tự học javascript

### [Học Javascript - Tự học javascript miễn phí - freetuts](http://freetuts.net/hoc-javascript)

freetuts.net/hoc-javascript

Xin giới thiệu với các bạn tổng tất cả các bài viết học Javascript miễn phí dành cho tất cả các đối tượng. Đây là series **tự học Javascript** hoàn toàn miễn phí, mỗi ...

‎[Bài 13: DOM Element trong ...](http://freetuts.net/dom-element-trong-javascript-367.html) · ‎[Bài 16: DOM HTML trong ...](http://freetuts.net/dom-html-trong-javascript-370.html) · ‎[Javascript căn bản](http://freetuts.net/hoc-javascript/javascript-can-ban)

### [Học Javascript cơ bản và nâng cao, Học lập trình Javascript miễn phí ...](http://vietjack.com/javascript/)

vietjack.com/javascript/

**Học Javascript** cơ bản và nâng cao, Học lập trình Javascript miễn phí hay nhất ... phí hay nhất - **Học Javascript** cơ bản và nâng cao cho người mới học **từ** Cú ...

‎[JavaScript là gì](http://vietjack.com/javascript/javascript_la_gi.jsp) · ‎[Javascript - Cú pháp](http://vietjack.com/javascript/cu_phap_cua_javascript.jsp) · ‎[Javascript - Page Redirect](http://vietjack.com/javascript/dieu_huong_trang_trong_javascript.jsp) · ‎[Hộp thoại](http://vietjack.com/javascript/hop_thoai_alert_prompt_confirmation_trong_javascript.jsp)

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

### [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....

Chuyển đến [2.Quest 2: Sức mạnh tính toán – Các toán **tử** cơ bản](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#ftoc-heading-4) - **Javascript** hỗ trợ đầy đủ các toán**tử** cơ bản cộng trừ nhân chia. Ngoài ra, bạn sẽ làm ...

### [Học Javascript bằng tiếng Anh miễn phí với 6 website tốt nhất](https://thachpham.com/web-development/javascript/website-hoc-javascript-mien-phi-bang-tieng-anh.html)

https://thachpham.com/.../javascript/website-hoc-javascript-mien-phi-bang-tieng-anh....

8 thg 10, 2014 - Code Avengers là một website **học Javascript** và HTML/CSS **từ** cơ bản đến nâng cao thông qua hình thức vừa học vừa thực hành như ...

### [Hướng dẫn học Javascript trực tuyến miễn phí - Hoàng Luyến](https://hoangluyen.com/hoc-javascript/)

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

Xếp hạng: 1,8 - ‎42 phiếu bầu

26 thg 10, 2015 - Series bài **học** hướng dẫn **Javascript** toàn tập **từ** cơ bản đến nâng cao giúp bạn **tự học** lập trình Web hiệu quả **từ** Tài liệu, Video và Ebook ...

### [JavaScript cơ bản bài 1: Giới thiệu JavaScript - YouTube](https://www.youtube.com/watch?v=W1gcqM5pqVc)

[**▶ 8:03**](https://www.youtube.com/watch?v=W1gcqM5pqVc)

https://www.youtube.com/watch?v=W1gcqM5pqVc

10 thg 10, 2013 - Tải lên bởi JimiDARK

JavaScript cơ bản bài 4: Kiểu dữ liệu - Duration: 7:24. JimiDARK 13,879 views · 7:24 · [01] **Tự học Javascript** ...

### [Javascript (Jquery) #1: Cơ bản và cách tự học - YouTube](https://www.youtube.com/watch?v=SRD77huWlvM)

[**▶ 31:09**](https://www.youtube.com/watch?v=SRD77huWlvM)

https://www.youtube.com/watch?v=SRD77huWlvM

24 thg 11, 2014 - Tải lên bởi Simon Sai

Hướng dẫn cơ bản lập trình phía client bằng **Javascript** và Jquery, một số cách để bạn **tự** nghiên cứu code. Các bạn tải code mẫu ...

### [Chỉ dùm cách học javascript hiệu quả nhất! - programming - Dạy Nhau ...](https://daynhauhoc.com/t/chi-dum-cach-hoc-javascript-hieu-qua-nhat/21013)

https://daynhauhoc.com/t/chi-dum-cach-hoc-javascript-hieu-qua-nhat/21013

22 thg 2, 2016 - Là một lập trình viên em cảm thấy rất **tự** hào vì điều đó. Riêng cá nhân ... Anh có thể**học** tiếp về Node.js để chạy **Javascript** trên server. **Học** 1 ...

### [Giáo trình Giáo trình JavaScript - 109 trang tiếng việt - SinhvienIT](http://sinhvienit.net/forum/giao-trinh-javascript-109-trang-tieng-viet.9408.html)

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

28 thg 6, 2009 - CÁC TOÁN **TỬ** (OPERATOR) 17 2.1.1.Gán. 17 2.1.2.So sánh. 17 2.1.3.Số **học**. 18 2.1.4.Chuỗi 18 2.1.5.Logic. 19 2.1.6.Bitwise. 19. BÀI TẬP. 19

Bạn đã truy cập trang này 2 lần. Lần truy cập cuối: 27/02/2017

### [Tự học JavaScript/JavaScript cơ bản/Hàm – Wikibooks tiếng Việt](https://vi.wikibooks.org/wiki/T%E1%BB%B1_h%E1%BB%8Dc_JavaScript/JavaScript_c%C6%A1_b%E1%BA%A3n/H%C3%A0m)

https://vi.wikibooks.org/wiki/Tự\_học\_JavaScript/JavaScript\_cơ\_bản/Hàm

Hàm[sửa]. Hàm là một khối lệnh , một tập hợp của một hay nhiều hơn một dòng lệnh nằm giửa { và } dưới tên mục function bao gồm các loại hàm sau.

### [Khóa học lập trình JAVASCRIPT cơ bản - Học lập trình ... - MyClass.vn](http://myclass.vn/course/lap-trinh-javascript-co-ban/)

myclass.vn/course/lap-trinh-javascript-co-ban/

Đến với khóa **học** lập trình **Javascript** cơ bản này sẽ giúp các bạn hiểu sâu hơn. ... **JavaScript** có khái niệm **từ** khóa, do đó, **JavaScript** gần như không thể được ...

### [Javascript là gì, tự học javascript cơ bản cho người mới bắt đầu - Logo](http://congdongcoder.com/javascript-la-gi/)

congdongcoder.com/javascript-la-gi/

Qua các bài **học** về HTML và CSS, chúng ta đã biết HTML là thành phần cấu tạo nên một website và CSS giúp trang trí ( màu sắc, kích thước ... ) cho các thành ...

### [Tự học JavaScript/JavaScript cơ bản/Lệnh JavaScript – Wikibooks ...](https://vi.wikibooks.org/wiki/T%E1%BB%B1_h%E1%BB%8Dc_JavaScript/JavaScript_c%C6%A1_b%E1%BA%A3n/L%E1%BB%87nh_JavaScript)

https://vi.wikibooks.org/wiki/Tự\_học\_JavaScript/JavaScript\_cơ.../Lệnh\_JavaScript

**Javascript**[sửa]. **Javascript** là một ngôn ngữ lập trình cao cấp có cấu trúc dùng để lập trình trang mạng. Javacript được dùng trong trang mạng bằng cách dùng ...

### [Tự học JavaScript/JavaScript cơ bản – Wikibooks tiếng Việt](https://vi.wikibooks.org/wiki/T%E1%BB%B1_h%E1%BB%8Dc_JavaScript/JavaScript_c%C6%A1_b%E1%BA%A3n)

https://vi.wikibooks.org/wiki/Tự\_học\_JavaScript/JavaScript\_cơ\_bản

Lập trình cơ bản[sửa]. Lập trình cơ bản để làm trang mạng HTML thêm linh động thí dụ như thay đổi màu nền của trang mạng, tạo công cụ như đồng hồ, lịch để ...

### [Tự học JavaScript – Wikibooks tiếng Việt](https://vi.wikibooks.org/wiki/T%E1%BB%B1_h%E1%BB%8Dc_JavaScript)

https://vi.wikibooks.org/wiki/Tự\_học\_JavaScript

**JavaScript**[sửa]. **JavaScript** (JS) sáng chế bởi Netscape là một ngôn ngữ dùng để lập Trình Trang Mạng HTML cơ bản và theo hướng Đối Tượng DOM để làm ...

### [Giáo trình tự học HTML, CSS, Javascript cho người mới học](http://hocasp.net/giao-trinh-tu-hoc-html-css-javascript-cho-nguoi-moi-hoc/)

hocasp.net/giao-trinh-tu-hoc-html-css-javascript-cho-nguoi-moi-hoc/

11 thg 4, 2015 - Share giáo trình **tự học** HTML và CSS tiếng việt do hocasp,net sưu tầm. ... 0.0.1 Vì sao bạn nên **học** HTML, CSS và **JAVASCRIPT** trước khi **học** ...

Bạn đã truy cập trang này vào ngày 28/02/2017.

### [Tự Học Javascript - Hiosdroid](http://hiosdroid.com/sharing/tu-hoc-javascript/)

hiosdroid.com/sharing/tu-hoc-javascript/

29 thg 10, 2016 - Để **học Javascript** thì khá dễ dàng. Bài viết này chia sẻ các tài liệu và các kênh online sẽ giúp ích cho các bạn trong việc **tự** học ngôn ngữ ...

### [Ngôn ngữ JavaScript là tương lai của lập trình? - Techmaster](https://techmaster.vn/posts/33621/ngon-ngu-javascript-la-tuong-lai-cua-lap-trinh)

https://techmaster.vn › Blog

30 thg 10, 2015 - **Học** lập trình web trực tuyến cơ bản đến nâng cao ... Ngoài ra, kể **từ** khi **JavaScript**chạy trong một trình duyệt, bạn không cần phải tải một phần ...

### [Tự Học JavaScript - Dragonbax196 - Wattpad](https://www.wattpad.com/story/1995963-t%E1%BB%B1-h%E1%BB%8Dc-javascript)

https://www.wattpad.com/story/1995963-tự-học-javascript

**JAVASCRIPT** 1, Mở đầu về **Javascript Javascript** là ngôn ngữ thứ hai mình muốn giời thiệu với các bạn vừa mới bước chân vào con đường wapmaster, ...

### [Học Javascript cơ bản cho người mới bắt đầu 2016 - Blog Lợi](http://blogloi.com/hoc-javascript-co-ban/)

blogloi.com › Javascript căn bản

Xếp hạng: 5 - ‎2 phiếu bầu

14 thg 1, 2016 - Hướng dẫn học javascript cơ bản cho người mới bắt đầu, **tự học javascript** cơ bản tại nhà với series từ javascript đến framework jquery mới ...

# **Học Javascript**

Đăng bởi: Administrator

Xin giới thiệu với các bạn tổng tất cả các bài viết học Javascript miễn phí dành cho tất cả các đối tượng. Đây là series tự học Javascript hoàn toàn miễn phí, mỗi series được tổng hợp các bài viết phân chia từng bài rất cụ thể và rõ ràng giúp người học có thể follow và học tại nhà.

Như các bạn biết khi học [lập trình web](http://freetuts.net/dinh-nghia/lap-trinh-web-la-gi-thiet-ke-web-la-gi-11.html) nói chung và học về [lập trình frontend](http://freetuts.net/dinh-nghia/frontend-developer-la-gi-backend-developer-la-gi-15.html) nói riêng thì bắt buộc bạn phải học javascript, một website thiếu Javascript thì sẽ trở nên cọc cằn, chức năng không đẹp và không thân thiện với người dùng. vì vậy nếu bạn không học JS thì đó quả là một sai lầm và lạc hậu về công nghệ.

# DANH SÁCH SERIES

* + [JAVASCRIPT CĂN BẢN](http://freetuts.net/hoc-javascript/javascript-can-ban)
  + [BÀI TẬP JAVASCRIPT](http://freetuts.net/hoc-javascript/bai-tap-javascript)
  + [JAVASCRIPT NÂNG CAO](http://freetuts.net/hoc-javascript/javascript-nang-cao)
  + [JAVASCRIPT TUTORIALS](http://freetuts.net/hoc-javascript/javascript-tutorials)
  + [JAVASCRIPT FUNCTION](http://freetuts.net/hoc-javascript/function)  
    bài viết 1:  
    <h1 >Học Javascript</h1>  
    <h1>DANH SÁCH SERIES</h1>

# 

# 

# Học Javascript cơ bản và nâng cao, Học lập trình Javascript miễn phí hay nhất

**[Học Javascript cơ bản và nâng cao, Học lập trình Javascript miễn phí hay nhất]** JavaScript là một ngôn ngữ chương trình thông dịch, nhẹ. Nó được thiết kế để tạo các ứng dụng mạng trung tâm. Nó là ngôn ngữ tích hợp với Java, được lấy tên là JavaScript vì thời đó Java là một hiện tượng, nhưng trên thực tế JavaScript không được phát triển dựa trên Java. JavaScript rất dễ dàng để thực hiện bởi vì nó được tích hợp với HTML. Đây là ngôn ngữ chương trình mở và đa nền tàng.

JavaScript là một ngôn ngữ lập trình của HTML và WEB. Nó là nhẹ và được sử dụng phổ biến nhất như là một phần của các trang web, mà sự thi hành của chúng cho phép Client-Side script tương tác với người sử dụng và tạo các trang web động. Nó là một ngôn ngữ chương trình thông dịch với các khả năng hướng đối tượng.

Các file JavaScript thường được lưu dưới đuôi **js** và được nhúng vào các trang HTML. Do đó, để hiểu tác dụng của JavaScript thì điều kiện tiền đề là bạn cần phải biết cách HTML hoạt động. Nếu bạn còn cảm thấy mới mẻ với HTML, bạn có thể tham khảo tham khảo loạt bài [**hướng dẫn HTML cơ bản và nâng cao**](http://vietjack.com/html/index.jsp) ngay trên trang của chúng mình.

Dưới đây là danh sách các bài hướng dẫn học JavaScript cơ bản và nâng cao:

**MỤC LỤC**

## Các khái niệm JavaScript cơ bản

* [**Javascript là gì**](http://vietjack.com/javascript/javascript_la_gi.jsp)
* [**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)
* [**Javascript - Lệnh Switch**](http://vietjack.com/javascript/lenh_switch_case_trong_javascript.jsp)
* [**Javascript - Vòng lặp While**](http://vietjack.com/javascript/vong_lap_while_trong_javascript.jsp)
* [**Javascript - Vòng lặp For**](http://vietjack.com/javascript/vong_lap_for_trong_javascript.jsp)
* [**Javascript - Vòng lặp For...in**](http://vietjack.com/javascript/vong_lap_forin_trong_javascript.jsp)
* [**Javascript - Điều khiển vòng lặp**](http://vietjack.com/javascript/dieu_khien_vong_lap_break_va_continue_trong_javascript.jsp)
* [**Javascript - Hàm**](http://vietjack.com/javascript/ham_trong_javascript.jsp)
* [**Javascript - Sự kiện**](http://vietjack.com/javascript/su_kien_trong_javascript.jsp)
* [**Javascript - Cookie**](http://vietjack.com/javascript/javascript_va_cookies.jsp)
* [**Javascript - Page Redirect**](http://vietjack.com/javascript/dieu_huong_trang_trong_javascript.jsp)
* [**Javascript - Hộp thoại**](http://vietjack.com/javascript/hop_thoai_alert_prompt_confirmation_trong_javascript.jsp)
* [**Javascript - Từ khóa Void**](http://vietjack.com/javascript/tu_khoa_void_trong_javascript.jsp)
* [**Javascript - In trang**](http://vietjack.com/javascript/in_trang_trong_javascript.jsp)

## Đối tượng trong JavaScript

* [**Javascript - Đối tượng (Object)**](http://vietjack.com/javascript/doi_tuong_trong_javascript.jsp)
* [**Javascript - Number**](http://vietjack.com/javascript/doi_tuong_number_trong_javascript.jsp)
* [**Javascript - Boolean**](http://vietjack.com/javascript/doi_tuong_boolean_trong_javascript.jsp)
* [**Javascript - String**](http://vietjack.com/javascript/doi_tuong_string_trong_javascript.jsp)
* [**Javascript - Array**](http://vietjack.com/javascript/doi_tuong_array_trong_javascript.jsp)
* [**Javascript - Date**](http://vietjack.com/javascript/doi_tuong_date_trong_javascript.jsp)
* [**Javascript - Math**](http://vietjack.com/javascript/doi_tuong_math_trong_javascript.jsp)
* [**Javascript - RegExp**](http://vietjack.com/javascript/regular_expression_va_regexp_trong_javascript.jsp)
* [**Javascript - HTML DOM**](http://vietjack.com/javascript/document_object_model_trong_javascript.jsp)

## Hoạt động JavaScript nâng cao

* [**Javascript - Xử lý lỗi**](http://vietjack.com/javascript/xu_ly_loi_trong_javascript.jsp)
* [**Javascript - Form Validation**](http://vietjack.com/javascript/form_validation_trong_javascript.jsp)
* [**Javascript - Animation**](http://vietjack.com/javascript/hieu_ung_trong_javascript.jsp)
* [**Javascript - Đa phương tiện**](http://vietjack.com/javascript/da_phuong_tien_trong_javascript.jsp)
* [**Javascript - Debug**](http://vietjack.com/javascript/debug_trong_javascript.jsp)
* [**Javascript - Image Map**](http://vietjack.com/javascript/image_map_trong_javascript.jsp)
* [**Javascript - Trình duyệt**](http://vietjack.com/javascript/kha_nang_tuong_thich_trinh_duyet_trong_javascript.jsp)

## Các hàm JavaScript có sẵn

* [**Javascript - Danh sách các hàm**](http://vietjack.com/javascript/cac_ham_co_san_trong_javascript.jsp)

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

**Chú ý: Link các bài học ở menu bên trái**. Nếu không xem được, vui lòng xem hướng dẫn[**tại đây**](http://vietjack.com/images/bai_hoc.PNG)

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

Bài viết số 2:

<h1>Học Javascript cơ bản và nâng cao, Học lập trình Javascript miễn phí hay nhất</h1>  
<h2>Các khái niệm JavaScript cơ bản</h2>  
<h2>Đối tượng trong JavaScript</h2>  
<h2>Hoạt động JavaScript nâng cao</h2>  
<h2>Các hàm JavaScript có sẵn</h2>

# **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! |  \* =========================================================  \*/ |

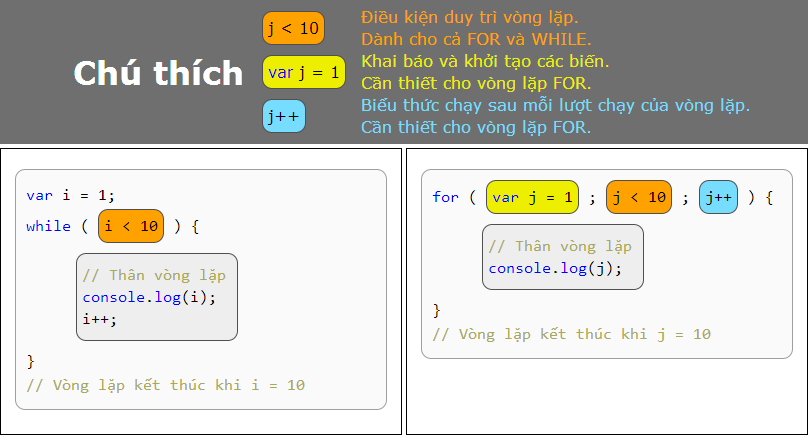
[+ expand source](https://thachpham.com/web-development/javascript/javascript-co-ban-hanh-trinh-cua-mot-anh-hung.html#)

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>