HTTP: Response Messages

Let's look at what HTTP response messages look like!

WE'LL COVER THE FOLLOWING

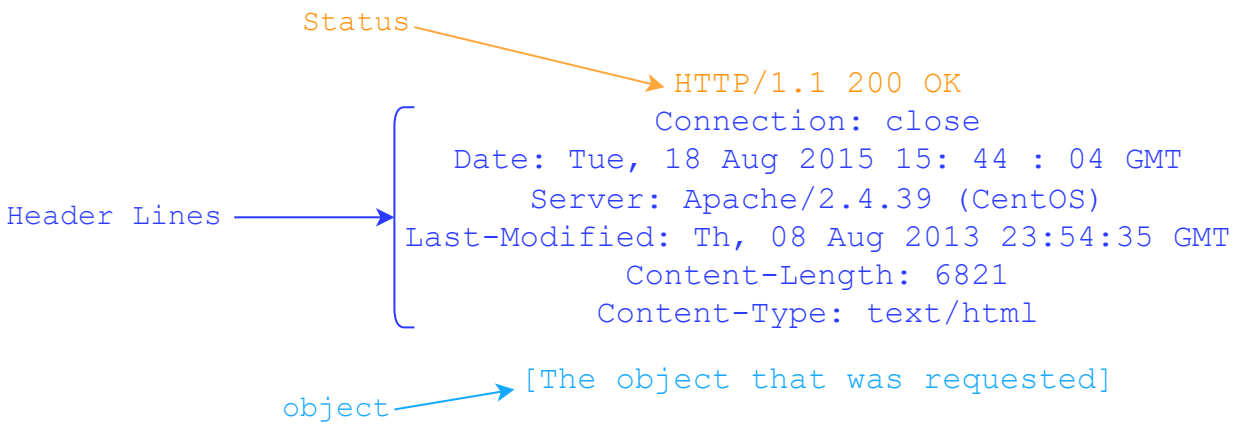
Introduction

Let’s start with a typical example of an HTTP response message:

```
HTTP/1.1 200 OK
Connection: close
Date: Tue, 18 Aug 2015 15: 44 : 04 GMT
Server: Apache/2.2.3 (CentOS)
Last-Modified: Tue, 18 Aug 2015 15:11:03 GMT
Content-Length: 6821
Content-Type: text/html

[The object that was requested]
```

It has 3 parts: an initial **status line**, some **header lines** and an **entity body**.



An HTTP response message

> **Note: HTTP response messages don't have the URL or the method fields**. Those are strictly for request messages.

Status Line

- HTTP response status lines start with the **HTTP version**.

Status Code

- The **status code** comes next which tells the client if the request succeeded or failed.
- There are a lot of status codes:
  - 1xx codes fall in the informational category
  - 2xx codes fall in the success category
  - 3xx codes are for redirection
  - 4xx is client error
  - 5xx is server error

Here is a list of some common status codes and their meanings:

- `200 OK`: the request was successful, and the result is appended with the response message.
- `404 File Not Found`: the requested object doesn't exist on the server.

- `400 Bad Request`: generic error code that indicates that the request was in a format that the server could not comprehend.
- `500 HTTP Internal Server Error`: the request could not be completed because the server encountered some unexpected error.
- `505 HTTP Version Not Supported`: the requested HTTP version is not supported by the server.

Have a look at pages 39 and 40 of RFC 2616 for a comprehensive list.

Header Lines

Let's study the header lines.

- **Connection type**. In this case, indicates that the server will `close` the TCP connection after it sends the response.
- **Date**. The date at which the response was generated.
- **Server**. Gives server software specification of the server that generated the message. Apache in this case.
- **Last-Modified.** The date on which the object being sent was last modified.
- **Content-Length.** The length of the object being sent in 8-bit bytes.
- **Content-Type.** The type of content. The type of the file is not determined by the file extension of the object, but by this header.

The **response** body contains the file requested.

How HTTP Headers Are Chosen

Lastly, you must be wondering how browsers decide which HTTP headers to include in requests and how servers decide which headers to return in the response. That **depends on a complex mix of factors such as the browser, the user configurations and products**.

Quick Quiz on HTTP!

1 What is HTTP?

COMPLETED 0%

1 of 5

In the next lesson, we'll look at real HTTP responses via a simple command-line tool!