

Giới thiệu về JavaScript và đoạn mã JavaScript đầu tiên

Tìm hiểu khái niệm cơ bản về JavaScript để viết các JS trong tài liệu HTML, chú thích, lưu log và chạy Javascript ở Console

JavaScript là gì?

JavaScript (JS) là một ngôn ngữ lập trình thông dịch (*được dịch lúc chạy*), JavaScript ngày nay được ứng dụng rộng rãi trên nhiều môi trường khác nhau, nó tạo ra các ứng dụng chạy phía Server (*xem như NodeJS*), các ứng dụng di động (*xem React*)... Javascript được sử dụng rất phổ biến để tích hợp vào các trang web, chạy ngay trong trình duyệt phía client giúp cho Website có được cách tương tác linh hoạt hơn với người dùng.

Trong phần cơ bản về JS này chỉ trình bày JavaScript nhúng vào các trang web (HTML) chạy trên môi trường trình duyệt, tuy nhiên kiến thức cũng đủ làm cơ sở để học nâng cao với các Javascript Engine khác nhau (như NodeJS - V8).

Đọc thêm: Mỗi trình duyệt có một [JavaScript Engine](#), nó có chức năng phân tích mã JavaScript, tạo ra môi trường chạy mã JavaScript - như trong Google Chrome là Blink, FireFox là Gecko, Safari là Nitro, Microsoft Edge là Chakra ...

ECMAScript (ES) và Javascript: ES là các đặc tả kỹ thuật cho một ngôn ngữ kịch bản (Script) do tổ chức European Computer Manufacturers Association (ECMA) soạn thảo và ban hành thành tiêu chuẩn ISO (Hiện đang phiên bản ES6 - ECMA-262, nếu thích xem đầy đủ tải ở đây: [Standard ECMA-262](#)). Từ đặc tả kỹ thuật này một số ngôn ngữ kịch bản triển khai như **Javascript**, **JScript**, **ActionScript**. Như vậy Javascript có thể được coi là ES được mở rộng thêm (thêm các API tương tác với DOM).

Viết mã JavaScript đầu tiên

Cách 1 -Nhúng JS trong HTML Cách thứ nhất là mã JavaScript được viết nhúng trong văn bản HTML với thẻ `<script>`

Thẻ `<script>` này có thể đặt bên trong thẻ `<head>` hoặc `<body>` của văn bản HTML

```
<html>
  <head>
    <script >
      <!--Chỗ này viết code JavaScript-->
    </script>
  </head>

  <body>
    <script>
      <!--Chỗ này cũng viết được code JavaScript-->
    </script>
```

```
</body>
</html>
```

Chương trình Hello world! JavaScript

HTML sau có chứa JS, JS này có chức năng viết dòng **Hello World!** ở cửa sổ trình duyệt.

```
<html>
  <head> </head>
  <body>
    <script>
      document.write("Hello World!");
    </script>
  </body>
</html>
```

Hàm **document.write()** để viết một chuỗi ra văn bản HTML.

Hàm **document.write()** có thể sử dụng để xuất văn bản text thuần túy, mã HTML hoặc cả hai. Ví dụ định dạng văn bản viết ra bằng thẻ HTML **<h1>**

```
document.write("<h1>Hello World!</h1>");
```

Lưu ý: Thẻ **<script>** để chứa code JS có thể đặt bất kỳ đâu trong HTML, và khi trình duyệt tải HTML bắt gặp Javascript ở đâu sẽ chạy ngay code JS ở đó. Vì thế ý tưởng đặt thẻ **<script>** trong thẻ **<body>** và đặt ở cuối (gần thẻ đóng body) là rất tốt vì nó cải thiện tốc độ hiện thị trang (HTML tải xong thì chạy JS!).

Hàm alert

Bạn có thể sử dụng hàm **alert()** để hiện thị một hộp thoại (popup) thông báo. Bạn làm quen hàm này sớm vì mục đích kiểm thử quá trình chạy code JS của bạn viết ra.

```
<script type="text/javascript">
  alert("Đây là một thông báo!");
</script>
```

Trong ví dụ trên thẻ **<script>** có thuộc tính **type="text/javascript"**, đây là theo chuẩn cũ, từ HTML5 trở đi thì không cần dùng đến.

Tải mã JavaScript từ file ngoài

Cách 2 - Tải JS từ file Mã JavaScript có thể không cần viết trực tiếp cùng với file văn bản HTML. Các mã này được viết trong một file, thường file này có đặt tên với phần mở rộng là **.js** sau đó tải vào HTML bằng thẻ **<script>** với cú pháp:

```
<script src="địa-chỉ-file-js"></script>
```

Ví dụ mã JavaScript viết trong file **demo.js** sau đó file đó được nhúng vào HTML bằng thẻ **<script>** và vị trí file chỉ ra trong thuộc tính **src**

Bạn dùng trình soạn thảo text nào đó (như **notepad++**) tạo file **demo.js** có nội dung như sau:

```
alert("Đây là hộp thoại!");
```

Chú ý ở file **demo.js** không có thẻ **script**

Sau đó bạn muốn mã JS trong file đó chạy ở đâu trong HTML thì nhúng file JS vào đó bằng thẻ **script**:

```
<html>
  <head>
    <title> </title>
    <script src="demo.js"></script>
  </head>
  <body>
  </body>
</html>
```

Mã tải file ngoài trên bạn có thể đặt trong **head** hoặc **body**, bạn đặt ở vị trí nào mà mã JavaScript được chạy.

Sử dụng cách tải JavaScript từ file ngoài có một số lợi ích

- Phân chia độc lập giữa mã HTML và JavaScript để dễ đọc hơn
- Giúp cho mã HTML và JavaScript dễ bảo trì hơn
- Giúp trình duyệt cached file JavaScript để tăng tốc trang.

Ta cũng thường dùng cách này để nạp các thư viện Javascript từ các host khác nhau. Ví dụ thư viện sau <https://ajax.googleapis.com/ajax/libs/d3js/5.7.0/d3.min.js> bạn muốn tích hợp vào trang mà không cần thiết phải tải **.js** đó về.

```
<script src="https://ajax.googleapis.com/ajax/libs/d3js/5.7.0/d3.min.js"></script>

<script>
  //Sau 1s đổi nền trang sang màu đỏ - hãy tạo HTML và chạy thử xem
  d3.select("body").transition().delay(1000).style("background-color", "red");
</script>
```

Chú thích trong JavaScript

Chú thích trong JavaScript (các chữ, mã không được thi hành - bạn viết vào để ghi chú gợi nhớ khi đọc lại code). Bạn có thể viết một dòng chú thích sau ký hiệu **//** hoặc nhiều dòng chú thích nằm giữa **/* chú thích ... */**

```
<script>

  // Đây là một chú thích

  alert("Hộp thoại!");

  /* Đây là các
```

```
dòng chú thích  
trong JavaScript */
```

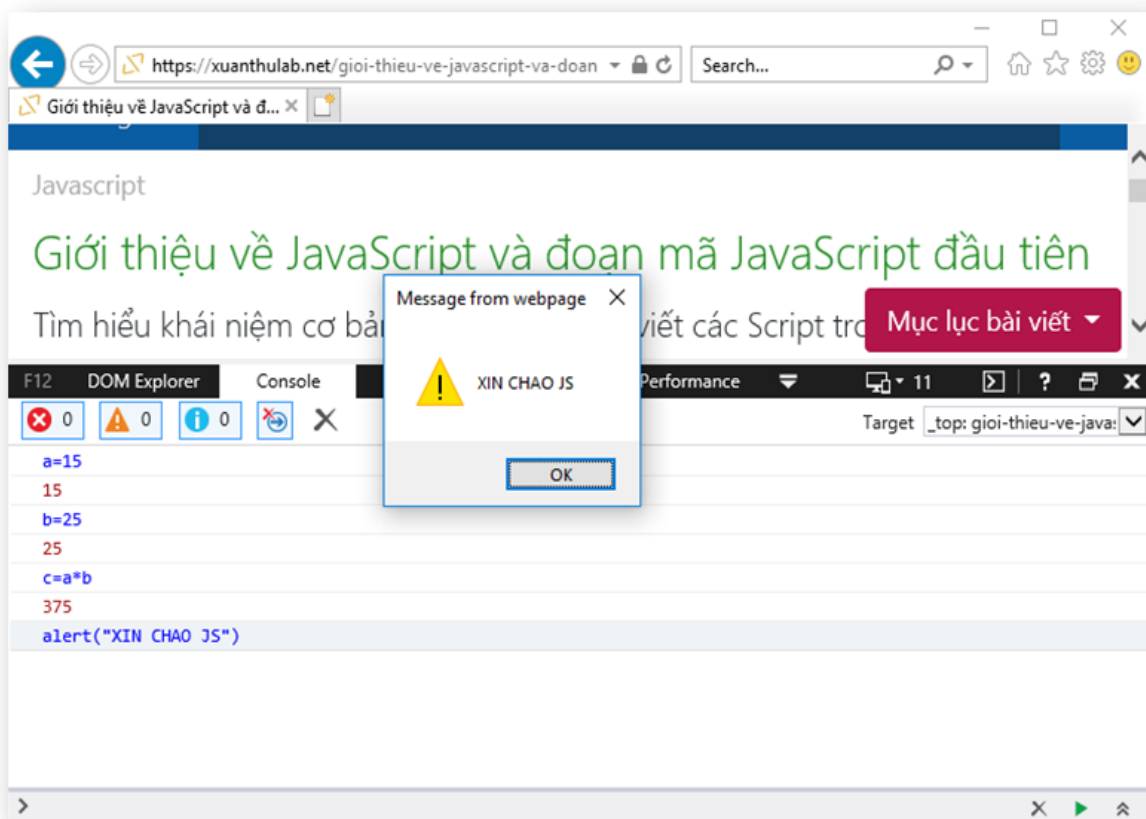
```
</script>
```

Chạy JS trong Console

Trong các trình duyệt để hỗ trợ nhà phát triển kiểm tra thử code Javascript, xem log ... chúng đều cung cấp một cửa sổ, nơi đó có một **dấu nhắc lệnh** để bạn gõ (hoặc dán - paste) mã script mà mã đó sẽ chạy luôn. Bạn có thể mở cửa sổ Console bằng cách:

- Trong Chrome, Firefox gõ phím: **CTRL+SHIFT+J**
- Trong IE nhấn **F12** để xuất hiện công cụ Developpe, sau đó nhấn **CTRL+2**

Ví dụ, mở **Console** trình duyệt của bạn ra, gõ thử như hình vẽ và xem JS nó hoạt động ra sao (ở hình vẽ có thực hiện gán biến $a = 15$, $b = 25$, sau đó thực hiện phép nhân, cuối cùng là popup một thông báo.



Log Javascript

Trong quá trình các đoạn code hoạt động, nếu bạn lưu lại các dấu vết ở bất kỳ thời điểm nào bạn hãy dùng cú pháp:

```
<script>  
  console.log("Dòng thông tin sẽ lưu ra log");
```

```
</script>
```

Tham số của `console.log` cũng có thể là một đối tượng bất kỳ (học về đối tượng sau)

```
<script>  
  var d = new Date();  
  console.log(d);  
</script>
```

Lúc nào muốn xem lại Log hãy mở cửa sổ `Console` của trình duyệt ra