tổng hợp các kiến thức javascript cơ bản

### [Javascript căn bản - freetuts](https://www.google.com.vn/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwiZxrTD3eTSAhXCKJQKHZ58DIQQFggXMAA&url=http%3A%2F%2Ffreetuts.net%2Fhoc-javascript%2Fjavascript-can-ban&usg=AFQjCNH8yF3g9s4ABUfLcj0wizZMZpjxWQ&sig2=c76147LGcFy_Stek6Eqlxw&bvm=bv.150120842,d.dGo)

freetuts.net › Javascript

14 thg 12, 2015 - Nhằm giúp **các bạn** có nguồn tài liệu học **Javascript** căn **bản** tiếng Việt miễn phí thì mình biên soạn **tổng** cộng hơn 50 bài học. ... Để học **Javascript** tốt thì đòi hỏi **bạn** phải biết CSS căn**bản** kết **hợp** với HTML để tạo nên giao ... trong **Javascript** · Bài 51: Bảng biểu **thức** Regular Expression trong **Javascript** ...

Bạn đã truy cập trang này vào ngày 27/02/2017.

### [Học Javascript: Tổng hợp một số ví dụ học javascript căn bản](https://hoangluyen.com/mot-so-vi-du-hoc-javascript-can-ban/)

https://hoangluyen.com › Học lập trình web

Xếp hạng: 2 - ‎21 phiếu bầu

27 thg 10, 2015 - Danh sách một số ví dụ học **Javascript** căn **bản** giúp **bạn** đọc nắm rõ **kiến thức** và ôn lại những **kiến thức** đã được học.

### [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....

**Kiến thức cơ bản** về HTML và CSS có thể giúp bạn tạo được website đơn giản. ... **Các** ứng dụng thường thấy ở **Javascript** có thể kể đến như: ... Trường **hợp** nếu bạn không muốn sử dụng Console để thực hành mà muốn viết lên ...... từ 0, nên biến cuối cùng trong mảng sẽ có số chỉ mục bằng **tổng**số lượng biến trừ đi 1.

Bạn đã truy cập trang này vào ngày 19/03/2017.

### [Ebook Tài liệu tự học HTML, CSS, Javascript, PHP&MySQL, ASP.NET ...](http://sinhvienit.net/forum/tai-lieu-tu-hoc-html-css-javascript-php-mysql-asp-net-tieng-viet-day.380487.html)

sinhvienit.net › ... › Tài liệu về lập trình › Xây dựng ứng dụng Web

27 thg 4, 2015 - NET tiếng việt đây!, **Tổng hợp** bộ tài liệu cực hay về web gồm: HTML: ... **Các** lệnh thường gặp trong css ... Những **kiến thức cơ bản** nhất về SQL

[**Tổng hợp** những điều **cơ bản** cần biết để học lập trình Java](http://sinhvienit.net/forum/tong-hop-nhung-dieu-co-ban-can-biet-de-hoc-lap-trinh-java.435934.html)

7 Tháng Ba 2016

[**tổng hợp** bài tập HTML, CSS và Javascrip](http://sinhvienit.net/forum/tong-hop-bai-tap-html-css-va-javascrip.361608.html)

1 Tháng Giêng 2015

[Các kết quả khác từ sinhvienit.net](https://www.google.com.vn/search?q=t%E1%BB%95ng+h%E1%BB%A3p+c%C3%A1c+ki%E1%BA%BFn+th%E1%BB%A9c+javascript+c%C6%A1+b%E1%BA%A3n+site:sinhvienit.net&biw=1366&bih=662&sa=X&ved=0ahUKEwiZxrTD3eTSAhXCKJQKHZ58DIQQrQIIOCgCMAM)

### [Học lập trình web cơ bản bằng HTML 5, CSS 3 và Javascript](https://techmaster.vn/khoa-hoc/25487/web-co-ban-html5-css3-va-javascript)

https://techmaster.vn › Khóa học

HTML, CSS và **Javascript** là bộ 3 ngôn ngữ cần thiết cho mọi web developer. ... Khóa học này sẽ cung cấp cho bạn những **kiến thức cơ bản**, cần thiết nhất để bắt ... CSS: Bố cục layout, **kiến thức** về responsive web design và cách tạo **các** hiệu ... **Tổng** quan về HTML (5 lesson) .... CHỌN GÓI KHÓA HỌC PHÙ **HỢP** VỚI BẠN.

### [Kiến thức tổng quan: JavaScript Cộng đồng phát triển game Việt](http://vietgamedev.net/blog/232/ki%E1%BA%BFn-th%E1%BB%A9c-t%E1%BB%95ng-quan-javascript/)

vietgamedev.net/blog/232/kiến-thức-tổng-quan-javascript/

27 thg 6, 2012 - **Kiến thức tổng** quan: **JavaScript**. Nguồn hình: ... Kiểu của **JavaScript** được chia ra làm hai loại: kiểu **cơ bản** và đối tượng. Đối tượng trong ...

### [Lập trình web cơ bản với Javascript jQuery - Tìm việc nhanh - Tuyển ...](http://smartjob.vn/lap-trinh-web-co-ban-voi-javascript-jquery/)

smartjob.vn/lap-trinh-web-co-ban-voi-javascript-jquery/

Giáo trình **Javascript** jQuery bản **tổng hợp** từ **cơ bản** đến nâng cao bằng tiếng ... **Các kiến thức** thực tế cần thiết cho việc xây dựng một trang web sẽ được liệt kê ...

### [Học jQuery từ cơ bản đến nâng cao - Unica](https://unica.vn/hoc-jquery-tu-co-ban-den-nang-cao)

https://unica.vn/hoc-jquery-tu-co-ban-den-nang-cao

Và một trong những công cụ cần thiết đến từ **các** thư viện **JavaScript** mở được ... Khóa học “jQuery**cơ bản** đến nâng cao” ra đời giúp **các** bạn có thể chỉnh sửa ... xử lý **tổng hợp** 04:59; Bài 23 Tổng kết**kiến thức** đã học trong project 3 04:02 ...

### [Tổng hợp hơn 250 bài học Javascript Tiếng Việt từ cơ bản đến nâng ...](https://www.dancongnghe.vn/threads/tong-hop-hon-250-bai-hoc-javascript-tieng-viet-tu-co-ban-den-nang-cao.695833/)

https://www.dancongnghe.vn › KIẾN THỨC › Lập trình › HTML / CSS / Javascript

... thức và công nghệ máy tính · Diễn đàn; **KIẾN THỨC** ... Mình đã **tổng hợp** được loạt bài học về**Javascript** từ **cơ bản** đến nâng cao. **Các** bạn có thể vào địa chỉ http://vietjack.com/**javascript**/index.jsp de học tập. Ngoài ra **các** bạn cũng có thể ...

### [Khóa học thiết kế và lập trình website PHP chuyên nghiệp](http://itplus-academy.edu.vn/Thiet-ke-va-lap-trinh-web-PHP-chuyen-nghiep.html)

itplus-academy.edu.vn/Thiet-ke-va-lap-trinh-web-PHP-chuyen-nghiep.html

Cung cấp **các** kỹ thuật lập trình hiệu quả, **kiến thức cơ bản** sử dụng ... **Các** bài labguide chi tiết từ dự án thực tế giúp học viên dễ dàng **tổng hợp** kiến ... Buổi 05: **Javascript cơ bản** và cấu trúc điều khiển - Ngôn ngữ **Javascript** và cách sử dụng.

### [Học Javascript cơ bản và nâng cao, Học lập trình Javascript miễn phí ...](http://mocnoi.com/hoidap-ct-1286975-hoc-javascript-co-ban-va-nang-cao-hoc-lap-trinh-javascript-mien-phi-hay-nhat.htm)

mocnoi.com › Thảo luận › Kiến thức văn học 12

22 thg 9, 2016 - **Kiến thức** văn học 12 ... **Tổng Hợp** Toàn Bộ Video Học Excel Từ **Cơ Bản** Đến Nâng Cao · Sự khác nhau giữa SGK vật lý 12 **cơ bản** và ... **Các** file **JavaScript** thường được lưu dưới đuôi js và được nhúng vào **các** trang HTML.

### [Lập Trình Archives - - BCDOnline Blog](http://bcdonline.net/category/lap-trinh/)

bcdonline.net/category/lap-trinh/

Chuyên mục **tổng hợp các** bài viết về lập trình Web, lập trình C, C++ **kiến thức** về ... thiêt kế web bằng HTML và CSS, ngôn ngữ lập trình Web PHP kết hợp **cơ** sở dữ ... Chào **các bạn** với việc trình bày một website thì không thể không nhắc đến ... Jquery là một thư viện được tạo ra từ **JavaScript**, Ajax hay nói cách khác đó là ...

### [Danh sách khoá học - CiOne - CiOne.Vn](https://www.cione.vn/class)

https://www.cione.vn/class

Khoá học này giúp bạn **tổng hợp kiến thức** của HTML & CSS để bố cục trang web ... cho Developer cung cấp **các kiến thức** sử dụng Photoshop **cơ bản** như cắt, ... Khoá học **JavaScript** & DOM cung cấp cho học viên **các kiến thức** để viết mã ...

### [Kiến thức Javascript - Kiến Thức Wordpress Từ Căn Bản Đến Nâng Cao](https://kienthucwp.com/kien-thuc-website/kien-thuc-javascript)

https://kienthucwp.com/kien-thuc-website/kien-thuc-javascript

Là nơi lưu trữ những **kiến thức javascript** từ **cơ bản** đến nâng cao. ... **Tổng hợp các** cách ẩn và cấm truy cập thanh admin bar của wordpress 142 lượt xem.

### [[Bài Tập-Hướng Dẫn Giải Ngôn Ngữ Lập Trình Javascript] Bài Tổng ...](https://www.youtube.com/watch?v=L7fs4BA1-Dg)

[**▶ 17:23**](https://www.youtube.com/watch?v=L7fs4BA1-Dg)

https://www.youtube.com/watch?v=L7fs4BA1-Dg

5 thg 12, 2016 - Tải lên bởi Nguyen Vi Khoa

[Bài Tập-Hướng Dẫn Giải Ngôn Ngữ Lập Trình **Javascript**] Bài **Tổng Hợp** HTML**Cơ Bản**. Nguyen Vi Khoa ...

### [Thủ thuật lập trình javascript bạn nên nhớ - Chia Sẽ Kiến Thức Lập Trình](http://codevivu.com/thu-thuat-lap-trinh/thu-thuat-lap-trinh-javascript-ban-nen-nho/)

codevivu.com › THỦ THUẬT LẬP TRÌNH

17 thg 12, 2016 - chào **các bạn** hôm nay codevivu.com sẽ chia sẽ cho **các bạn** những thủ ... kết **hợp**cùng với **javascript** để trong việc thiết kế hiệu ứng giao diện.

### [Tổng hợp các video PHP của Đỗ Mạnh Hà | Dịch vụ chia sẻ kiến thức ...](http://laptrinhx.com/topic/8401/tong-hop-cac-video-php-cua-do-manh-ha)

laptrinhx.com › Diễn đàn

22 thg 8, 2016 - Những video về PHP của Đỗ Mạnh Hà! **Các** bạn lập trình viên học thể nghiên cứu và ... LaptrinhX là dự án giáo dục chia sẻ **kiến thức** lập trình, phi chính phủ, phi lợi nhuận, ... PHP **cơ bản** bài 1.1: Giới thiệu ngôn ngữ PHP ... php251 performance249 **javascript**246 framework239 network239 interface224 ...

### [Download Lập trình web bằng HTML, DHTML và JAVASCRIPT (Tài ...](http://taimienphi.vn/download-lap-trinh-web-bang-html-dhtml-va-javascript-17684)

taimienphi.vn/download-lap-trinh-web-bang-html-dhtml-va-javascript-17684

Xếp hạng: 3 - ‎4 phiếu bầu

17 thg 3, 2015 - **Kiến thức cơ bản** về HTML, **JavaScript**, CSS và ASP - **Các** bài tập cơ ... **Tổng hợp các** kênh dạy lập trình căn bản bằng tiếng Việt. By Hoàng ...

### [kiến thức lập trình | Dạy thiết kế web](http://www.daythietkeweb.com/tag/kien-thuc-lap-trinh/)

www.daythietkeweb.com/tag/kien-thuc-lap-trinh/

Học **kiến thức** lập trình **JavaScript cơ** sở thông qua **các** hướng dẫn thực hành ... Quyển sách do **bạn**Mr.Nav trình bày và **tổng hợp** từ **các** bài viết trên izwebz.

### [Tổng hợp những series/bài viết hay nhất trên blog Tôi đi code dạo](https://kipalog.com/posts/Tong-hop-nhung-series-bai-viet-hay-nhat-tren-blog-Toi-di-code-dao)

https://kipalog.com/.../Tong-hop-nhung-series-bai-viet-hay-nhat-tren-blog-Toi-di-cod...

28 thg 5, 2016 - Series "C# hay ho": Bao gồm những bài viết lặt vặt, **tổng hợp** những điều thú vị ... Đọc xong series. **kiến thức** C# **cơ bản** của bạn sẽ vững hơn nhiều, ... thiệu những điểm sida khó nhằn của **JavaScript** mà bao senior developer ...

# **Javascript căn bản**

Đăng bởi: Administrator | Vào ngày: 14-12-2015 | View: 31675

Javascript đóng vai trò rất quan trọng khi bạn xây dựng website. Nhiệm vụ của Javascript là xây dựng các hiệu ứng động giúp website trở nên thân thiện với người dùng hơn.

Nhằm giúp các bạn có nguồn tài liệu học Javascript căn bản tiếng Việt miễn phí thì mình biên soạn tổng cộng hơn 50 bài học. Trong 50 bài này mình có tham khảo nhiều nguồn khác nhau và nhiều nhất là trang W3C, vì vậy có thể coi như đây là bộ tài liệu chuẩn cho các bạn. Bên cạnh mỗi bài viết mình sẽ bổ sung thêm các bài tập và các câu hỏi liên quan đến bài viết đó nên bạn sẽ có đầy đủ tài liệu từ bài tập Javascript cho đến tài liệu học javascript.

Để học Javascript tốt thì đòi hỏi bạn phải biết CSS căn bản kết hợp với HTML để tạo nên giao diện và kết hợp chúng với nhau. Hy vọng qua series tự học Javascript căn bản này sẽ giúp bạn có nguồn tài liệu tổng quan hơn.

Note: Series có tham khảo ở một số trang nước ngoài như W3C

### **Danh sách bài học**

* [Bài 01: Javascript là gì? Viêt ứng dụng javascript đầu tiên](http://freetuts.net/javascript-la-gi-viet-ung-dung-javascript-dau-tien-263.html)
* [Bài 02: Biến và khai báo biến trong javascript](http://freetuts.net/bien-va-khai-bao-bien-trong-javascript-265.html)
* [Bài 03: Hàm alert() - confirm() - prompt() trong javascript](http://freetuts.net/ham-alert-confirm-prompt-trong-javascript-264.html)
* [Bài 04: Các toán tử toán học và toán tử gán trong javascript](http://freetuts.net/cac-toan-tu-toan-hoc-va-toan-tu-gan-trong-javascript-266.html)
* [Bài 05: Lệnh kiểm tra điều kiện if else trong javascript](http://freetuts.net/lenh-kiem-tra-dieu-kien-if-else-trong-javascript-267.html)
* [Bài 06: Vòng lặp for trong javascript](http://freetuts.net/vong-lap-for-trong-javascript-270.html)
* [Bài 07: Vòng lặp while - do while trong javascript](http://freetuts.net/vong-lap-while-do-while-trong-javascript-271.html)
* [Bài 08: Hàm và tạo hàm (function) trong javascript](http://freetuts.net/ham-va-tao-ham-function-trong-javascript-274.html)
* [Bài 09: Một số ví dụ học javascript căn bản (Phần 1)](http://freetuts.net/mot-so-vi-du-hoc-javascript-can-ban-phan-1-278.html)
* [Bài 10: Thao tác với mảng trong javascript](http://freetuts.net/thao-tac-voi-mang-trong-javascript-288.html)
* [Bài 11: Các hàm xử lý mảng trong javascript](http://freetuts.net/cac-ham-xu-ly-mang-trong-javascript-289.html)
* [Bài 12: DOM là gì? Các loại DOM trong Javascript](http://freetuts.net/dom-la-gi-cac-loai-dom-trong-javascript-366.html)
* [Bài 13: DOM Element trong javascript](http://freetuts.net/dom-element-trong-javascript-367.html)
* [Bài 14: Sự kiện (Event) trong Javascript](http://freetuts.net/su-kien-event-trong-javascript-368.html)
* [Bài 15: Return TRUE/FALSE của Events trong Javascript](http://freetuts.net/return-true-false-cua-events-trong-javascript-369.html)
* [Bài 16: DOM HTML trong Javascript](http://freetuts.net/dom-html-trong-javascript-370.html)
* [Bài 17: DOM CSS trong Javascript](http://freetuts.net/dom-css-trong-javascript-371.html)
* [Bài 18: Thêm sự kiện (Event) bằng Javascript](http://freetuts.net/them-su-kien-event-bang-javascript-373.html)
* [Bài 19: Hàm addEventListener() trong Javascript](http://freetuts.net/ham-addeventlistener-trong-javascript-374.html)
* [Bài 20: Đối tượng this trong Javascript](http://freetuts.net/doi-tuong-this-trong-javascript-376.html)
* [Bài 21: Bài tập javascript - xây dựng menu dropdow](http://freetuts.net/bai-tap-javascript-xay-dung-menu-dropdow-372.html)
* [Bài 22: Hàm removeEventListener() trong Javascript](http://freetuts.net/ham-removeeventlistener-trong-javascript-375.html)
* [Bài 23: Sự kiện onload trong Javascript](http://freetuts.net/su-kien-onload-trong-javascript-378.html)
* [Bài 24: Biến toàn cục và biến cục bộ trong Javascript](http://freetuts.net/bien-toan-cuc-va-bien-cuc-bo-trong-javascript-377.html)
* [Bài 25: Sử dụng Firebug để kiểm tra lỗi Javascript](http://freetuts.net/su-dung-firebug-de-kiem-tra-loi-javascript-380.html)
* [Bài 26: Lệnh console.log kết hợp với Firebug trong Javascript](http://freetuts.net/lenh-consolelog-ket-hop-voi-firebug-trong-javascript-381.html)
* [Bài 27: Lệnh break - continue trong Javascript](http://freetuts.net/lenh-break-continue-trong-javascript-382.html)
* [Bài 28: DOM Nodes trong Javascript](http://freetuts.net/dom-nodes-trong-javascript-383.html)
* [Bài 29: BOM là gì? BOM trong javascript](http://freetuts.net/bom-la-gi-bom-trong-javascript-384.html)
* [Bài 30: BOM - Window trong Javascript](http://freetuts.net/bom-window-trong-javascript-385.html)
* [Bài 31: BOM - Location điều hướng và xử lý URL trong Javascript](http://freetuts.net/bom-location-dieu-huong-va-xu-ly-url-trong-javascript-386.html)
* [Bài 32: BOM - History trong Javascript](http://freetuts.net/bom-history-trong-javascript-387.html)
* [Bài 33: BOM - Cookie trong Javascript](http://freetuts.net/bom-cookie-trong-javascript-388.html)
* [Bài 34: BOM - Window Navigator trong Javascript](http://freetuts.net/bom-window-navigator-trong-javascript-389.html)
* [Bài 35: BOM - Screen trong Javascript](http://freetuts.net/bom-screen-trong-javascript-390.html)
* [Bài 36: setTimeout() và setInterval() trong Javascript](http://freetuts.net/settimeout-va-setinterval-trong-javascript-391.html)
* [Bài 37: Chuỗi (string) trong Javascript](http://freetuts.net/chuoi-string-trong-javascript-392.html)
* [Bài 38: Các hàm xử lý chuỗi trong Javascript](http://freetuts.net/cac-ham-xu-ly-chuoi-trong-javascript-393.html)
* [Bài 39: Hàm typeof trong Javascript](http://freetuts.net/ham-typeof-trong-javascript-398.html)
* [Bài 40: Đối tượng Number trong Javascript](http://freetuts.net/doi-tuong-number-trong-javascript-397.html)
* [Bài 41: Hàm xử lý Number trong Javascript](http://freetuts.net/ham-xu-ly-number-trong-javascript-399.html)
* [Bài 42: Đối tượng Date trong Javascript](http://freetuts.net/doi-tuong-date-trong-javascript-401.html)
* [Bài 43: Hàm xử lý ngày tháng (Date) trong Javascript](http://freetuts.net/ham-xu-ly-ngay-thang-date-trong-javascript-400.html)
* [Bài 44: Lệnh switch case trong Javascript](http://freetuts.net/lenh-switch-case-trong-javascript-402.html)
* [Bài 45: Tìm hiểu JavaScript Hoisting](http://freetuts.net/tim-hieu-javascript-hoisting-403.html)
* [Bài 46: Use Strict là gì? Strict Mode trong javascript](http://freetuts.net/use-strict-la-gi-strict-mode-trong-javascript-407.html)
* [Bài 47: Đối tượng (object) trong Javascript](http://freetuts.net/doi-tuong-object-trong-javascript-408.html)
* [Bài 48: Thao tác với đối tượng (Object) trong Javascript](http://freetuts.net/thao-tac-voi-doi-tuong-object-trong-javascript-409.html)
* [Bài 49: Javascript Object Prototypes](http://freetuts.net/javascript-object-prototypes-410.html)
* [Bài 50: RegExp - Regular Expression trong Javascript](http://freetuts.net/regexp-regular-expression-trong-javascript-405.html)
* [Bài 51: Bảng biểu thức Regular Expression trong Javascript](http://freetuts.net/bang-bieu-thuc-regular-expression-trong-javascript-418.html)
* [Bài 52: HTML5 localStorage và sessionStorage](http://freetuts.net/html5-localstorage-va-sessionstorage-760.html)

Nhận xét:

<h1>Javascript căn bản</h1>  
<h3>Danh sách bài học</h3>

# **Tổng hợp một số ví dụ học Javascript căn bản**

Bởi Hoàng Luyến gửi  [Bình luận](https://hoangluyen.com/mot-so-vi-du-hoc-javascript-can-ban/#facebook-comment)

Bài viết này nằm trong bộ Series [Học Javascipt cơ bản](https://hoangluyen.com/series/hoc-javascipt-co-ban/).Xem tất cả

## **Danh sách một số ví dụ học Javascript căn bản giúp bạn đọc nắm rõ kiến thức và ôn lại những kiến thức đã được học**

Chúng ta đã trải qua 8 bài [học Javascript](https://hoangluyen.com/hoc-javascript/) căn bản rồi. Tiếp tục bài số 9 này mình chủ yếu ôn lại kiến thức bài học trước, đưa ra một số ví dụ khi học Javascript để bạn nắm vững hơn những gì đã được học.

### **Nội dung bài học**

1. Bài tập tìm min – max với javascript

2. Bài tập giải phương trình bậc nhất với javascript

3. Bài tập kiểm tra số chính phương với javascript

### **Lời kết**

Hi vọng với 3 phần ví dụ bài tập Javascript trên có thể giúp bạn ôn lại 8 bài học trước và nắm vững về nó. Giờ thì hãy cùng mình bước tới bài 10 nhé!

Nhận xét:

<h1>Tổng hợp một số ví dụ học Javascript căn bản</h1>  
<h2>Danh sách một số ví dụ học Javascript căn bản giúp bạn đọc nắm rõ kiến thức và ôn lại những kiến thức đã được học</h2>  
<h3>Nội dung bài học</h3>  
<h3>Lời kết</h3>

# **Javascript cơ bản – Hành trình của một Anh Hùng**

[Kiến thức cơ bản về HTML và CSS](https://thachpham.com/web-development/html-css/html-va-css-can-ban-danh-cho-cho-moi-nguoi.html) có thể giúp bạn tạo được website đơn giản. Nhưng nếu bạn mong muốn 1 website sinh động và phức tạp hơn, bạn cần Javascript. Javascript là ngôn ngữ lập trình đơn giản, nhưng cực kì mạnh mẽ và phổ biến cho lập trình web. Các ứng dụng thường thấy ở Javascript có thể kể đến như:

* Tương tác với HTML và thay đổi nội dung và định dạng trên website dễ dàng.
* Tương tác với các hành động của người dùng như nhấn chuột, gõ phím…
* Xử lý và kiểm tra các dữ liệu trên form trước khi gửi về server.
* Tạo và truy xuất thông tin lưu trong cookie trên máy người dùng.
* Đóng vai trò như 1 ngôn ngữ lập trình phía server (sử dụng các framework như Node.js).

Có nhiều phương pháp để học Javascript, và tốt nhất là để người học được tự tay mày mò trong suốt quá trình tìm hiểu. Bài viết hôm nay sẽ áp dụng phương pháp đó, và tiếp cận nó theo một cách mới để mong bạn đọc dễ làm quen và hình dung hơn: gamification – trò chơi hóa nội dung bài học.

Hãy tưởng tượng bạn là nhân vật chính trong một game nhập vai, khởi đầu từ con số 0 tròn trĩnh để đấu tranh trở thành Anh Hùng trong cõi Javascript. Không gì hứa hẹn một hành trình bằng phẳng cả, nhưng đừng ngại ngần khi định mệnh đã gọi tên!

INTRO CHAPTER!

Hãy tạo nên huyền thoại của riêng bạn về 1 Anh Hùng Javascript!



Javascript có thể được sử dụng dễ dàng với thẻ HTML script: chỉ cần đưa các câu lệnh Javascript vào trong cặp thẻ hoặc nhúng 1 file Javascript bên ngoài.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07 | <html>  <body>  <script type="text/javascript">  // Gõ code ở đây  </script>  </body>  </html> |

|  |  |
| --- | --- |
| 01  02  03  04  05 | <html>  <body>  <script type="text/javascript" src="đường-dẫn-đến-file-javascript.js"></script>  </body>  </html> |

Tuy nhiên, trong hành trình Javascript cơ bản này, bạn không cần phải chèn code hay file Javascript vào 1 file html và chạy file này. Bạn sẽ gõ code trực tiếp trên trình duyệt bằng công cụ Console. Để mở Console, hãy bấm F12 và chọn tab Console ở khung công cụ lập trình được hiển thị, hoặc sử dụng phím tắt nhanh Ctrl+Shift+J (Chrome/Firefox).



Trường hợp nếu bạn không muốn sử dụng Console để thực hành mà muốn viết lên web thì hãy sử dụng các phương thức xuất dữ liệu ra trang tên là document.write(). Ví dụ:

|  |  |
| --- | --- |
| 01  02  03 | document.write("Tôi tên là Phúc!"); // Hiển thị chữ Tôi tên là Phúc ở website  var name = "Phúc";  document.write(name); // Hiển thị chữ Phúc ở website. |

Xong chưa nào? Hãy sẵn sàng chinh phục những thử thách để viết nên câu chuyện về Anh Hùng Javascript của riêng bạn!

WELCOME TO CHAPTER 1!

Chúc mừng bạn đã bắt đầu hành trình Anh Hùng Javascript!

### **Quest 1: Kiến tạo Anh Hùng – Biến và kiểu dữ liệu**

Việc lưu trữ dữ liệu là một trong những điều quan trọng nhất khi lập trình. Thông thường, các dữ liệu tạm sẽ được lưu giữ bằng các biến (variable) trong bộ nhớ. Đối với Javascript, bạn cần khai báo biến bằng từ khóa var trước khi gán dữ liệu cho biến đó.

Mỗi Anh Hùng đều cần có 1 tên gọi để lưu danh sử sách, và tiện khoe với các em gái xinh đẹp trong quán rượu :”>. Hãy gõ lại đoạn code bên dưới vào Console và nhập tên vào hộp thoại được xổ ra.

NEW SKILL!

Hàm prompt(): dùng để nhận nhập liệu từ người dùng và trả về dạng chuỗi kí tự.

|  |  |
| --- | --- |
| 01  02 | var ten = prompt('Xin chào! Hãy gọi tên Anh Hùng của bạn:');  ten; |

Hàm prompt() là hàm viết sẵn của hệ thống để nhận nhập liệu từ người dùng (ta sẽ tìm hiểu thêm về hàm ở phần sau). Kết quả của hàm sẽ trả về bất cứ gì người dùng nhập vào ở hộp thoại. Nếu bạn muốn sử dụng kết quả đó, bạn cần phải lưu trữ nó bằng 1 biến. Ở dòng thứ 1, ta đã khai báo biến ten và gán cho nó kết quả trả về của hàm prompt(). Sau này, bất cứ khi nào bạn cần, bạn chỉ cần gọi biến cần thiết để lấy dữ liệu mà biến đó giữ. Đó là lý do khi bạn gọi lên biến ten ở dòng 2, bạn sẽ thấy lại giá trị mà bạn vừa nhập. Sướng chưa 😀

Demo: http://jsfiddle.net/6QhmR/1/

#### **Biến**

Đoạn code trên là 1 ví dụ khi sử dụng biến. Hãy nghĩ 1 biến như 1 ngăn tủ được đặt tên, và tên biến là tên ngăn tủ, giá trị của biến là vật dụng trong ngăn tủ. Bạn có thể chứa bất kì thứ gì bạn muốn trong ngăn tủ đó, và khi cần tìm lại, bạn chỉ cần tra đúng tên ngăn tủ mà bạn cần.

Biến có 2 phần, là tên biến và giá trị của biến. Tên biến rất đơn giản, khi bạn có thể đặt tên biến tùy ý với các chữ cái hoa hay thường, các con số và dấu gạch chân (\_). Còn với giá trị của biến thì chúng ta cần để ý thêm về kiểu dữ liệu. Ở ví dụ trên, ta đã dùng biến ten để chứa 1 hàm prompt và chứa dữ liệu kiểu string. Javascript hỗ trợ nhiều kiểu dữ liệu, nhưng ở mức cơ bản bạn sẽ cần nắm vững các kiểu dữ liệu sau:

* String dùng để chứa chuỗi kí tự và phải được bao quanh bởi cặp nháy đơn (‘…’) hay nháy đôi (“…”).
* Number dùng để chứa dữ liệu kiểu số nguyên, số thập phân và không nằm trong cặp nháy đơn hay nháy đôi.
* Boolean là kiểu logic, chỉ có 2 giá trị là đúng (true) hoặc sai (false).
* Object là một đối tượng nói chung với các thuộc tính và phương thức riêng. Ta sẽ tìm hiểu về Object ở phần 7.
* Array là mảng, dùng để chứa tập hợp nhiều biến. Phần 8 sẽ giúp bạn hiểu rõ hơn.

#### **Thao tác với biến**

Để chứa đồ trong tủ, bạn cần phải tìm 1 ngăn tủ rỗng, dán tên cho ngăn tủ đó (khai báo biến) và đặt 1 vật gì đó vào bên trong (gán giá trị cho biến).

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29 | // Khai báo với từ khóa var  var level;    // Sau khi khai báo, dùng dấu = để gán giá trị  // Giá trị kiểu số (Number)  level = 1;    // Khai báo và gán 1 giá trị ngay lúc đó gọi là khởi tạo  // Giá trị kiểu boolean  var male = true;    // Giá trị kiểu chuỗi kí tự (String)  var phancap = "Anh Hùng";    /\* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \* THỬ THÁCH LEVEL 1  \* -----------------  \* Khởi tạo thêm 3 biến để chứa thông tin về cấp độ (level),  \* điểm kinh nghiệm (xp) và điểm kĩ năng (skillPoints).  \* Giá trị khởi tạo của từng biến như sau:  \* - Cấp độ: 1  \* - Điểm kinh nghiệm: 0  \* - Điểm kĩ năng: 5  \* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \*/ |

Sau khi đặt xong biến, bạn có thể thử in ra bằng cách sử dụng hàm alert() để xem giá trị. Ví dụ:

|  |  |
| --- | --- |
| 01  02  03 | var name = "Thachpham.com";    alert(name); |

[+ expand source](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#)

LEVEL UP! Chúc mừng bạn lên cấp 2! XP +150

### **Quest 2: Sức mạnh tính toán – Các toán tử cơ bản**

Javascript hỗ trợ đầy đủ các toán tử cơ bản cộng trừ nhân chia. Ngoài ra, bạn sẽ làm quen với 2 toán tử mới là ++ (tăng giá trị của biến kiểu Number lên 1.0 đơn vị) và — (giảm giá trị của biến kiểu Number xuống 1.0 đơn vị). Thứ tự tính toán (trong trường hợp không sử dụng dấu ngoặc tròn để gom nhóm ưu tiên) là nhân chia trước, cộng trừ sau và từ trái sang phải.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07 | var level = 1;  var xp = 0;  var skillPoints = 5;    level = level + 1; // hoặc tương đương là level++;  xp = 150 \* (level - 1) + 200; // sử dụng dấu ngoặc tròn để gom nhóm ưu tiên  skillPoints = skillPoints + xp / 10; |

Ngoài ra, Javascript sử dụng toán tử + để nối các chuỗi kí tự.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26 | // Nối 3 chuỗi đơn giản  "Javascript" + " " + "căn bản";    // Nối 1 chuỗi và 1 biến String  ten + " sẽ là Anh Hùng Javascript";    // Nối 1 chuỗi với 1 biến Number  "Cấp độ hiện tại của bạn là " + level;    /\* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \* THỬ THÁCH LEVEL 2  \* -----------------  \* Khởi tạo thêm 3 biến là strength, agility và intel.  \* Sử dụng giá trị của điểm kĩ năng (skillPoints) phía trên  \* và tính toán giá trị khởi tạo của các biến:  \* - strength bằng 60% điểm kĩ năng cộng với 10% của điểm kinh nghiệm.  \* - agility bằng 20% điểm kĩ năng cộng với 20% của điểm kinh nghiệm.  \* - intel bằng 20% điểm kĩ năng cộng với 1000% của cấp độ.  \* Sau khi tính toán xong, hãy xuất các chỉ số theo cú pháp:  \* "Chỉ số strength của bạn là " + chỉ số  \* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \*/ |

[+ expand source](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#)

LEVEL UP! Chúc mừng bạn lên cấp 3! XP +200

### **Quest 3: Sức mạnh logic – Các toán tử so sánh**

Trên con đường hành hiệp, Anh Hùng sẽ gặp rất nhiều cám dỗ và lựa lọc. Chính vì thế, Anh Hùng cần phải so sánh chính xác và sử dụng sức mạnh logic để nhận ra chân tướng đúng/sai của mọi sự vật, sự việc.

Trong Javascript, kết quả của mọi so sánh đều là 1 biến kiểu boolean: hoặc là true, hoặc là false. Để so sánh giữa các biến hay các biểu thức, bạn có thể sử dụng === cho so sánh bằng, !== cho so sánh không bằng, < và > cho so sánh hơn kém.

NEW SKILL!

Hàm console.log(): có thể được dùng để in giá trị của một biến.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25 | // So sánh 2 biến String  var test1 = ("Javascript" === "JAVAscript");  var test2 = ("Javascript" === "Javascript");  var test3 = ("Javascript" !== "javascript");  console.log(test1);  console.log(test2);  console.log(test3);    // So sánh 2 biến Number  test1 = (1 > 2);  test2 = (0 < -5);  test3 = (1.5 === 1.4999);  console.log(test1);  console.log(test2);  console.log(test3);    // So sánh 2 biểu thức Number  test1 = (1 - 2 + 3 - 4) <= 5;  test2 = (1 - 2 + 3 - 4)/5 >= 1;  console.log(test1);  console.log(test2);    // So sánh 1 biến Number và 1 biến String  test3 = 15 === "15";  console.log(test3); |

LEVEL UP! Chúc mừng bạn lên cấp 4! XP +300

### **Quest 4: Lựa chọn định mệnh – Điều kiện và rẽ nhánh**

Chúc mừng bạn đã nắm vững các kĩ năng cơ bản, và đã đến lúc để lựa chọn phân cấp Anh Hùng. Bạn sẽ được lựa chọn giữa 3 phân cấp với 3 đặc điểm về chỉ số khác nhau.

Các lựa chọn trong Javascript sẽ dẫn đến các điều kiện rẽ nhánh khác nhau. Để tạo ra các lựa chọn trong Javascript, bạn sẽ sử dụng từ khóa if hoặc if...else. Các lựa chọn có thể lồng ghép với nhau để tạo ra các luồng rẽ nhánh phức tạp. Tuy nhiên, cần cẩn trọng vì quá nhiều luồng rẽ nhánh sẽ khiến đoạn code khó hiểu và khó bảo trì về sau.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62 | /\* Câu lệnh if thông thường  \* ------------------------  \* Biểu thức  \* Khi biểu thức logic trong ngoặc  \*/  if (level < 4) {  console.log('Bạn chưa đạt cấp độ tối thiểu để tham gia thử thách.');  }    // Câu lệnh if...else  if (level >= 4) {  console.log('Bạn đã đủ cấp độ để tham gia thử thách level 4.');  } else {  console.log('Bạn cần đạt ít nhất level 4 để tham gia thử thách.');  }    // Câu lệnh if...else if...else  if (level > 4) {  console.log('Bạn dư sức vượt qua thử thách level 4.');  } else if (level === 4) {  console.log('Bạn đủ sức vượt qua thử thách level 4.');  } else {  console.log('Về uống sữa thêm cho mau lớn nha cưng!');  }    // Ở phần 1, bạn đã khai báo biến <code>ten</code>  // và sử dụng để giữ giá trị về tên Anh Hùng của bạn  // Ta sẽ kiểm tra xem bạn có thể gọi lại chính xác không  // để thưởng hoặc phạt theo kết quả  var kiemTraTen = prompt('Tên Anh Hùng của bạn là gì?');  if (kiemTraTen === ten) {  console.log('Chính xác! Bạn có trí nhớ rất tốt! intel +5!');  intel = intel + 5;  } else {  console.log('Tệ thật! Tên chính mình mà không nhớ sao?! intel -5!');  intel = intel - 5;  }    /\* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \* THỬ THÁCH LEVEL 4  \* -----------------  \* Sử dụng hàm prompt('Bạn muốn chọn phân cấp nào? a: Chiến  \* Binh | b: Sát Thủ | c: Thợ Săn') để yêu cầu người  \* dùng nhập vào 1 kí tự hoặc là a, hoặc b, hoặc c. Giá trị  \* nhập vào sẽ được xử lý như sau:  \* 1. Nếu là kí tự a:  \* - Cập nhật phân cấp Anh Hùng (biến phancap): Chiến Binh.  \* - Điều chỉnh chỉ số: strength +10, agility -10  \* 2. Nếu là kí tự b:  \* - Cập nhật phân cấp Anh Hùng (biến phancap): Sát Thủ.  \* - Điều chỉnh chỉ số: strength -10, agility +5, intel +5  \* 3. Nếu là kí tự c:  \* - Cập nhật phân cấp Anh Hùng (biến phancap): Thợ Săn.  \* - Điều chỉnh chỉ số: strength -10, intel +10  \* 4. Nếu không phải 3 trường hợp trên  \* - Không cập nhật và điều chỉnh gì cả.  \* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \*/ |

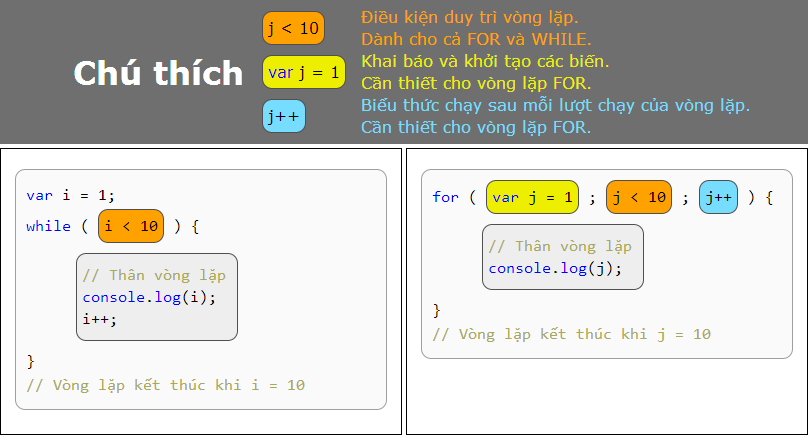
[+ expand source](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#)

LEVEL UP! Chúc mừng bạn lên cấp 5! XP +400

### **Quest 5: Đấu trường Sinh Tử – Vòng lặp**

Anh Hùng nào cũng phải trải qua quá trình tập luyện và đấu tranh gian khổ để đạt đến vinh quang. Quá trình đó đòi hỏi những nỗ lực lặp đi lặp lại đến khi đạt được một ý đồ, một mục đích nào đó.

Việc lặp các thao tác trong lập trình được gọi là vòng lặp, và là một trong những thành phần quan trọng nhất của lập trình. Vòng lặp sẽ có 1 điều kiện để duy trì và khi điều kiện đó không được đảm bảo thì vòng lặp sẽ kết thúc. Đối với Javascript cơ bản, có 2 cách để tạo vòng lặp: vòng lặp for và vòng lặp while. Cả 2 đều bao gồm 2 thành phần: điều kiện để duy trì vòng lặp, và thân vòng lặp chứa các thao tác sẽ được lặp.



|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13 | // Ví dụ vòng lặp while  var i = 1;  while (i < 10) {  console.log(i);  i++;  }  console.log(i);    // Ví dụ vòng lặp for  for (var j = 1; j < 10;j++) {  console.log(j);  }  console.log(j); |

BATTLE! Bạn chạm trán 1 con rồng già, và buộc phải giết nó để vượt qua

NEW SKILL!

Hàm Math.random(): dùng để tạo giá trị ngẫu nhiên từ 0 đến 1. Để tạo xác suất ngẫu nhiên theo %, có thể sử dụng (Math.random() \* 100).

NEW SKILL!

Hàm alert(): dùng để xuất một chuỗi kí tự qua hộp thoại thông báo. Có thể dùng để debug hay xuất thông tin cho người dùng.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53 | /\* ========================================================  \* Trước khi vào trận chiến, cần xem lại các chỉ số của bạn  \* và xem xét các thông tin của đối thủ.  \* ========================================================\*/  level = 5; // Đảm bảo bạn sẽ ở cấp độ 5  var health=level\*100;// Khi lượng máu <= 0 thì bạn sẽ chết  console.log(strength); // Quyết định mức sát thương  console.log(agility); // Quyết định xác suất né đòn  console.log(intel); // Quyết định xác suất chí mạng    /\* ========================================================  \* Rồng Già  \* --------  \* Chậm chạp và yếu ớt hơn so với đồng loại, nhưng vẫn rất  \* nguy hiểm với khả năng phun lửa có thể gây sát thương chí  \* mạng rất cao  \* ========================================================\*/  var dragonHealth = 600;  var dragonStr = 100;  var dragonAgi = 15;  var dragonInt = 80;    /\* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \* THỬ THÁCH LEVEL 5  \* -----------------  \* Viết trò chơi Diệt Rồng theo các bước sau:  \* 1. Mỗi lượt chơi, Anh Hùng và Rồng lần lượt tấn công nhau.  \* => GỢI Ý: Sử dụng vòng lặp while.  \* 2. Anh Hùng tấn công trước, Rồng tấn công sau.  \* 3. Trò chơi kết thúc khi một phe hết máu (health < 0 hoặc dragonHealth < 0).  \* => GỢI Ý: đặt 1 biến boolean để kiểm tra.  \* 4. Sử dụng hàm (Math.random() \* 100) để lấy xác suất ngẫu nhiên theo %.  \* 5. Rồng né đòn thành công khi % xác suất ngẫu nhiên bé hơn dragonAgi.  \* => GỢI Ý: so sánh dragonAgi với (Math.random() \* 100).  \* 6. Anh Hùng né đòn thành công khi % xác suất ngẫu nhiên bé hơn agility.  \* => GỢI Ý: so sánh agility với (Math.random() \* 100) (khác ở 5).  \* 7. Sát thương do Anh Hùng gây ra bằng với chỉ số strength.  \* => GỢI Ý: khi Anh Hùng đánh trúng, giảm máu Rồng theo giá trị sát thương.  \* 8. Sát thương do Rồng gây ra bằng với chỉ số dragonStr.  \* => GỢI Ý: khi Rồng đánh trúng, giảm máu Anh Hùng theo giá trị sát thương.  \* 9. Anh Hùng gây sát thương chí mạng khi % xác suất ngẫu nhiên lớn hơn intel.  \* => GỢI Ý: so sánh intel với 1 giá trị (Math.random() \* 100) (khác ở 5 và 6).  \* 10. Rồng gây sát thương chí mạng khi % xác suất ngẫu nhiên lớn hơn dragonInt.  \* => GỢI Ý: so sánh dragonInt với 1 giá trị (Math.random() \* 100) (khác ở 5,6,9)).  \* 11. Sát thương chí mạng sẽ gây thêm 200% sát thương.  \* => GỢI Ý: nếu gây sát thương chí mạng, giảm máu đi 2 lần sát thương.  \*  \* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \*/ |

[+ expand source](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#)

BATTLE WON! Chúc mừng bạn đã đạt danh hiệu “Dũng sĩ diệt Rồng”!

LEVEL UP! Chúc mừng bạn lên cấp 6! XP +600

### **Quest 6: Tạo và gọi Hàm**

Hàm là 1 đoạn code riêng biệt có thể sử dụng nhiều lần. Hàm có thể nhận tham số từ biến và trả kết quả về để gán cho biến.

Trước khi sử dụng, hàm cần được khởi tạo với từ khóa function tênHàm(thamSố1, thamSố2). Mọi biến mà hàm cần sử dụng từ bên ngoài hàm được truyền vào giữa hai ngoặc, và được gọi là tham số. Để bắt hàm trả về 1 giá trị nào đó, bạn cần sử dụng từ khóa return tênBiếnMuốnTrảVề;. return sẽ kết thúc hàm ngay lập tức, nên nếu có câu lệnh nào phía sau, câu lệnh đó sẽ không được thực thi.

Một lưu ý khi sử dụng hàm là đừng ôm đồm quá nhiều xử lý trong một hàm. 1 hàm xử lý chuỗi không cần phải biết đầy đủ cắt chuỗi, đảo chuỗi, nhân đôi chuỗi.., mà chỉ cần thực hiện tốt 1 chức năng thôi. Đây là một thói quen lập trình tốt, sẽ giúp code đơn giản, dễ hiểu và dễ bảo trì qua thời gian.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | // Lấy ví dụ trò chơi diệt Rồng ở phần 5  // Tạo hàm gây sát thương để giảm máu  // Tham số truyền vào: biến Number satThuong  // Kết quả trả về: biến Number luongMauHienTai  function gaySatThuong(luongMauHienTai, satThuong) {  luongMauHienTai = luongMauHienTai - satThuong;  return luongMauHienTai;  }    // Để sử dụng hàm, bạn cần phải gọi hàm  var luongMau = 100;  var satThuong = 49;  var luongMauConLai = gaySatThuong(luongMau, satThuong);  console.log(luongMauConLai);    /\* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \* THỬ THÁCH LEVEL 6  \* -----------------  \* Viết lại trò chơi Diệt Rồng, có sử dụng hàm để:  \* 1. Tính toán khả năng né đòn.  \* 2. Tính toán lượng máu còn lại sau khi bị sát thương.  \*  \* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \*/ |

LEVEL UP! Chúc mừng bạn lên cấp 7! XP +700

### **Quest 7: Đối tượng**

Đối tượng Javascript cũng giống như một vật thể thực ngoài đời: có các tính chất (thuộc tính) và khả năng (phương thức/hàm). Hiểu một cách khác, đối tượng trong Javascript là tập hợp của các thuộc tính và các phương thức (về bản chất cũng là hàm) bên trong.

Có thể xem đối tượng như 1 biến đặc biệt trong Javascript, được khai báo trong cặp ngoặc nhọn {...} và sử dụng dấu . để truy xuất các thuộc tính và phương thức bên trong.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54 | // Cú pháp khai báo 1 đối tượng  var RongGia = {  // Khai báo các thuộc tính  name: 'Rồng Già',  phanLop: 'Rồng',  age: 6969,  health: 600,  strength: 100,  agility: 15,  intel: 80,    // Khai báo các phương thức  bay: function() {  console.log('Flappy Dragon!');  },  phunLửa: function() {  console.log('Rồng phun lửa');  },  ngủ: function() {  console.log('Zzz... Zzz...');  }  };    // Sử dụng các thuộc tính như biến thông thường  console.log(RongGia.name);  console.log(RongGia.phanLop);  console.log(RongGia.age);    // Sử dụng phương thức như gọi hàm bình thường  RongGia.bay();  RongGia.phunLửa();  RongGia.ngủ();    // Bạn cũng có thể thêm các thuộc tính và phương thức  // sau khi đã khởi tạo đối tượng  RongGia.level = 10;  RongGia.chết = function() {  console.log('Rồng lên bàn thờ!');  };    /\* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \* THỬ THÁCH LEVEL 7  \* -----------------  \* Khai báo một đối tượng AnhHung để chứa các thông tin về  \* Anh Hùng của bạn:  \* 1. Các thuộc tính: tên, phân lớp, cấp độ, các chỉ số..  \* 2. Các phương thức ví dụ như đi, chạy, nhảy, ngủ..  \*  \* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \*/ |

LEVEL UP! Chúc mừng bạn lên cấp 8! XP +800

### **Quest 8: Mảng**

Mảng là tập hợp nhiều phần tử, với mỗi phần từ là 1 biến được sắp xếp theo thứ tự và có đánh số chỉ mục (đánh số từ 0) cho từng biến để tiện truy xuất. Mỗi biến trong mảng có thể mang bất kì kiểu dữ liệu nào: từ Number, String đến cả đối tượng và mảng khác. Các biến trong mảng được đánh số tuần tự tăng dần từ 0, nên biến cuối cùng trong mảng sẽ có số chỉ mục bằng tổng số lượng biến trừ đi 1.

Mảng được khai báo bên trong cặp ngoặc vuông var tênMảng = [biến1, biến2];. Các biến bên trong cũng được truy xuất bằng ngoặc vuông theo cú pháp tênBiến[sốChỉMục]. Số lượng các phần tử bên trong mảng có thể được truy xuất nhanh bằng thuộc tính tênBiến.length (vì bản chất Mảng Javascript cũng là 1 đối tượng). Ngoài ra, để thêm phần tử vào 1 mảng đã được khởi tạo, bạn cần sử dụng phương thức tênBiến.push().

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42 | // Khai báo và khởi tạo mảng ví dụ  var phanlopAnhHung = ['Chiến Binh','Sát Thủ','Thợ Săn'];  console.log(phanlopAnhHung.length);  console.log(phanlopAnhHung[0]);  console.log(phanlopAnhHung[1]);  console.log(phanlopAnhHung[phanlopAnhHung.length - 1]);  // hàm console.log() rất hữu ích để xem toàn bộ phần tử trong mảng  console.log(phanlopAnhHung);    // Mảng rỗng  var danhHieu = [];    // Thêm phần tử mới  danhHieu.push('Dũng sĩ diệt rồng');  console.log(danhHieu);  phanlopAnhHung.push('Anh Hùng');  console.log(phanlopAnhHung.length);  console.log(phanlopAnhHung[phanlopAnhHung.length - 1]);    // Truy xuất từng phẩn tử trong mảng với vòng lặp  var soPhanTu = phanlopAnhHung.length;  for (var i=0; i < soPhanTu; i++) {  console.log(phanlopAnhHung[i ]);  }    /\* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \* THỬ THÁCH LEVEL 8  \* -----------------  \* Hãy sử dụng vòng lặp và hàm prompt() để yêu cầu người dùng  \* nhập vào câu trả lời cho 3 câu hỏi sau:  \* 1. Bạn có người yêu chưa?  \* 2. Bạn có thích ăn rau dền không?  \* 3. Người yêu bạn có thích ăn rau dền không?  \* Sau đó, lưu tất cả câu trả lời vào 1 mảng và xuất ngược lại  \* từng câu trả lời bằng hàm alert().  \*  \* =========================================================  \* | HÃY TỰ TAY CODE TRƯỚC KHI KÉO XUỐNG XEM CODE MẪU! |  \* =========================================================  \*/ |

[+ expand source](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#)

CHAPTER 1 COMPLETE! Chúc mừng bạn đã trở thành Tân Anh Hùng Javascript!

### **Lời Kết**

Con đường cách mạng, còn lắm gian truân! Vì thế, Javascript dù rất đơn giản để học, nhưng để sử dụng thành thục phải trải qua thử nghiệm và thực hành nhiều. Với kiến thức cơ bản mà bài viết này cung cấp, mình mong rằng bạn sẽ có nền tảng tốt để học nhanh và phát triển nhanh kĩ năng Javascript của bản thân. Nếu có thắc mắc, ý kiến hay cần trợ giúp gì, hãy cùng thảo luận ở phần bình luận bên dưới nhé.

BÀI NÀY:

<h1>Javascript cơ bản – Hành trình của một Anh Hùng</h1>  
<h3>Quest 1: Kiến tạo Anh Hùng – Biến và kiểu dữ liệu</h3>  
<h3>Quest 2: Sức mạnh tính toán – Các toán tử cơ bản</h3>  
<h3>Quest 3: Sức mạnh logic – Các toán tử so sánh</h3>  
<h3>Quest 3: Sức mạnh logic – Các toán tử so sánh</h3>  
<h3>Quest 5: Đấu trường Sinh Tử – Vòng lặp</h3>  
<h3>Quest 6: Tạo và gọi Hàm</h3>  
<h3>Quest 7: Đối tượng</h3>  
<h3>Lời Kết</h3>  
<h4>Biến</h4>  
<h4>Thao tác với biến</h4>