến trong javascript có thể chứa nhiều kiểu dữ liệu khác nhau có thể là số, chuỗi, đối tượng ...
- Tuy nhiên, Javascript vẫn có các kiểu dữ liệu cụ thể gồm: xâu, số, logic, đối tượng, mảng và không định nghĩa



```
var a = typeof ""; //Kiểu xâu ký tự  
var b = typeof "Thiết kế Web"; //Kiểu xâu ký tự  
var c = typeof 2020; //Kiểu số  
var d = typeof true; //Kiểu logic  
var f = typeof {name:"TKW", gender:"Nam", year:2020}; //Kiểu đối tượng  
var g = typeof undefined; //Không định nghĩa  
var h = typeof ["Kế toán","CNTT","ĐH Công nghiệp Hà nội"]; //Kiểu mảng
```



# Biến, kiểu dữ liệu, hàm trong Javascript

## Kiểu dữ liệu trong Javascript

- Đặc biệt, trong Javascript có khả năng xác định kiểu dữ liệu động theo giá trị mà biến số được nhận về thông qua một phép gán
- JavaScript đánh giá các biểu thức từ trái sang phải. Các trình tự khác nhau có thể tạo ra các kết quả khác nhau

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript</h2>

<p>JavaScript đánh giá các biểu thức từ trái sang phải. Kết quả sẽ được thực hiện phép
tính tổng giữa hai số sau đó tiếp tục nối xâu:</p>

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

<script>
var x = 16 + 4 + "Khoa CNTT";
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript</h2>

<p>Khi thêm 1 số với một xâu ký tự, JavaScript sẽ trả về kết quả là một xâu ký tự.</p>

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

<script>
var x = 16 + "Văn bản";
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

# Biến, kiểu dữ liệu, hàm trong Javascript

## Hàm trong Javascript

- Hàm trong JavaScript là một khối lệnh được thiết kế để thực hiện một nhiệm vụ xác định
- Hàm được thực thi khi nó được gọi, hàm có thể được gọi khi:
  - ✓ Một sự kiện xảy ra (khi người sử dụng kích chuột vào một nút lệnh).
  - ✓ Được gọi trong đoạn mã JavaScript
  - ✓ Tự động gọi



### ➤ Cú pháp:

```
function name(parameter1, parameter2, parameter3) {  
    code to be executed  
}
```



Phần thứ tư

# CÁC CÂU LỆNH TRONG JAVASCRIPT

# 4



# Các câu lệnh trong Javascript

Giới thiệu các cấu trúc

- Cấu trúc tuần tự là các câu lệnh được theo trình tự xuất hiện của chúng
- Cấu trúc điều kiện là cấu trúc dùng để diễn đạt một việc sẽ được thực hiện khi một điều kiện cụ thể được thỏa mãn
- Cấu trúc lặp (Câu lệnh lặp) cho phép một lập trình viên xác định thứ tự thực thi các lệnh trong mỗi chương trình, thể hiện qua việc:
  - ✓ Bỏ qua một vài câu lệnh để thực thi các câu lệnh khác.
  - ✓ Lặp lại một hoặc khối lệnh trong khi điều kiện còn đúng.

```
var i;  
for (i = 0; i < cars.length; i++) {  
    text += cars[i] + "<br>";  
}
```

```
if (hour < 18) {  
    greeting = "Good day";  
} else {  
    greeting = "Good evening";  
}
```

```
var x, y;  
x = 5 + 6;  
y = x * 10;
```

# Các câu lệnh trong Javascript

Các cấu trúc điều kiện

➤ Cấu trúc **if ... else**

➤ Cấu trúc **switch ... case**

## Syntax

```
switch(expression) {  
  case n:  
    code block  
    break;  
  case n:  
    code block  
    break;  
  default:  
    code block  
}
```

## Syntax

```
if (condition1) {  
  block of code to be executed if condition1 is true  
} else if (condition2) {  
  block of code to be executed if the condition1 is false and condition2 is true  
} else {  
  block of code to be executed if the condition1 is false and condition2 is false  
}
```

```
switch (new Date().getDay()) {  
  case 6:  
    text = "Today is Saturday";  
    break;  
  case 0:  
    text = "Today is Sunday";  
    break;  
  default:  
    text = "Looking forward to the Weekend";  
}
```

```
if (time < 10) {  
  greeting = "Good morning";  
} else if (time < 20) {  
  greeting = "Good day";  
} else {  
  greeting = "Good evening";  
}
```



# Các câu lệnh trong Javascript

Các cấu trúc điều khiển

- Cấu trúc lặp **for**
- Cấu trúc lặp **for/in**: khi chúng ta muốn duyệt toàn bộ các thuộc tính của một đối tượng
- Cấu trúc lặp **while**
- Cấu trúc lặp **do...while**

Syntax

```
do {  
    code block to be executed  
}  
while (condition);
```

```
for (statement 1; statement 2; statement 3) {  
    code block to be executed  
}
```

**Statement 1** is executed before the loop (the code block) starts.

**Statement 2** defines the condition for running the loop (the code block).

**Statement 3** is executed each time after the loop (the code block) has been executed.

Syntax

```
while (condition) {  
    code block to be executed  
}
```

# Các câu lệnh trong Javascript

## Các câu lệnh khác

- Lệnh **break** dùng để kết thúc sớm vòng lặp
- Lệnh **continue** để bắt đầu một lần lặp mới
- Câu lệnh **return**: kết thúc hàm, trả về một kết quả cho chương trình gọi nó
- Câu lệnh **new**: khởi tạo cách thể hiện của một đối tượng

```
for (i = 0; i < 10; i++) {  
    if (i === 3) { break; }  
    text += "The number is " + i + "<br>";  
}
```

```
var x = myFunction(4, 3);    // Function is called, return value will end up in x
```

```
function myFunction(a, b) {  
    return a * b;           // Function returns the product of a and b  
}
```

```
for (i = 0; i < 10; i++) {  
    if (i === 3) { continue; }  
    text += "The number is " + i + "<br>";  
}
```

# Xin cảm ơn

# MÔN THIẾT KẾ WEB

BỘ MÔN KỸ THUẬT VÀ MẠNG MÁY TÍNH

