

[Trang Chủ](#) / [Bài Viết](#) / Điểm Danh Những HTTP Status Code Thông Dụng Developer Cần Phải Thuộc Lòng

Điểm Danh Những HTTP Status Code Thông Dụng Developer Cần Phải Thuộc Lòng

Nam Nguyễn 16 June 2017

Hình dung bạn cần xây dựng một web service API dùng để quản lý các bài viết. Câu hỏi đặt ra là service này sẽ cần trả về status code là gì khi client gửi request để cập nhật nội dung một bài viết với `post_id` có giá trị là một chuỗi thay vì đúng ra phải là một số integer:

```
{
  "post_id": "haha",
  "content": "hihi"
}
```

Tương tự service này sẽ cần trả về status code là gì khi client gửi request để import danh sách bài viết được export trước đó vào database nhưng thay vì gửi file **XML** hoặc **JSON** thì client lại gửi một file binary?

Đối với các developer việc nắm vững ý nghĩa của các HTTP status code là rất quan trọng khi xây dựng các ứng dụng web đóng vai trò làm service API. Trả về status code đúng ý nghĩa sẽ giúp phía client có thể quyết định chính xác các bước tiếp xử lý theo sau như thế nào. Mặc dù vậy thì không nhiều developer hiểu đúng ý nghĩa của các status code.

HTTP Status Code Là Gì

HTTP Status Code là một mã gồm 3 chữ số được gửi trả về cho client từ server dùng để mô tả trạng thái của quá trình server xử lý một yêu cầu (request) cho trước gửi từ client tới server dưới giao thức HTTP.

Lưu ý rằng tác dụng duy nhất của HTTP status code là để thông tin về trạng thái xử lý request của server.

Kiểm Tra HTTP Status Code Trên Trình Duyệt

HTTP status code được gửi trả về trong **response header** trả về từ server.

Chúng ta có thể kiểm tra HTTP status code của một request sử dụng trình duyệt. Đầu tiên trên trình duyệt bạn mở [developer tools](#) sử dụng phím **F12** và click vào tab *Network*. Sau đó truy cập vào một trang bất kỳ ví dụ như Facebook, bạn sẽ thấy thông tin về HTTP status code của các request hiển thị ở cột status như sau:

Network									
View: [Icons] Preserve log [x] Disable cache [x] Offline No throttling [v]									
Name	Status	Type	Initiator	Size	Time	Waterfall	40.00		
lumbar.gif?a=%7B%22UID%22%3A%225f5d5983fda1090cc717c9f53d0f1e7c6d210c0013ead06f71481... p.imgur.com	200 OK	gif	Other		637 B 4 B	261 ms 218 ms			
LRAWPHjb.jpg l.imgur.com	200 OK	gif	(index) Parser	10.9 KB 10.2 KB	224 ms 199 ms				
chUAkrhb.jpg l.imgur.com	200 OK	jpeg	(index) Parser	5.0 KB 4.4 KB	333 ms 223 ms				
button.b731e6c82269f70f6fc53634e044a4c6.js platform.twitter.com/js	200 OK	script	widgets.js:1 Script	4.6 KB 4.2 KB	286 ms 275 ms				
collect?v=1&_v=56&a=732890609&t=pageview&_s=1&d=http%3A%2F%2Fimgur.com%2F&ul=en-us&d... www.google-analytics.com/r	200	gif	Other	187 B 35 B	216 ms 210 ms				
742377892535530?v=2.7.16 connect.facebook.net/signals/config	307 Internal Redirect		fbevents.js:24 Script	0 B 0 B	41 ms 9 ms				
xDivTtbb.jpg l.imgur.com	200 OK	jpeg	xDivTtbb.jpg Redirect	10.1 KB 9.5 KB	227 ms 198 ms				
iu37cm3ppd=1&d=dtb-pub&csif=t&d=brt_pm_an s.amazon-adsystem.com	200 OK	document	amzn_ads.js:1 Script	272 B 64 B	279 ms 258 ms				
pixel?r=1223212340;rf=0;a=p-f8oruOqDFMeI?url=http%3A%2F%2Fimgur.com%2F&pan=0;fpa=P0-1825... pixel.quantserve.com	200 OK	gif	Other	287 B 35 B	54 ms 47 ms				
742377892535530?v=2.7.16 connect.facebook.net/signals/config	200	script	742377892535530 Redirect	8.8 KB 27.1 KB	84 ms 81 ms				
follow_button.7bf2b0e802ada47dae9548b7a1739fed.en.html platform.twitter.com/widgets	200 OK	document	widgets.js:10 Script	13.0 KB 33.1 KB	44 ms 31 ms				
?id=742377892535530&ev=PageView&d=http%3A%2F%2Fimgur.com%2F&r=&if=false&ts=14975998... www.facebook.com/tr	200	gif	Other	164 B 44 B	73 ms 66 ms				
?id=742377892535530&ev=Microdata&d=http%3A%2F%2Fimgur.com%2F&r=&if=false&ts=14975998... www.facebook.com/tr	200	gif	Other	98 B 44 B	79 ms 66 ms				
data:image/svg+xml,...	200 OK	svg+xml	follow_button.7bf2b0e...en.html:44 Script	(from memory cache)	0 ms 0 ms				
jot?l=%7B%22widget_origin%22%3A%22http%3A%2F%2Fimgur.com%2F%22%2C%22widget_frame%... syndication.twitter.com/i	200	gif	Other	480 B 43 B	261 ms 247 ms				
Favicon-16x16.png s.imgur.com/images	200 OK	png	Other	15.0 KB 14.4 KB	223 ms 220 ms				
Favicon-96x96.png	200	png	Other	15.9 KB	195 ms				
1131 requests 6.7 MB transferred Finish: 41.34 s DOMContentLoaded: 3.62 s Load: 5.13 s									
Console									

Ở hình trên bạn thấy ngoài một request trả về HTTP status code là **307** (Internal redirect) thì hầu hết các request còn lại đều trả về trạng thái **200** (OK). Trong phần tiếp theo dưới đây chúng ta sẽ tìm hiểu kỹ hơn về ý nghĩa của các con số này.

Khi click vào một request bất kỳ bạn cũng sẽ thấy thông tin về HTTP status code hiển thị ở phía dưới mục **General** bên phải như hình dưới đây:

The screenshot shows the Chrome DevTools Network tab. On the left, a list of requests is displayed, including hUCaT, css?family=Open+Sans, global.css, gallery.css, jquery.min.js, react15.js, global.js (selected), advertising.js, px.js, runSlots.js, gallery.js, U2Q0uUgg.png, like.php, imgur-logo.svg, imgur.woff, and cJZKeOuBrn4kERxqtaUH3VtXRa8TVwTICgirnJhmVJw.woff2. The right pane shows the details for the selected request (global.js?1497566795) with a status of 200 OK. The General tab is active, showing the Request URL, Request Method (GET), Status Code (200 OK), Remote Address (151.101.40.193:80), and Referrer Policy (no-referrer-when-downgrade). The Response Headers tab is also visible, showing headers like Accept-Ranges, Age, Connection, Content-Encoding, Content-Length, Content-Type, Date, ETag, Fastly-Debug-Digest, Last-Modified, Server, Vary, Via, X-Cache, X-Cache-Hits, X-origin-ip, X-Served-By, and X-Timer. The Request Headers tab shows headers like Accept, Accept-Encoding, Accept-Language, Cache-Control, Connection, Cookie, Host, Pragma, Referer, and User-Agent. The Query String Parameters tab shows the parameter 1497566795.

Ý Nghĩa Các HTTP Status Code Thông Dụng

Các HTTP status code có 3 chữ số và được phân thành 5 loại chính dựa vào chữ số đầu tiên:

- 1xx: Các status code loại này dùng để đơn giản thông báo với client rằng server đã **nhận** được request. Các status code 1xx ít được sử dụng và thậm chí không được định nghĩa trong HTTP phiên bản 1.0.
- 2xx: Các status code loại này có ý nghĩa rằng request được server **nhận**, **hiểu** và **xử lý** thành công.
- 3xx: Các status code loại này có ý nghĩa rằng server sẽ chuyển tiếp request hiện tại sang một request khác và client cần thực hiện việc gửi request tiếp theo đó để có thể hoàn tất. Thông thường khi trình duyệt nhận được status code loại này nó sẽ tự động thực hiện việc gửi request tiếp theo để lấy về kết quả.
- 4xx: Các status code loại này có ý nghĩa rằng đã có lỗi từ phía *client* trong khi gửi request. Ví dụ như sai URL, sai HTTP method, không có quyền truy cập vào trang...
- 5xx: Các status code loại này có ý nghĩa rằng server đã có lỗi từ phía *server* trong khi xử lý request. Ví dụ như database chết hoặc server bị hết bộ nhớ...

Các Status Code Thông Dụng

Ở phần này chúng ta sẽ cùng nhau tham khảo các status code hay sử dụng khi xây dựng web service (hoặc sử dụng API).

2xx: Thành Công

Trong các HTTP status code thuộc loại 2xx thì **200** OK là status code thường được sử dụng nhiều. Các request được xử lý thành công bởi server thường sử dụng status là **200**.

3xx: Chuyển Tiếp

Trong các HTTP status code thuộc loại 3xx thì **301** Moved Permanently, **302** Found và **307** Temporary Redirect thường được sử dụng nhiều. Trong khi **302** và **307** được sử dụng cho các request mà việc chuyển tiếp mang ý nghĩa tạm thời thì **301** sử dụng cho các request mà việc chuyển tiếp là cố định. Sự khác biệt ở đây đó là với các request trả về **301** thì ở các phiên làm việc tiếp theo client không nên gửi lại request này mà nên sử dụng request mới (hay request được chuyển tiếp).

Trong khi đó với các request trả về **302** và **307** thì client vẫn nên sử dụng request này thay vì request mới (request được chuyển tiếp) vì việc chuyển tiếp chỉ diễn ra tạm thời.

4xx: Lỗi Từ Phía Client

Trong các HTTP status code thuộc loại 4xx thì **400** Bad Request, **401** Unauthorized, **403** Forbidden và **404** Not Found là status code thường được sử dụng nhiều. Trong đó:

- 400** có ý nghĩa rằng server đã nhận được request nhưng phía client đã gửi một request lỗi. Một trong số lỗi có thể kể đến như lỗi cú pháp như upload tập tin không đúng format hoặc quá lớn, hoặc thiếu dữ liệu cần thiết...
- 401** có ý nghĩa rằng server đã nhận được request nhưng phía client không có quyền truy cập để xem hoặc chỉnh sửa hoặc xóa nội dung yêu cầu. Các request trả về status code **401** sẽ thay đổi nếu phía client thực hiện việc đăng nhập.
- 403** có ý nghĩa rằng server đã nhận được request nhưng từ chối việc xử lý request. Một trường hợp phổ biến mà status code **403** trả về đó là với các request muốn lấy về danh sách file trong một thư mục trên server ví dụ request tới địa chỉ `http://example.net/danh-sach-bang-diem/K07-A2.html` sẽ trả về bảng điểm lớp **A2** khóa **K07** nhưng request tới địa chỉ `http://example.net/danh-sach-bang-diem/` sẽ trả về status code là 403.
- 404** có ý nghĩa rằng server đã nhận được request nhưng không tìm thấy nội dung nào trả về với URL nhận được.

5xx: Lỗi Từ Phía Server

Trong các HTTP status code thuộc loại 5xx thì **500** Internal Server Error là status code thường được sử dụng nhiều. Status code **500** là một status với ý nghĩa chung nói rằng server đã gặp phải lỗi khi xử lý request mà không đưa ra lý do cụ thể.

Trên đây bạn đã tham khảo các HTTP status code thường hay sử dụng nhất. Để tham khảo chi tiết tất cả các HTTP status code bạn có thể tham khảo địa chỉ [này](#).

HTTP

API

Server

Service

Lập Trình Web

[Chỉnh sửa](#)

1 Phản Hồi

[A Nguyen](#) 02 August 2018

Bài viết rất hay

[Thêm bình luận](#)

2

Vote

Thêm Phản Hồi

Nội dung tối thiểu 10 ký tự...

Gửi

Bài Viết Liên Quan

[Giao Thức HTTP/2 Có Gì Mới](#)

19 March 2016

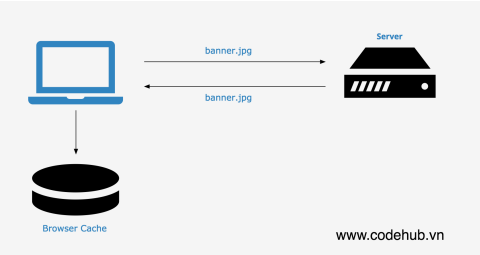


Khác Biệt Giữa Phương Thức POST và GET

25 April 2017

Các Web Developer Đã Sử Dụng "Cache-Control" Như Thế Nào

19 November 2017



[Fixed] Warning: Cannot modify header information - headers already sent by

21 November 2017

```
1 <!-- Khoảng Trống -->
2 <?php
3 // ...
4 header("location: /login.php");
5 ?>
6 <!-- Khoảng Trống -->
7
```

Hướng Dẫn Xử Lý Lỗi 404 Error trong Laravel

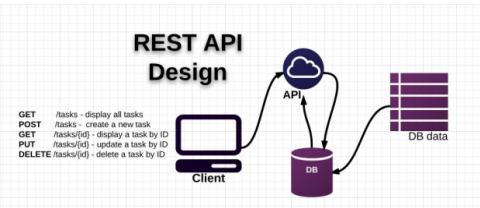
27 December 2017



Bài Viết Nổi Bật

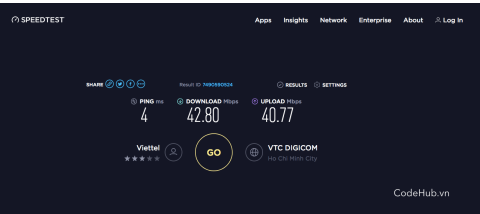
RESTful API Cho Người Bắt Đầu

13 March 2017



Kiểm tra tốc độ đường truyền internet với Speedtest

22 July 2018

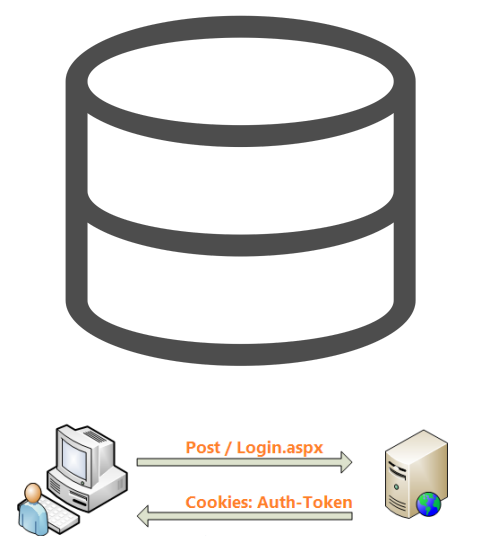


JSON Là Gì và Sử Dụng JSON Như Thế Nào

12 March 2017

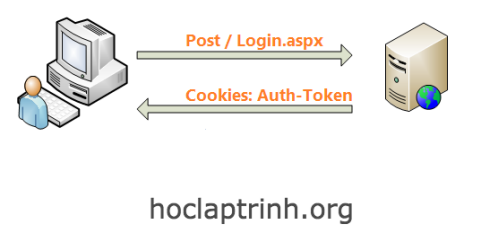
Cơ Sở Dữ Liệu và Hệ Quản Trị Cơ Sở Dữ Liệu

19 September 2015



Session Là Gì?

19 March 2016



Thiết kế và phát triển bởi [CodeHub](#) © 2019