



# Giáo trình JavaScript

....., tháng ... năm .....

# Giáo trình JavaScript

\*\*\*

CH-ƠNG 1 LỜI NÓI ĐẦU .....	4
CH-ƠNG 2 NHẬP MÔN JAVASCRIPT .....	5
2.1. Nhúng JavaScript vào file HTML.....	5
2.3. Thẻ <NOScript> và </NOSCRIPT>.....	6
2.3. Hiển thị một dòng text .....	7
2.4. Giao tiếp với người sử dụng .....	9
2.5. Điểm lại các lệnh và mở rộng .....	12
CH-ƠNG 3 BIẾN TRONG JAVASCRIPT .....	13
3.1. Biến và phân loại biến .....	13
3.2. Biểu diễn từ tổ trong JavaScript .....	13
3.3. Kiểu dữ liệu .....	13
1.1.1. Kiểu nguyên (Integer).....	14
1.1.2. Kiểu dấu phẩy động (Floating Point) .....	14
1.1.3. Kiểu logic (Boolean) .....	15
1.1.4. Kiểu chuỗi (String) .....	15
<b>2.    XÂY DỰNG CÁC BIỂU THỨC TRONG JAVASCRIPT .....</b>	<b>16</b>
<i>ĐỊNH NGHĨA VÀ PHÂN LOẠI BIỂU THỨC .....</i>	<i>16</i>
<i>CÁC TOÁN TỬ (OPERATOR).....</i>	<i>16</i>
2.1.1. Gán .....	16
2.1.2. So sánh.....	16
2.1.3. Số học.....	17
2.1.4. Chuỗi .....	17
2.1.5. Logic .....	17
2.1.6. Bitwise .....	18
<i>BÀI TẬP .....</i>	<i>19</i>
2.1.7. Câu hỏi .....	19
2.1.8. Trả lời .....	19
<b>3.    CÁC LỆNH .....</b>	<b>20</b>
<i>CÂU LỆNH ĐIỀU KIỆN.....</i>	<i>20</i>
<i>CÂU LỆNH LẶP .....</i>	<i>21</i>
3.1.1. Vòng lặp for .....	21
3.1.2. while.....	22
3.1.3. Break .....	22
3.1.4. continue .....	23

---

CÁC CÂU LỆNH THAO TÁC TRÊN ĐỐI TƯỢNG .....	23
3.1.5. for...in .....	23
3.1.6. new .....	25
3.1.7. this .....	26
3.1.8. with .....	27
CÁC HÀM (FUNCTIONS).....	28
CÁC HÀM CÓ SẴN .....	30
3.1.9. eval .....	30
3.1.10. parseInt.....	31
3.1.11. parseFloat.....	32
MẢNG (ARRAY) .....	33
SỰ KIỆN .....	35
BÀI TẬP .....	38
3.1.12. Câu hỏi .....	38
3.1.13. Trả lời .....	40
<b>4. CÁC ĐỐI TƯỢNG TRONG JAVASCRIPT .....</b>	<b>43</b>
ĐỐI TƯỢNG NAVIGATOR .....	45
ĐỐI TƯỢNG WINDOW.....	46
4.1.1. Các thuộc tính.....	46
4.1.2. Các ph-ong thức .....	47
4.1.3. Các ch-ong trình xử lý sự kiện.....	48
ĐỐI TƯỢNG LOCATION .....	48
ĐỐI TƯỢNG FRAME.....	49
4.1.4. Các thuộc tính.....	49
4.1.5. Các ph-ong thức .....	49
4.1.6. Sử dụng Frame .....	49
ĐỐI TƯỢNG DOCUMENT .....	51
4.1.7. Các thuộc tính.....	52
4.1.8. Các ph-ong thức .....	52
ĐỐI TƯỢNG ANCHORS .....	52
ĐỐI TƯỢNG FORMS .....	53
4.1.9. Các thuộc tính.....	53
4.1.10. Các ph-ong thức .....	53
4.1.11. Các ch-ong trình xử lý sự kiện.....	53
ĐỐI TƯỢNG HISTORY .....	54
4.1.12. Các thuộc tính.....	54
4.1.13. Các ph-ong thức .....	54
ĐỐI TƯỢNG LINKS .....	54

---

4.1.14.	Các thuộc tính.....	54
4.1.15.	Các ch-ong trình xử lý sự kiện.....	55
<b>ĐỐI T-ỢNG MATH</b> .....		<b>55</b>
4.1.16.	Các thuộc tính.....	55
4.1.17.	Các ph-ong thức .....	55
<b>ĐỐI T-ỢNG DATE</b> .....		<b>56</b>
4.1.18.	Các ph-ong thức .....	56
<b>ĐỐI T-ỢNG STRING</b> .....		<b>57</b>
4.1.19.	Các ph-ong thức .....	57
<b>CÁC PHẦN TỬ CỦA ĐỐI T-ỢNG FORM</b> .....		<b>58</b>
4.1.20.	Thuộc tính type .....	59
4.1.21.	Phần tử button .....	59
4.1.22.	Phần tử checkbox .....	60
4.1.23.	Phần tử File Upload .....	62
4.1.24.	Phần tử hidden .....	62
4.1.25.	Phần tử Password .....	62
4.1.26.	Phần tử radio .....	63
4.1.27.	Phần tử reset .....	65
4.1.28.	Phần tử select .....	65
<b>5.</b>	<b>MÔ HÌNH ĐỐI T-ỢNG (OBJECT MODEL)</b> .....	<b>75</b>
<b>ĐỐI T-ỢNG VÀ THUỘC TÍNH</b> .....		<b>75</b>
<b>TẠO CÁC ĐỐI T-ỢNG MỚI</b> .....		<b>76</b>
5.1.1.	Sử dụng khởi tạo đối t-ợng.....	76
5.1.2.	Sử dụng một hàm xây dựng(Constructor Function).....	77
5.1.3.	Lập mục lục cho các thuộc tính của đối t-ợng .....	78
5.1.4.	Định nghĩa thêm các thuộc tính cho một kiểu đối t-ợng.....	79
5.1.5.	Định nghĩa các cách thức .....	79
5.1.6.	Sử dụng cho các tham chiếu đối t-ợng (Object References) .....	80
5.1.7.	Xoá đối t-ợng .....	80
<b>6.</b>	<b>BẢNG TỔNG KẾT CÁC TỪ KHOÁ</b> .....	<b>81</b>
<b>7.</b>	<b>TỔNG KẾT</b> .....	<b>82</b>

---

## CH-ƠNG 1 LỜI NÓI ĐẦU

Với HTML and Microsoft FrontPage bạn đã biết cách tạo ra trang Web - tuy nhiên chỉ mới ở mức biểu diễn thông tin chứ ch-a phải là các trang Web động có khả năng đáp ứng các sự kiện từ phía ng-ời dùng. Hãng Netscape đã đ-a ra ngôn ngữ script có tên là LiveScript để thực hiện chức năng này. Sau đó ngôn ngữ này đ-ợc đổi tên thành JavaScript

để tận dụng tính đại chúng của ngôn ngữ lập trình Java. Mặc dù có những điểm t-ơng

đồng giữa Java và JavaScript, nh-ng chúng vẫn là hai ngôn ngữ riêng biệt.

JavaScript là ngôn ngữ d-ới dạng script có thể gắn với các file HTML. Nó không đ-ợc biên dịch mà đ-ợc trình duyệt diễn dịch. Không giống Java phải chuyển thành các mã để biên dịch, trình duyệt đọc JavaScript d-ới dạng mã nguồn. Chính vì vậy bạn có thể dễ dàng học JavaScript qua ví dụ bởi vì bạn có thể thấy cách sử dụng JavaScript trên các trang Web.

JavaScript là ngôn ngữ dựa trên đối t-ợng, có nghĩa là bao gồm nhiều kiểu đối t-ợng, ví dụ đối t-ợng **Math** với tất cả các chức năng toán học. Tuy vậy JavaScript không là ngôn ngữ

h-ớng đối t-ợng nh- C++ hay Java do không hỗ trợ các lớp hay tính thừa kế.

JavaScript có thể đáp ứng các sự kiện nh- tải hay loại bỏ các form. Khả năng này cho phép

JavaScript trở thành một ngôn ngữ script động.

Giống với HTML và Java, JavaScript đ-ợc thiết kế độc lập với hệ điều hành. Nó có thể chạy trên bất kỳ hệ điều hành nào có trình duyệt hỗ trợ JavaScript. Ngoài ra JavaScript giống Java ở khía cạnh an ninh: JavaScript không thể đọc và viết vào file của ng-ời dùng.

Các trình duyệt web nh- Netscape Navigator 2.0 trở đi có thể hiển thị những câu lệnh

JavaScript đ-ợc nhúng vào trang HTML. Khi trình duyệt yêu cầu một trang, server sẽ gửi đầy đủ nội dung của trang đó, bao gồm cả HTML và các câu lệnh JavaScript qua mạng tới client. Client sẽ đọc trang đó từ đầu đến cuối, hiển thị các kết quả của HTML và xử lý các câu lệnh JavaScript khi nào chúng xuất hiện.

Các câu lệnh JavaScript đ-ợc nhúng trong một trang HTML có thể trả lời cho các sự kiện của ng-ời sử dụng nh- kích chuột, nhập vào một form và điều h-ớng trang. Ví dụ bạn có thể kiểm tra các giá trị thông tin mà ng-ời sử dụng đ-a vào mà không cần đến bất cứ một quá trình truyền trên mạng nào. Trang HTML với JavaScript đ-ợc nhúng sẽ kiểm tra các giá trị đ-ợc đ-a vào và sẽ thông báo với ng-ời sử dụng khi giá trị đ-a vào là không hợp lệ.

Mục đích của phần này là giới thiệu về ngôn ngữ lập trình JavaScript để bạn có thể viết các script vào file HTML của mình.

---

### 5.1.6. Sử dụng cho các tham chiếu đối tượng (Object References)

JavaScript có một từ khoá đặc biệt là **this** mà bạn có thể sử dụng nó cùng với một cách thức để gọi tới đối tượng hiện thời. Ví dụ, giả sử bạn có một hàm validate dùng để xác nhận giá trị thuộc tính của một đối tượng nằm trong một khoảng nào đó:

```
function validate(obj, lowval, hival){
    if ( (obj.value<lowval)|| (obj.value>hival) )
        alert("Invalid value!")
}
```

Sau đó bạn có thể gọi hàm **validate** từ mỗi thẻ sự kiện **onChange**:

```
<INPUT TYPE="TEXT" NAME="AGE" SIZE=3
      onChange="validate(this,18,99)" >
```

Khi liên kết với một thuộc tính form, từ khoá **this** có thể gọi tới form cha của đối tượng hiện thời. Trong ví dụ sau, **myForm** có chứa đối tượng **Text** và một nút bấm. Khi người sử dụng kích vào nút bấm, trường text sẽ hiển thị tên form. Thẻ sự kiện **onClick** của nút bấm sử dụng **this.form** để gọi tới form cha là **myForm**.

```
<FORM NAME="myForm">
    Form name:<INPUT TYPE="text" NAME="text1"
                VALUE="Beluga">
<P>
    <INPUT TYPE="button" NAME="button1"
          value="Show Form Name"
          onClick="this.form.text1.value=this.form.name">
</FORM>
```

### 5.1.7. Xoá đối tượng

Trong JavaScript cho Navigator 2.0, bạn không thể xoá các đối tượng-chúng vẫn tồn tại trong khi bạn đã rời khỏi trang đó. Trong khi JavaScript cho Navigator 3.0 cho phép bạn có thể xoá một đối tượng bằng cách đặt cho nó trở tới giá trị Null (nếu nh- đó là lần cuối cùng gọi tới đối tượng). JavaScript sẽ đóng đối tượng đó ngay lập tức thông qua biểu thức gán.

---

## 6. BẢNG TỔNG KẾT CÁC TỪ KHOÁ

Sau đây là các từ được định nghĩa là một phần trong ngôn ngữ JavaScript và không đ-ợc sử dụng là tên biến:

<b>abstract</b>	<b>eval</b>	<b>int</b>	<b>static</b>
<b>boolean</b>	<b>extends</b>	<b>interface</b>	<b>super</b>
<b>break</b>	<b>false</b>	<b>long</b>	<b>switch</b>
<b>byte</b>	<b>final</b>	<b>native</b>	<b>synchronized</b>
<b>case</b>	<b>finally</b>	<b>new</b>	<b>this</b>
<b>catch</b>	<b>float</b>	<b>null</b>	<b>throw</b>
<b>char</b>	<b>for</b>	<b>package</b>	<b>throws</b>
<b>class</b>	<b>function</b>	<b>parseFloat</b>	<b>transient</b>
<b>const</b>	<b>goto</b>	<b>parseInt</b>	<b>true</b>
<b>continue</b>	<b>if</b>	<b>private</b>	<b>try</b>
<b>default</b>	<b>implements</b>	<b>protected</b>	<b>var</b>
<b>do</b>	<b>import</b>	<b>public</b>	<b>void</b>
<b>double</b>	<b>in</b>	<b>return</b>	<b>while</b>
<b>else</b>	<b>instanceof</b>	<b>short</b>	<b>with</b>

## 7. TỔNG KẾT

Nh- vậy, tài liệu không những đã giới thiệu sơ qua về JavaScript, mà nó còn là sách tham khảo hết sức hữu ích để phát triển ứng dụng của bạn.

Bạn có thể tham khảo toàn diện JavaScript trong quyển *Teach Yourself JavaScript in 14 Days*, hoặc *JavaScript Guide*

Do JavaScript là ngôn ngữ còn mới và có sự thay đổi nhanh chóng, bạn nên đến với trang Web của hãng Netscape ( <http://www.netscape.com> ) để có các thông tin mới nhất về ngôn ngữ này.

Tailieu.vn