

**Karmi Phuc**



# JavaScript dễ dàng với Chrome Dev Tools

Phiên bản tiếng Anh đầy đủ hơn: <http://karmiphuc.com/cong-nghe/4-tricks-to-debug-javascript-like-pro/>

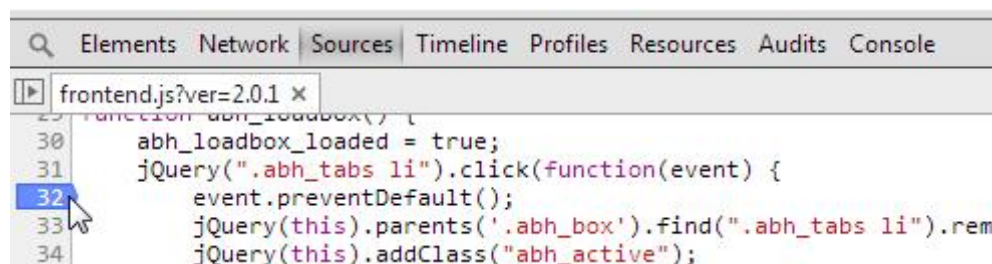
Google Chrome đang trên quá trình trở thành trình duyệt hàng đầu cho mọi người dùng với mọi nhu cầu. Đối với riêng lập trình viên, Chrome Dev Tools là một tiện ích đặc lực để giúp giảm thiểu những phiền phức và rắc rối khi lập trình web nói chung và lập trình Javascript nói riêng. Đối với mình, việc sử dụng Chrome Dev Tools thậm chí còn mang lại hiệu quả cao hơn hẳn so với lão làng Firebug, với hàng loạt mẹo vặt giúp giải quyết vấn đề tốt trong thời gian ngắn hơn.

## 1. Kiểm soát luồng chạy với Breakpoints

Một **breakpoint** là một điểm dừng có chủ đích trong code, dùng để hỗ trợ lập trình viên trong quá trình debug. Khi chương trình chạy đến một breakpoint, bạn có thể "bước qua" (step-through) từng dòng lệnh và kiểm tra xem mọi logic có đúng như ý đồ. Ở trạng thái dừng của breakpoint, bạn cũng có thể kiểm tra giá trị của từng biến tại thời điểm hiện tại. Điều này giúp xác định một lỗi thường gặp nhưng ít được để ý: sử dụng một biến khi chưa khởi tạo biến đó.

Có 2 cách để đặt breakpoint:

1. Bằng tay: bấm vào số dòng của code trong tab Source



Đặt breakpoint bằng tay ở tab Sources trong Chrome Dev Tools

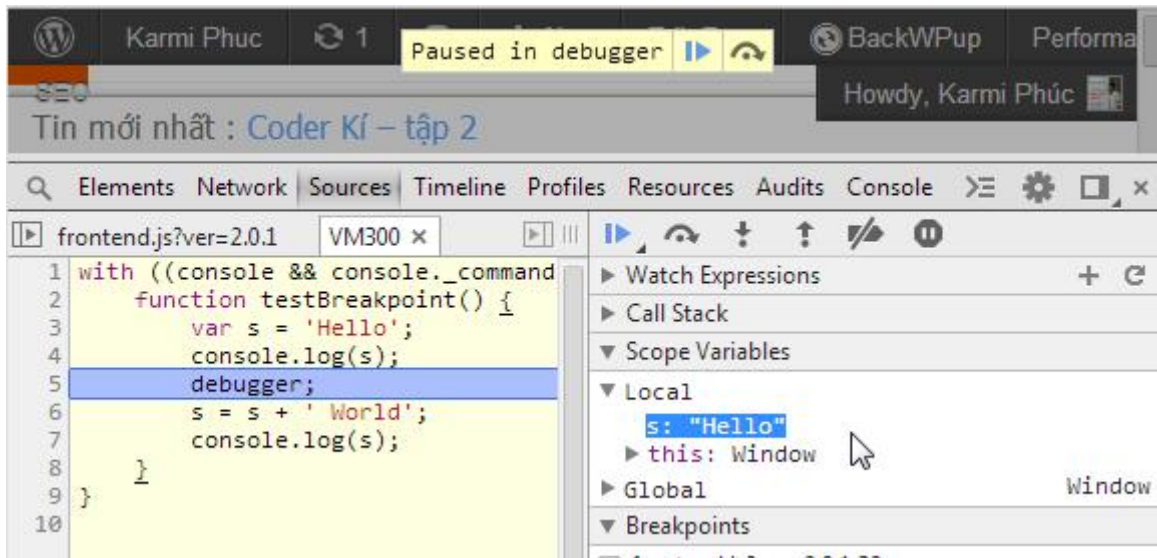
2. Bằng code: sử dụng lệnh `debugger;` trong code.

```
1 | var s = 'Hello' ;  
2 | console.log(s);
```

```

3 | debugger;
4 | s = s + ' World' ;
5 | console.log(s);

```



Kết quả chạy của đoạn code trên

Sau khi breakpoint được gọi, bạn có thể kiểm soát quá trình thực hiện của đoạn code tiếp theo bằng các nút sau:

**Continue:** chương trình sẽ tiếp tục chạy cho đến khi gặp một breakpoint khác (nếu có).

**Step over:** chạy từng dòng lệnh, nhưng sẽ không nhảy vào trong các hàm khác (xem ví dụ và giải thích thêm ở đây <http://jsfiddle.net/karmiphuc/C5kMy/>).

**Step into:** chạy từng dòng lệnh giống như *Step over*, nhưng sẽ nhảy vào từng dòng lệnh trong các hàm khác (xem ví dụ và giải thích thêm ở đây <http://jsfiddle.net/karmiphuc/C5kMy/>).

## 2. Thuần phục console API

Console API là tên gọi chung của các hàm được định nghĩa sẵn trong Chrome Dev Tools. Các hàm này có thể được gọi thông qua đối tượng toàn cục `console`, và dùng để hiển thị bất cứ thông tin gì bạn muốn xem: giá trị của một biến, một đối tượng, hay thậm chí một DOM element. Các thông tin này được xuất ra ở tab Console trong Chrome Dev Tools.

Các hàm thường được dùng nhất trong Console API:

## console.log

Hàm này sẽ in giá trị của một hay nhiều biến/đối tượng/công thức truyền vào.

```
> console.log(new Date())
Sun Oct 20 2013 17:49:43 GMT+0700 (SE Asia Standard Time)
< undefined
> console.log("Hello, Kami. Current timestamp is ", new Date())
Hello, Kami. Current timestamp is Sun Oct 20 2013 17:51:06 GMT+0700 (SE Asia Standard Time)
< undefined
```

In giá trị của ngày hiện tại và in kèm theo một chuỗi

## console.dir

`console.dir()` liệt kê tất cả thuộc tính bên trong một đối tượng hay một mảng trong Javascript.

```
> console.dir([1,2,3,"tada"])
▼ Array[4] 3
  0: 1
  1: 2
  2: 3
  3: "tada"
  length: 4
  ▶ __proto__: Array[0]
< undefined
```

## console.table

Đây có lẽ là hàm hữu ích nhất khi làm việc với nhiều đối tượng và mảng. Hàm này sẽ giúp in các giá trị của mảng hay đối tượng dưới dạng bảng. Bảng này hỗ trợ lập trình viên để quan sát, so sánh và sắp xếp các giá trị theo ý muốn.

```
> console.table([{"id":1,name:"kami"}, {"id":2,name:"phuc"}])
```

(index)	id	name
0	1	"kami"
1	2	"phuc"

```
< undefined
```

## console.time

Nếu bạn quan tâm đến tốc độ chạy của các hàm hay thuật toán, đây là hàm dành riêng cho bạn khi giúp đo thời gian chạy giữa `console.time()` và `console.timeEnd()` theo mili giây.

```

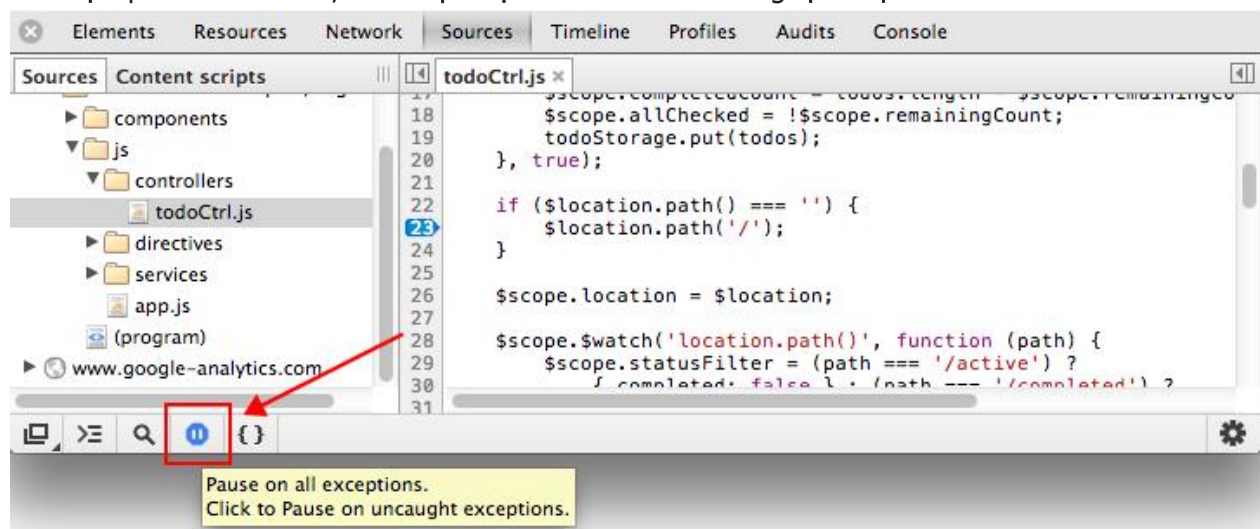
> console.time('start WS_timer')
undefined
> console.timeEnd('start WS_timer')
start WS_timer: 6301.000ms
< undefined
> console.time('start WS_timer 2');
for (var i = 0; i<1017; ++i) {} // Wasting some time here...
console.timeEnd('start WS_timer 2');
start WS_timer 2: 2.000ms
< undefined

```

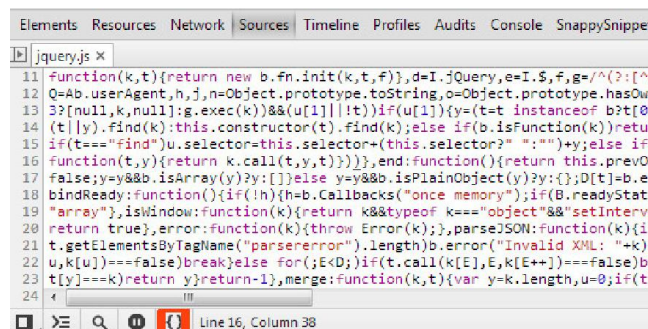
### 3. Giải minify và làm đẹp code

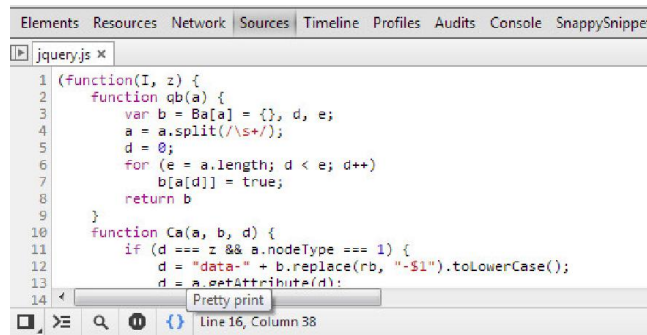
Minify Javascript được dùng để giảm kích thước truyền tải và tăng tốc website. Tuy nhiên, đôi khi bạn phải đào bới trong đống code Javascript đã được minify để tìm kiếm lỗi, và đó thực sự là một nỗi ác mộng.

Tuy nhiên, nếu bạn đang sử dụng Chrome Dev Tools, bạn có thể thở phào nhẹ nhõm. Bạn chỉ cần bật qua tab Sources, và nhấp chọn vào nút có hình ngoặc nhọn



("Pretty Print") ở góc dưới bên trái. Khi đó, toàn bộ code Javascript sẽ được giải minify và trở về format dễ đọc, dễ debug và dễ đặt breakpoint hơn.





## 4. Pro tips khi debug với tab Sources của Chrome Dev Tools

- Chắc chắn bạn hiểu cấu trúc và logic của chương trình trước khi cần sử dụng breakpoint.
- Cần cân nhắc vị trí đặt breakpoint. Quá nhiều breakpoint sẽ khiến quá trình debug bị rối và không hiệu quả.
- Sẽ có một bài viết khác để giải thích và ví dụ thêm về việc sử dụng breakpoint.
- Sử dụng Ctrl+O (Win) hay Cmd+O để mở danh sách các file Javascript và CSS đang có.
- Tổ hợp Ctrl+Shift+O (Win) hay Cmd+Shift+O sẽ hiện danh sách các hàm Javascript và các CSS selector hiện có.
- Khi lập trình web ở máy tính local, bạn có thể chỉnh sửa trực tiếp và lưu lại dễ dàng với chức năng Save/Save as của Chrome Dev Tools.

### LỜI KẾT

Còn bạn thì sao? Bạn nghĩ sao về Chrome Dev Tools và những mẹo lập trình tên? Hãy chia sẻ thêm mẹo lập trình mà bạn đang áp dụng để làm việc hiệu quả với mình ở comment nha!

Tác giả: [Karmi Phúc](#).

Phiên bản tiếng Anh đầy đủ hơn: <http://karmiphuc.com/cong-nghe/4-tricks-to-debug-javascript-like-pro/>

Tham khảo thêm nhiều nhiều nữa về Chrome Dev Tools:  
<https://developers.google.com/chrome-developer-tools/>

[Share on Facebook](#)

*Read more? Quick thoughts from [?#karmiphucblog](#) on Facebook.*

*Follow me on [Kipalog](#), or [Medium](#) or at [my personal Facebook](#).*

📁 Công nghệ, Web Development

🔑 chrome, code, devtools, javascript

💬 3 Comments

---

PREVIOUS

**7 mẹo ít biết để tối ưu hóa HTML/CSS/JS đúng chuẩn**

---

NEXT

**Theo đuổi ước mơ – Sống cho ra sống**

---

## 3 thoughts on “Debug Javascript dễ dàng với Chrome Dev Tools”



**Trần Hoàng Thái**

23/07/2014 at 7:23 am

có phải a phúc hồi giao lưu ở giảng đường 1 UIT k ta

REPLY



**Karmi Phuc**

23/07/2014 at 8:01 am

hi Thái, mình là Phúc IS02 (2007) 😊 Không biết đợt đó là sự kiện nào ha, Ngày hội khoa HTTT hay là tuần sinh hoạt cuối khóa?

REPLY

**Lôi Hạ**

03/12/2014 at 12:21 am

Ngày hội khoa HTTPT đấy anh Phúc

REPLY

## Leave a Reply

Your email address will not be published. Required fields are marked \*

**COMMENT****NAME \*****EMAIL \*****WEBSITE**

POST COMMENT