cách sử dụng biến trong javascript

### [Biến và khai báo biến trong javascript - freetuts](https://www.google.com.vn/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwj0hMeYrevSAhXINJQKHQERCvAQFggXMAA&url=http%3A%2F%2Ffreetuts.net%2Fbien-va-khai-bao-bien-trong-javascript-265.html&usg=AFQjCNEZ18Kle4eL-ImRtq8AEdbzGr6Yyg&sig2=UKAWsRxSjOZWZ_VJoKwc9w&bvm=bv.150475504,d.dGo)

freetuts.net › Javascript › Javascript căn bản

6 thg 1, 2015 - Trong bài này ta sẽ tìm hiểu các khai báo **biến** và gán giá trị cho **biến trong javascript**. , đồng thời học thêm **cách dùng** hàm document.write để ...

### [Javascript - Biến trong JavaScript | 208 bài học Javascript miễn phí ...](https://www.google.com.vn/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=0ahUKEwj0hMeYrevSAhXINJQKHQERCvAQFggfMAE&url=http%3A%2F%2Fvietjack.com%2Fjavascript%2Fbien_trong_javascript.jsp&usg=AFQjCNG1_gbuUOonm4tgFgLLlD5R8W6-vw&sig2=lpB3TxuMD_Aylc-Wj1aZOg&bvm=bv.150475504,d.dGo)

vietjack.com/javascript/bien\_trong\_javascript.jsp

Bạn có thể đặt dữ liệu vào **trong** nơi chứa này và sau đó tham chiếu đến dữ liệu này một **cách** đơn giản là đặt tên cho nơi chứa. Trước khi bạn **sử dụng** một **biến** ...

### [Các biến và các kiểu dữ liệu trong JavaScript - VOER](https://www.google.com.vn/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&cad=rja&uact=8&ved=0ahUKEwj0hMeYrevSAhXINJQKHQERCvAQFgglMAI&url=https%3A%2F%2Fvoer.edu.vn%2Fm%2Fcac-bien-va-cac-kieu-du-lieu-trong-javascript%2F43e30b08&usg=AFQjCNH5Qju7WbAQNDPDe4asfaPuHI_-qA&sig2=NCvjqxP0iDLSU9-I4bf73g&bvm=bv.150475504,d.dGo)

https://voer.edu.vn/m/cac-bien-va-cac-kieu-du-lieu-trong-javascript/43e30b08

Các **biến** và các kiểu dữ liệu **trong JavaScript** (giáo trình - tài liệu - học liệu từ VOER) ... Chúng ta cũng có thể khai báo **biến** bằng **cách** gán giá trị cho nó mà không ... Các **biến** toàn cục không cần thiết phải **sử dụng** từ khóa var khi khai báo.

### [Cách sử dụng biến (Variable) trong JavaScript - Web cơ bản](http://webcoban.vn/javascript/cach-su-dung-bien-variable-trong-javascript.html)

webcoban.vn/javascript/cach-su-dung-bien-variable-trong-javascript.html

**Cách sử dụng biến** Variable **trong JavaScript**, **biến** là gì, **cách** khai báo **biến trong JavaScript**.

### [Sử Dụng Biến Trong Javascript - JavaScript Cơ Bản 1 - Codehub.vn](http://www.codehub.vn/JavaScript-Co-Ban-1-1/Su-Dung-Bien-Trong-Javascript)

www.codehub.vn/JavaScript-Co-Ban-1-1/Su-Dung-Bien-Trong-Javascript

**Biến** được **sử dụng** để lưu trữ một (hoặc một vài) giá trị nào đó. Giá trị của **biến** có thể thay đổi **trong**quá trình chạy mã lệnh. ...

### [Khai báo biến trong javascript và cách dùng biến trong ... - Học ASP.net](http://hocasp.net/6-khai-bao-bien-trong-javascript/)

hocasp.net/6-khai-bao-bien-trong-javascript/

26 thg 4, 2015 - 1 Khai báo **biến**; 2 **Cách** đặt tên **biến**; 3 Kiểu dữ liệu của **biến**; 4 Xác định kiểu của**biến**. 4.1 Giá trị của **biến** = undefined. 5 **Sử dụng biến trong** ...

### [Sử Dụng Biến Trong Javascript | Tien Thanh](http://tienthanhbk.blogspot.com/2008/06/s-dng-bin-trong-javascript.html)

tienthanhbk.blogspot.com/2008/06/s-dng-bin-trong-javascript.html

Nhận xét này đã bị tác giả xóa. Sang HT 00:43 8 tháng 8, 2013. Cảm ơn về bài viết, cho mình hỏi thêm về việc **sử dụng biến** GIATRI từ hàm này qua hàm khác tí, ...

### [Hướng dẫn khai báo biến trong JavaScript - Kênh Lập Trình](http://kenhlaptrinh.net/khai-bao-bien-trong-javascript/)

kenhlaptrinh.net/khai-bao-bien-trong-javascript/

27 thg 10, 2016 - Trong bài viết này mình sẽ hướng dẫn các bạn **cách** khai báo **biến** ... Để khai báo một **biến trong JavaScript** ta **sử dụng** từ khóa var tenbien;.

### [Bài 2: Phạm vi hoạt động của biến trong Javascript - Thien An Blog](https://thienanblog.com/javascript/javascript-co-ban/bai-2-pham-vi-hoat-dong-cua-bien-trong-javascript/)

https://thienanblog.com › Javascript › Javascript cơ bản

3 thg 5, 2015 - Ở bài Tìm hiểu kiểu dữ liệu **trong Javascript** chúng ta đã được biết qua **cách** để khai báo và **sử dụng** Javascript rồi. Trong đó, chúng ta cũng tìm ...

### [Biến Trong Ngôn Ngữ Lập Trình JavaScript :: Bài viết :: STDIO](https://www.stdio.vn/articles/read/453/bien-trong-ngon-ngu-lap-trinh-javascript)

https://www.stdio.vn/articles/read/453/bien-trong-ngon-ngu-lap-trinh-javascript

27 thg 10, 2015 - Vậy **trong JavaScript** cú phái khai báo **biến** việc thao tác, **sử dụng** ... làm việc với vùng nhớ một **cách** dễ dàng bằng **cách** chỉ gọi định danh này.

### [Bài 4: Biến và hằng trong Javascript - Vietpro Academy](https://hocthietkeweb.net.vn/bai-4-bien-va-hang-trong-javascript.html)

https://hocthietkeweb.net.vn › Video Bài giảng › Video Học Thiết kế web

5 thg 8, 2015 - **Trong** mọi trường hợp, nơi bạn thấy trước một **cách** hợp lý rằng các giá ... giá trị cho**biến** và hằng số; – làm thế nào để **sử dụng** các **biến trong** ...

### [[JS] Hiểu thế nào cho đúng về biến this trong JavaScript? | Small Feet](https://dofeet.wordpress.com/2015/03/07/js-hieu-the-nao-cho-dung-ve-bien-this-trong-javascript/)

https://dofeet.wordpress.com/.../js-hieu-the-nao-cho-dung-ve-bien-this-trong-javascri...

7 thg 3, 2015 - Một **trong** những điểm dễ nhầm lẫn đó là **biến** this vì **trong** JS nó không chỉ đơn giản ... Các ngữ cảnh này lại được sắp xếp để thực hiện chương trình một **cách** tuần tự. ... mX = "I love**JavaScript**" // **sử dụng** đối tượng toàn cục.

### [Hỏi Cách sử dụng & khai báo biến chung trong Javascript - SinhvienIT](http://sinhvienit.net/forum/cach-su-dung-khai-bao-bien-chung-trong-javascript.272884.html)

sinhvienit.net › ... › Ngành công nghệ phần mềm › Lập trình ứng dụng Desktop › Java

10 thg 8, 2013 - **Cách sử dụng** & khai báo **biến** chung **trong Javascript**, Mình có đoạn code như bên dưới, mình có thắc mắc là khai báo làm sao để **sử dụng** ...

### [Javascript cơ bản - Hành trình của một Anh Hùng - Thạch Phạm](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html)

https://thachpham.com/.../javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung....

Chuyển đến [1.2.Thao tác với **biến**](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#ftoc-heading-3) - Để chứa đồ **trong** tủ, bạn cần phải tìm 1 ngăn tủ rỗng, dán tên ... Sau khi khai báo, **dùng** dấu = để gán giá trị ... Sau khi đặt xong **biến**, bạn có thể thử in ra bằng **cách sử dụng** hàm alert() để xem giá trị.

Bạn đã truy cập trang này 3 lần. Lần truy cập cuối: 19/03/2017

### [Biến trong Javascript - VinaInfotech](http://vinainfotech.com/bien-trong-javascript-179.html)

vinainfotech.com/bien-trong-javascript-179.html

**Biến trong Javascript**. ... Học CSS - Javascript ... vào **biến**), Lúc này D được coi là **biến**, khi cần thì mở két lấy ra **dùng**(Giá trị **biến** được **sử dụng** – truy cập **biến**).

### [Kiểm tra kiểu dữ liệu trong javascript | Tek - Chia sẻ kiến thức công ...](http://tek.eten.vn/kiem-tra-kieu-du-lieu-trong-javascript)

tek.eten.vn/kiem-tra-kieu-du-lieu-trong-javascript

7 thg 8, 2012 - Để xác định chính xác kiểu của một **biến javascript** lúc runtime **trong** một ... Tôi sẽ **sử dụng** các **cách** thức ở trên để kiểm tra kiểu của **biến** DOM ...

### [Khai báo và sử dụng Biến trong JavaScript. - Sinh Viên CNTT](http://sinhviencntt.net/khai-bao-vao-su-dung-bien-trong-javascript/)

sinhviencntt.net/khai-bao-vao-su-dung-bien-trong-javascript/

22 thg 1, 2017 - [Học JavaScript] Bài 7 Khai báo vào **sử dụng Biến trong JavaScript** ... + Khai báo từng **biến** đơn riêng từng dòng lệnh, **cách** này hay **dùng** vì ...

### [[Hỏi] - Cách sử dụng & khai báo biến chung trong Javascript - Diễn ...](http://www.ddth.com/showthread.php/1197829-H%E1%BB%8Fi-C%C3%A1ch-s%E1%BB%AD-d%E1%BB%A5ng-amp-khai-b%C3%A1o-bi%E1%BA%BFn-chung-trong-Javascript)

www.ddth.com › Diễn đàn › Lập trình › Các ngôn ngữ lập trình › Java

10 thg 8, 2013 - 3 bài đăng - ‎2 tác giả

Mình có đoạn code như bên dưới, mình có thắc mắc là khai báo làm sao để **sử dụng biến** GIATRI từ hàm jx . load() ở trên làm điều kiện lặp cho ...

### **[PDF]**[6. Javascript Basic.pdf - Khoa Công Nghệ Thông Tin | Đại học Tôn ...](http://it.tdt.edu.vn/~pnnam/BG_503023/BG_503023/6.%20Javascript%20Basic.pdf)

it.tdt.edu.vn/~pnnam/BG\_503023/BG\_503023/6.%20Javascript%20Basic.pdf

**Cách** 1 : Nhúng đoạn **javascript** vào **trong** file HTML. – **Cách** 2 : Viết ... **Javascript** không cần khai báo **biến** vẫn có thể **sử dụng** được. • Tên **biến** phân biệt hoa ...

### [Biến trong Javascript - Logo](http://congdongcoder.com/bien-trong-javascript/)

congdongcoder.com/bien-trong-javascript/

Cũng giống như các ngôn ngữ khác, **biến trong javascript dùng** để chứa giá trị. ... Có 2 **cách** khai báo **biến**, đó là khai báo thông thường và khai báo khởi tạo ...

# **Bài 02: Biến và khai báo biến trong javascript**

Trong bài này ta sẽ tìm hiểu các khai báo biến và gán giá trị cho biến trong javascript. , đồng thời học thêm cách dùng hàm document.write để in giá trị biến ra ngoài trình duyệt . Việc khai báo biến trong javascript cũng giống như [khai báo biến trong php](http://freetuts.net/khai-bao-bien-va-hang-so-trong-php-2.html) vậy, tuy cú pháp nó không giống nhau nhưng về cách gán giá trị thì y chang.

Nội dung chính

* [1. Khai báo biến trong javascript](http://freetuts.net/bien-va-khai-bao-bien-trong-javascript-265.html#goto-h2-0)
* [2. Gán giá trị cho biến trong javascript](http://freetuts.net/bien-va-khai-bao-bien-trong-javascript-265.html#goto-h2-1)
* [3. Gán kiểu giá trị cho biến](http://freetuts.net/bien-va-khai-bao-bien-trong-javascript-265.html#goto-h2-2)
* [4. In giá trị của biến ra trình duyệt](http://freetuts.net/bien-va-khai-bao-bien-trong-javascript-265.html#goto-h2-3)
* [Lời kết](http://freetuts.net/bien-va-khai-bao-bien-trong-javascript-265.html#goto-h2-4)

## **1. Khai báo biến trong javascript**

Để khai báo một biến ta sử dụng từ khóa var tenbien. Ví dụ:

|  |  |
| --- | --- |
| 1 | var username; |

Tuy nhiên không phải muốn đặt tên biến gì cũng được mà phải tuân theo quy tắc sau đây:

* Tên biến phải là các chữ không dấu viết hoa hoặc viết thường, các chữ số từ 0-9 và dấu gạch dưới (\_).
* Tên biến bắt đầu phải là chữ hoặc dấu gạch dưới (\_), nếu bắt đầu bằng số là sai
* Tên biến có thể đặt dài hay ngắn tùy vào lập trình viên.

Ví dụ:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | // Đúng  var username;    // Đúng  var \_username;    // Đúng  var \_\_username;    // Đúng  var username90;    // SAI  var 90thehalfheart; |

Ngoài cách khai báo trên ta còn có cách khai báo biến liên tiếp cách nhau bởi dấu phẩy, ví dụ:

|  |  |
| --- | --- |
| 1 | var username, \_username, \_\_username, username90; |

## **2. Gán giá trị cho biến trong javascript**

Để gán giá trị cho biến ta dùng dấu bằng (=) để gán vế phải vào vế trái, ví dụ gán giá trị thehalfheart vào biến username thì ta viết là username = 'thehalfheart'.

Có hai cách gán thông dụng đó là vừa khai báo biến vừa gán giá trị và khai báo rồi mới gán giá trị:

Ví dụ vừa khai báo vừa gán giá trị:

|  |  |
| --- | --- |
| 1 | var username = 'thehalfheart'; |

Ví dụ khai báo xong mới gán giá trị:

|  |  |
| --- | --- |
| 1  2 | var username;  username = 'thehalfheart'; |

## **3. Gán kiểu giá trị cho biến**

Giống như php, để xác định biến có kiểu dữ liệu là gì thì ta dựa vào giá trị mà nó đang có, có nghĩa khi bạn gán một con số không có dấu chấm động thì nó sẽ là kiểu INT, nếu gán một chuỗi thì là kiểu String.

Ví dụ:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | // Biến website đag kiểu String  var website = 'thehalfehart';    // Biến website chuyển sang kiểu INT  website = 12;    // Biến website chuyển sang kieeur float  website = 12.5; |

Trong Javascript bạn sẽ có các kiểu dữ liệu thông dụng như kiểu chuỗi (*String*), số (*Number*), mảng (*Array*), đối tượng (*Object*). Chi tiết từng kiểu dữ liệu chúng ta sẽ được học trong các bài tiếp theo.

## **4. In giá trị của biến ra trình duyệt**

Để in giá trị của biến hoặc một chuỗi nào đó ra trình duyệt thì ta sử dụng hàm document.write(value).

Ví dụ: [XEM DEMO](http://freetuts.net/editor.html?id=178)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | <html>  <head>  <title></title>  </head>  <body>  <script language="javascript">  var website = 'freetuts.net';  document.write(website);  </script>  </body>  </html> |

## **Lời kết**

Bài này mình dừng ở đây, hy vọng qua những ví dụ trên giúp bạn hiểu được cách sử dụng biến căn bản trong javascript, tuy nói là căn bản nhưng nó là nền tảng đấy nhé các bạn, hãy xem các ví dụ và thực hành theo, kết hợp nhuần nhuyễn giữa các phần để tự đưa ra những ví dụ cho riêng mình.

Câu hỏi và bài tập thực hành

* [Hãy khai báo 5 biến đúng và 5 biến sai trong Javascript](http://freetuts.net/bai-tap/hay-khai-bao-5-bien-dung-va-5-bien-sai-trong-javascript-2.html)
* [Trong Javascript có thể gán giá trị cho biến lúc khai báo không?](http://freetuts.net/bai-tap/trong-javascript-co-the-gan-gia-tri-cho-bien-luc-khai-bao-khong-23.html)
* [Javascript - khai báo biến không có từ var được không?](http://freetuts.net/bai-tap/javascript-khai-bao-bien-khong-co-tu-var-duoc-khong-36.html)

Ở bài viết này dùng:

<h1>Bài 02: Biến và khai báo biến trong javascript</h1>

<h2>1. Khai báo biến trong javascript</h2>

<h2>2. Gán giá trị cho biến trong javascript</h2>

<h2>3. Gán kiểu giá trị cho biến</h2>

<h2>4. In giá trị của biến ra trình duyệt</h2>

<h2">Lời kết</h2>

nhận xét: thẻ <h1> thì được dùng để chứa tiêu đề bài viết, tiêu đề luôn luôn chứa keyword và chỉ dùng 1 lần

thẻ<h2> thì được dùng nhiều lần và chứa các mục của bài viết.

bên cạnh đó có rất nhiều đường link liên quan đến keyword trên trang

# Biến trong JavaScript

## Kiểu dữ liệu JavaScript

Một trong những nét đặc thù chủ yếu nhất của một ngôn ngữ chương trình là tập hợp các kiểu dữ liệu nó hỗ trợ. Đây là kiểu các giá trị mà có thể được biểu diễn và được thao tác trong ngôn ngữ chương trình.

JavaScript cho bạn làm việc với 3 kiểu dữ liệu gốc sau:

* **Số,** ví dụ: 123, 120.50, …
* **Chuỗi** văn bản, ví dụ: "This text string", …
* **Boolean** ví dụ: true hoặc false.

JavaScript cũng định nghĩa hai kiểu dữ liệu thông thường, **null** và **undefined**, mỗi kiểu này chỉ định nghĩa một giá trị đơn. Bổ sung cho những kiểu dữ liệu này, JavaScript hỗ trợ một kiểu dữ liệu hỗn hợp được gọi là **object** (đối tượng). Chúng ta sẽ bàn luận chi tiết về các đối tượng trong một chương riêng.

**Ghi chú** − Java không phân biệt giá trị integer và giá trị dấu chấm động (floating-point). Tất cả các số trong JavaScript được biểu diễn như là giá trị dấu chấm động. JavaScript biểu diễn các số sử dụng định dạng dấu chấm động 64 bit được định nghĩa bởi chuẩn IEEE 754.

## Các biến trong JavaScript

Giống như nhiều ngôn ngữ chương trình khác, JavaScript có các biến. Biến có thể được xem là một nơi chứa (container) được đặt tên. Bạn có thể đặt dữ liệu vào trong nơi chứa này và sau đó tham chiếu đến dữ liệu này một cách đơn giản là đặt tên cho nơi chứa.

Trước khi bạn sử dụng một biến trong chương trình JavaScript, bạn phải khai báo nó. Biến được khai báo với từ khóa **var** như sau:

<script type="text/javascript">  
 <!--  
 var money;  
 var name;  
 //-->  
</script>

Bạn cũng có thể khai báo nhiều biến với cùng từ khóa **var** như sau:

<script type="text/javascript">  
 <!--  
 var money, name;  
 //-->  
</script>

Lưu giữ một giá trị trong một biến được gọi là **khởi tạo biến**. Bạn có thể khởi tạo biến tại thời điểm tạo biến hoặc tại thời điểm sau khi bạn cần biến đó.

Ví dụ, bạn có thể tạo một biến tên money và gán giá trị 2000.50 cho nó sau đó. Với biến khác, bạn có thể gán một giá trị tại thời điểm khởi tạo như sau:

<script type="text/javascript">  
 <!--  
 var name = "Ali";  
 var money;  
 money = 2000.50;  
 //-->  
</script>

**Ghi chú** − Chỉ sử dụng từ khóa **var** cho khai báo hoặc khởi tạo biến, một lần cho bất kỳ tên biến nào trong tài liệu. Bạn không nên khai báo lại cùng một biến hai lần.

JavaScript là ngôn ngữ **untyped** (không định kiểu). Nghĩa là một biến JavaScript có thể giữ một giá trị của bất kỳ kiểu dữ liệu nào. Không giống nhiều ngôn ngữ khác, bạn không phải nói cho JavaScript trong suốt quá trình khai báo biến về kiểu giá trị mà biến đó giữ. Kiểu giá trị của biến có thể thay đổi trong suốt quá trình thực thi một chương trình và JavaScript chăm sóc nó một cách tự động.

## Phạm vi biến trong JavaScript

Phạm vi của một biến là vùng chương trình trong đó nó được định nghĩa. Biến JavaScript chỉ có hai phạm vi sau:

* **Biến Global** − Một biến Global có phạm vi chung, nghĩa là nó có thể được định nghĩa bất cứ đâu trong JavaScript code.
* **Biến Local** − Một biến Local sẽ chỉ nhìn thấy trong một hàm, nơi nó được định nghĩa. Các tham số hàm luôn luôn là nội bộ cho hàm đó.

Trong thân của một hàm, một biến Local có quyền ưu tiên cao hơn biến Global với cùng một tên. Nếu bạn khai báo một biến Local hoặc tham số hàm với cùng tên với biến Global, bạn ẩn biến Global đó một cách hiệu quả. Bạn xem ví dụ sau:

<script type="text/javascript">  
 <!--  
 var myVar = "global"; // Khai báo một biến Global  
 function checkscope( ) {  
 var myVar = "local"; // Khai báo một biến Local  
 document.write(myVar);  
 }  
 //-->  
</script>

Nó sẽ cho kết quả sau:

local

## Tên biến trong JavaScript

Trong khi đặt tên biến trong JavaScript, bạn nên nhớ các quy tắc sau:

* Bạn không nên sử dụng bất kỳ từ khóa dành riêng nào cho một tên biến. Những từ khóa này được đề cập trong phần tới. Ví dụ, các tên biến **break** hoặc **boolean** là không hợp lệ.
* Tên biến JavaScript không nên bắt đầu với các số (0-9). Chúng phải bắt đầu với một chữ cái hoặc ký tự dấu gạch dưới. Ví dụ, **123test** là tên biến không hợp lệ nhưng **\_123test** là hợp lệ.
* Tên biến JavaScript là phân biệt kiểu chữ. Ví dụ, **Name** và **name** là hai biến khác nhau.

## Từ khóa dành riêng trong JavaScript

Bảng dưới đây liệt kê danh sách các từ khóa dành riêng trong JavaScript. Chúng không thể được sử dụng như là các biến, hàm, phương thức, nhãn vòng lặp, hoặc bất kỳ tên đối tượng nào khác trong JavaScript.

|  |  |  |  |
| --- | --- | --- | --- |
| abstract  boolean  break  byte  case  catch  char  class  const  continue  debugger  default  delete  do  double | else  enum  export  extends  false  final  finally  float  for  Hàm  goto  if  implements  import  in | instanceof  int  interface  long  native  new  null  package  private  protected  public  return  short  static  super | switch  synchronized  this  throw  throws  transient  true  try  typeof  var  void  volatile  while  with |

Loạt bài hướng dẫn **học lập trình Javascript cơ bản và nâng cao** của chúng tôi dựa trên nguồn tài liệu của: Tutorialspoint.com

Follow [**https://www.facebook.com/vietjackteam/**](https://www.facebook.com/vietjackteam/) để tiếp tục theo dõi các loạt bài mới nhất về Java,C,C++,Javascript,HTML,Python,Database,Mobile.... mới nhất của chúng tôi.

**Các bài học JavaScript khác tại VietJack**:

* [Javascript - Cú pháp](http://vietjack.com/javascript/cu_phap_cua_javascript.jsp)
* [Javascript - Kích hoạt](http://vietjack.com/javascript/kich_hoat_javascript_trong_cac_trinh_duyet.jsp)
* [Javascript - Vị trí trong HTML File](http://vietjack.com/javascript/vi_tri_javascript_trong_html_file.jsp)
* [Javascript - Biến](http://vietjack.com/javascript/bien_trong_javascript.jsp)
* [Javascript - Toán tử](http://vietjack.com/javascript/toan_tu_trong_javascript.jsp)
* [Javascript - Lệnh If...Else](http://vietjack.com/javascript/lenh_ifelse_trong_javascript.jsp)

Cách dùng thẻ ở bài viết này:

<h1>Biến trong JavaScript</h1>

<h2>Kiểu dữ liệu JavaScript</h2>

<h2>Các biến trong JavaScript</h2>

<h2>Phạm vi biến trong JavaScript</h2>

<h2>Tên biến trong JavaScript</h2>

<h2>Từ khóa dành riêng trong JavaScript</h2>

# **Các biến và các kiểu dữ liệu trong JavaScript**

# **Các biến (Variables)**

Biến là một tham chiếu đến một vị trí trong bộ nhớ. Nó dùng để chứa các giá trị có thể thay đổi khi script đang được thực thi. Chúng ta phải đặt tên cho biến. Chẳng hạn ta có thể tạo một biến có tên First. Tại mỗi thời điểm thực hiện, biến có thể chứa một giá trị mới. Khi chúng ta muốn xem giá trị của biến hoặc thay đổi giá trị của nó, ta chỉ cần dùng tên của biến.

Các biến thường có các quy ước đặt tên như sau:

* Tên biến phải bắt đầu bằng một chữ cái hoặc ký tự gạch dưới ("\_")
* Tên biến có thể chứa chữ số
* JavaScript phân biệt rõ chữ hoa và chữ thường, vì vậy tên biến bao gồm các ký tự từ "A" đến "Z" (chữ hoa) và các ký tự từ "a" đến "z" (chữ thường)

## **Khai báo biến**

Sử dụng từ khóa ‘**var’** để khai báo biến. Biến sẽ tồn tại ngay sau khi ta khai báo nó. Chúng ta có thể khởi tạo giá trị cho biến ngay khi khai báo:

var A = 10;

Chúng ta cũng có thể khai báo biến bằng cách gán giá trị cho nó mà không cần từ khóa var.

A = 10

Chúng ta có thể khai báo nhiều biến trên cùng một dòng bằng cách tách tên các biến bằng dấu phẩy.

Đây là một số ví dụ các biến:

var new\_amount

var answer = null

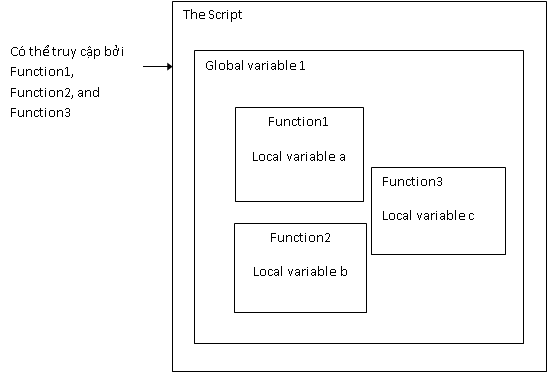
var old\_cost = 12.50

var result = ‘Unknown result’

result = true

## **Phạm vi của biến**

Phạm vi của biến được xác định tại vị trí mà nó được khai báo trong script. Nếu chúng ta khai báo một biến ngay phần đầu của script, thì nó được xem là một biến **global (toàn cục)**và có thể truy cập ở bất kỳ nơi đâu trong script. Nếu ta khai báo biến trong một hàm, nó được xem là biến **local (cục bộ)** và nó chỉ được sử dụng chỉ đối với hàm đó mà thôi. Những hàm khác trong script không thể truy xuất vào biến đó được. Hình dưới đây minh họa phạm vi của các biến trong script.

*Phạm vi của biến (variables)*

Các biến toàn cục không cần thiết phải sử dụng từ khóa var khi khai báo. Tuy nhiên, các biến cục bộ luôn luôn phải sử dụng từ khóa var khi khai báo

.

* Literals

Các literals là những giá trị cố định mà ta có thể dùng trong script. Giá trị của literal không bị thay đổi trong quá trình thực hiện script.

# **Các kiểu dữ liệu**

JavaScript có một tập hợp các kiểu dữ liệu như sau:

|  |  |
| --- | --- |
| **Kiểu dữ liệu** | **Mô tả** |
| Numbers | Các số nguyên hoặc số thực. Ví dụ: 487 hoặc 25.95 |
| Logical hoặc Boolean | True hoặc False |
| Strings | “Hello World” |
| Null | Keyword đặc biệt chỉ giá trị null. |

JavaScript luôn phân biệt các chữ hoa và chữ thường; null không giống như Null hoặc NULL

Dữ liệu luôn luôn thuộc một kiểu nào đó. Biến có kiểu dữ liệu phụ thuộc vào dữ liệu mà nó lưu trũ, kiểu dữ liệu của biến sẽ bị thay đổi khi giá trị của nó thay đổi.

Khi chúng ta khai báo một biến thì không cần phải chỉ ra kiểu dữ liệu của nó. Các kiểu dữ liệu sẽ tự động thay đổi nếu cần thiết trong quá trình thực thi script.

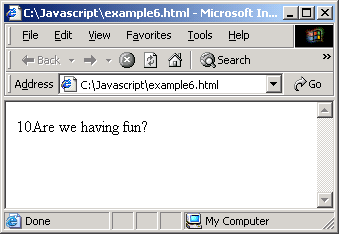
var x = 10

Sau đó, chúng ta có thể một chuỗi vào biến x trong script:

X = "Are you having fun?

<HTML>  
 <HEAD>   
 <SCRIPT LANGUAGE = "Javascript">  
 var x = 10;   
 document.write(x);  
 var x = "Are we having fun?";   
 document.write(x);  
 </SCRIPT>  
 </HEAD>  
</HTML>

**Kết quả:**

*Các biến JavaScript*

Trong JavaScript, ta có thể kết hợp hai biến thuộc các kiểu khác nhau. Chẳng hạn, nếu chúng ta kết hợp một chuỗi với một giá trị số với nhau thì JavaScript sẽ chuyển đổi các giá trị số sang các chuỗi.

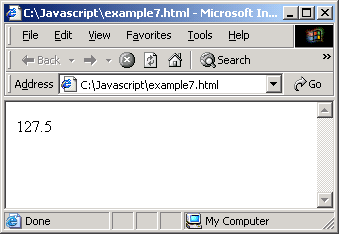
A = " This apple costs Rs." + 5

Nó sẽ cho ra kết quả là một chuỗi có giá trị "This apple costs Rs. 5"

Tuy nhiên, chúng ta không thể chuyển một chuỗi thành một số. Nếu chúng ta cộng một số thực 7.5 với một chuỗi là "12", kết quả sẽ là 127.5.

<HTML>  
 <HEAD>   
 <SCRIPT LANGUAGE = "Javascript">  
 var A = "12" + 7.5;  
 document.write(A);  
 </SCRIPT>  
 </HEAD>  
</HTML>

**Kết quả:**

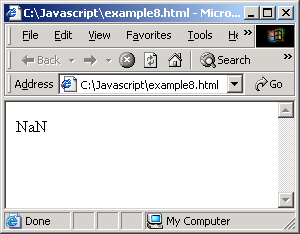
*JavaScript: variables (1)*

Trong JavaScript có hàm parseInt() và parseFloat() có chức năng chuyển đổi các chuỗi thành các số nguyên hoặc các số thực. Chẳng hạn, hàm parseInt("15") sẽ chuyển đổi chuỗi “15” sang giá trị là một số nguyên. Hàm parseFloat("35.45") sẽ chuyển một chuỗi thành một số thực là 34.45. Nếu hàm parseFloat() nhận ra một ký tự chứ không phải là một ký hiệu (+ hoặc -), một số (0 - 9), kí hiệu dấu thập phân hoặc một số mũ thì nó sẽ bỏ qua ký tự đó và tất cả các ký tự theo sau nó. Nếu ký tự đầu tiên không thể chuyển đổi được, hàm đó sẽ trả về giá trị “NaN” (Not a Number).

Trong ví dụ sau đây, chuỗi “a15” được chuyển vào hàm parseFloat (). Kết quả là NaN vì chuỗi đó không mở đầu bằng các ký tự có thể chuyển đổi được.

<HTML>  
 <HEAD>   
 <SCRIPT LANGUAGE = "Javascript">  
 var x = "a15"   
 a = parseFloat(x);  
 document.write(a);  
 </SCRIPT>  
 </HEAD>  
</HTML>

**Kết quả:**

*NaN*

Literals có thể có bất kỳ các kiểu dữ liệu nào sau đây :

1. **Số nguyên** – Chúng có thể được biểu diễn trong hệ thập phân, hệ thập lục phân và hệ nhị phân.
2. **Số thực (Floating-point) --** Các literal số thực (Floating-point literals) phải có ít nhât một chữ số. Số đó có thể có dấu thập phân hoặc “e” hoặc “E” theo sau một số nguyên. Số nguyên có thể dương (“+”) hoặc âm (“-”). Kí hiệu e hay E được hiểu là “lũy thừa cơ số 10”. Ví dụ: 10.24, 1.20e+22, 4E-8, .1815, v.v.
3. **Chuỗi** -- là chuỗi rỗng hoặc các ký tự được đặt trong dấu ngoặc đơn (‘Unexpected error’) hoặc dấu ngoặc kép “Hi! How are you”. Chuỗi đó phải bắt đầu và kết thúc bằng một dấu ngoặc của cùng một kiểu.

Sau đây là một số các ví dụ về literals chuỗi:

‘Unexpected error’

"Hi! How are you"

‘4531’

Khi dùng chuỗi, ta có thể chèn các ký tự đặc biệt vào chuỗi đó. Các ký tự đặc biệt sẽ thực hiện một công việc cụ thể. Ví dụ: chúng ta có thể thêm một dấu vạch chéo ngược (\) trong chuỗi đó. Dưới đây là bản liệt kê các ký tự đặc biệt và nghĩa của chúng:

|  |  |
| --- | --- |
| Ký tự | Ý nghĩa |
| \b | backspace |
| \f | form feed |
| \n | new line (xuống dòng và đưa con trỏ về đầu dòng đó) |
| \r | Carriage return (đưa con trỏ về đầu dòng hiện tại) |
| \t | Tab |

Những ví dụ dưới đây minh họa cách sử dụng các ký tự đặc biệt của JavaScript:

**Ký tự backspace**

<SCRIPT LANGUAGE = "Javascript">  
 alert(" This is line one \b This is line two");  
</SCRIPT>

**Ký tự form feed (sang trang)**

<SCRIPT LANGUAGE = "Javascript">  
 alert(" This is line one \f This is line two");  
  
</SCRIPT>

**Ký tự new line (xuống dòng)**

<SCRIPT LANGUAGE = "Javascript">  
   
alert(" This is line one \n This is line two");  
</SCRIPT>

**Ký tự carriage return (Phím xuống dòng)**

<SCRIPT LANGUAGE = "Javascript">  
 alert(" This is line one \r This is line two");  
 </SCRIPT>

**Ký tự tab**

<SCRIPT LANGUAGE = "Javascript">  
 alert(" This is line one \t This is line two");  
</SCRIPT>

Ngoài các ví dụ nêu trên, chúng ta có thể chèn các ký tự khác trong một chuỗi bằng cách đặt trước nó dấu backslash (\). Đây được xem là ký tự thoát (escaping character)

Dấu backslash (\) được dùng để bỏ qua ý nghĩa sử dụng của ký tự đứng sau nó, vì nhiều kí tự đặc biệt được thiết kế sẵn để phục vụ một chức năng cụ thể nào đó. Ví dụ, nếu ta muốn cho hiển thị ký tự backslash, đây là một kí tự đặc biệt nên ta phải loại bỏ ý nghĩa của nó.

1. **Boolean** - - Nó chỉ có thể nhận hai giá tri: True hoặc False. Kiểu dữ liệu này rất hữu dụng khi thực hiện các quyết định hoặc so sánh dữ liệu.
2. **null** - - Kiểu null chỉ có duy nhất một giá trị.: null. Null ngụ ý là không có dữ liệu. – Nó thực hiện chức năng là giữ chỗ trong một biến với ngụ ý là ở đó không có hữu dụng gì.

Số không hay một xâu rỗng và null là các giá trị khác nhau.

bài viết này dùng:

<h1>Các biến và các kiểu dữ liệu trong JavaScript</h1>

<h1>Các biến (Variables)</h1>

<h1 class="title" id="id1168843760357">Các kiểu dữ liệu</h1>

<h2 class="title" id="id1168842367840">Khai báo biến</h2>

<h2 class="title" id="id1168843721131">Phạm vi của biến</h2>

nhận xét: thẻ <h1> thì được dùng để chứa tiêu đề bài viết, tiêu đề luôn luôn chứa keyword và chỉ dùng 1 lần

thẻ<h2> thì được dùng nhiều lần và chứa các mục của bài viết.

đôi khi hai vấn đề trên vẫn không đúng với một số bài viết

bên cạnh đó có rất nhiều đường link liên quan đến keyword trên trang