



HTTP – STATUS CODES

Contents

1. Informational Responses :-	3
1.1 100 (Continue):	3
1.2 101 (Switching Protocols):	3
1.3 103 (Processing) :	3
1.4 103 (Early hints) :	3
2 Successful responses :-	3
2.1 200 (OK) :	3
2.4 201 (Created) :	4
2.5 202 (Accepted) :	4
2.6 203 (Non Authoritative information) :	4
2.7 204 (No content) :	4
2.8 205 (Reset Content) :	4
3 Redirection messages :-	5
3.1 300 (Multiple Choices) :	5
3.2 301 (Moved permanently) :	5
3.3 302 (Found) :	5
3.4 303 (See Other) :	5
3.5 304 (Not modified) :	5
3.6 305 (Use Proxy) :	6
3.7 306 (Unused) :	6
3.8 307 (Temporary Redirect) :	6
3.9 308 (Permanent Redirect) :	6
4 Client error Responses :-	7
4.1 400 (Bad request) :	7
4.2 401 (Unauthorised) :	7
4.3 402 (Payment Required) :	7
4.4 403 (Forbidden) :	7
4.5 404 (Not Found) :	7



4.6	405 (Method not allowed) :	8
4.7	406 (Not acceptable) :	8
4.8	407 (Proxy Authentication required) :	8
4.9	408 (Request Timeout) :	8
4.10	409 (Conflict) :	9
4.11	410 (Gone) :	9
4.12	411 (Length required) :	9
4.13	412 (Pre condition failed) :	9
4.14	413 (Payload too large) :	9
4.15	414 (URI too long) :	10
4.16	415 (Unsupported media type) :	10
4.17	416 (Range not satisfiable) :	10
4.18	417 (Expectation failed) :	10
5	Server error Responses :-	10
5.1	500 (Internal server error) :	10
5.2	501 (Not implemented) :	10
5.3	502 (Bad Gateway) :	11
5.4	503 (Service Unavailable) :	11
5.5	504 (Gateway Timeout) :	11
5.6	505 (HTTP version not supported) :	11
5.7	506 (Variant also negotiates) :	11
5.8	507 (Insufficient Storage) :	12
5.9	508 (Loop detected) :	12
5.10	510 (Not extended) :	12
5.11	511 (Network Authentication required) :	12



1. Informational Responses :-

1.1 100 (Continue):

Only a part of the request has been received by the server, but as long as it has not been rejected, the client should continue with the request.

1.2 101 (Switching Protocols):

The server switches protocol.

1.3 103 (Processing) :

This code indicates that the server has received and is processing the request, but no response is available yet.

1.4 103 (Early hints) :

This status code is primarily intended to be used with the link header, letting the user agent start preloading resources while the server prepares a response or preconnect to an origin from which the page will need resources.

2 Successful responses :-

2.1 200 (OK) :



The request is OK.

2.4 201 (Created) :

The request is complete, and a new resource is created.

2.5 202 (Accepted) :

The request is accepted for processing, but the processing is not complete.

2.6 203 (Non Authoritative information) :

The information in the entity header is from a local or third-party copy, not from the original server.

2.7 204 (No content) :

A status code and a header are given in the response, but there is no entity-body in the reply.

2.8 205 (Reset Content) :

The browser should clear the form used for this transaction for additional input.



3 Redirection messages :-

3.1 300 (Multiple Choices) :

The request has more than one possible response. The user agent or user should choose one of them. (There is no standardized way of choosing one of the responses, but HTML links to the possibilities are recommended so the user can pick.)

3.2 301 (Moved permanently) :

The URL of the requested resource has been changed permanently. The new URL is given in the response.

3.3 302 (Found) :

This response code means that the URI of requested resource has been changed *temporarily*. Further changes in the URI might be made in the future. Therefore, this same URI should be used by the client in future requests.

3.4 303 (See Other) :

The server sent this response to direct the client to get the requested resource at another URI with a GET request.

3.5 304 (Not modified) :

This is used for caching purposes. It tells the client that the response has not been modified, so the client can continue to use the same cached version of the response.



3.6 305 (Use Proxy) :

Defined in a previous version of the HTTP specification to indicate that a requested response must be accessed by a proxy. It has been deprecated due to security concerns regarding in-band configuration of a proxy.

3.7 306 (Unused) :

This response code is no longer used; it is just reserved. It was used in a previous version of the HTTP/1.1 specification.

3.8 307 (Temporary Redirect) :

The server sends this response to direct the client to get the requested resource at another URI with the same method that was used in the prior request. This has the same semantics as the 302 Found HTTP response code, with the exception that the user agent *must not* change the HTTP method used: if a POST was used in the first request, a POST must be used in the second request.

3.9 308 (Permanent Redirect) :

This means that the resource is now permanently located at another URI, specified by the Location: HTTP Response header. This has the same semantics as the 301 Moved Permanently HTTP response code, with the exception that the user agent *must not* change the HTTP method used: if a POST was used in the first request, a POST must be used in the second request.



4 Client error Responses :-

4.1 400 (Bad request) :

The server cannot or will not process the request due to something that is perceived to be a client error (e.g., malformed request syntax, invalid request message framing, or deceptive request routing).

4.2 401 (Unauthorised) :

Although the HTTP standard specifies "unauthorized", semantically this response means "unauthenticated". That is, the client must authenticate itself to get the requested response.

4.3 402 (Payment Required) :

This response code is reserved for future use. The initial aim for creating this code was using it for digital payment systems, however this status code is used very rarely and no standard convention exists.

4.4 403 (Forbidden) :

The client does not have access rights to the content; that is, it is unauthorized, so the server is refusing to give the requested resource. Unlike 401 Unauthorized, the client's identity is known to the server.

4.5 404 (Not Found) :



The server cannot find the requested resource. In the browser, this means the URL is not recognized. In an API, this can also mean that the endpoint is valid but the resource itself does not exist. Servers may also send this response instead of 403 Forbidden to hide the existence of a resource from an unauthorized client. This response code is probably the most well known due to its frequent occurrence on the web.

4.6 405 (Method not allowed) :

The request method is known by the server but is not supported by the target resource. For example, an API may not allow calling DELETE to remove a resource.

4.7 406 (Not acceptable) :

This response is sent when the web server, after performing [server-driven content negotiation](#), doesn't find any content that conforms to the criteria given by the user agent.

4.8 407 (Proxy Authentication required) :

This is similar to 401 Unauthorized but authentication is needed to be done by a proxy.

4.9 408 (Request Timeout) :

This response is sent on an idle connection by some servers, even without any previous request by the client. It means that the server would like to shut down this unused connection. This response is used much more since some browsers, like Chrome, Firefox 27+, or IE9, use HTTP pre-connection mechanisms to speed up surfing. Also note that some servers merely shut down the connection without sending this message.



4.10 409 (Conflict) :

This response is sent when a request conflicts with the current state of the server.

4.11 410 (Gone) :

This response is sent when the requested content has been permanently deleted from server, with no forwarding address. Clients are expected to remove their caches and links to the resource. The HTTP specification intends this status code to be used for "limited-time, promotional services". APIs should not feel compelled to indicate resources that have been deleted with this status code.

4.12 411 (Length required) :

Server rejected the request because the Content-Length header field is not defined and the server requires it.

4.13 412 (Pre condition failed) :

The client has indicated preconditions in its headers which the server does not meet.

4.14 413 (Payload too large) :

Request entity is larger than limits defined by server. The server might close the connection or return an Retry-After header field.



4.15 414 (URI too long) :

The URI requested by the client is longer than the server is willing to interpret.

4.16 415 (Unsupported media type) :

The media format of the requested data is not supported by the server, so the server is rejecting the request.

4.17 416 (Range not satisfiable) :

The range specified by the Range header field in the request cannot be fulfilled. It's possible that the range is outside the size of the target URI's data.

4.18 417 (Expectation failed) :

This response code means the expectation indicated by the Expect request header field cannot be met by the server.

5 Server error Responses :-

5.1 500 (Internal server error) :

The server has encountered a situation it does not know how to handle.

5.2 501 (Not implemented) :



The request method is not supported by the server and cannot be handled. The only methods that servers are required to support (and therefore that must not return this code) are GET and HEAD.

5.3 502 (Bad Gateway) :

This error response means that the server, while working as a gateway to get a response needed to handle the request, got an invalid response.

5.4 503 (Service Unavailable) :

The server is not ready to handle the request. Common causes are a server that is down for maintenance or that is overloaded. Note that together with this response, a user-friendly page explaining the problem should be sent. This response should be used for temporary conditions and the Retry-After HTTP header should, if possible, contain the estimated time before the recovery of the service. The webmaster must also take care about the caching-related headers that are sent along with this response, as these temporary condition responses should usually not be cached.

5.5 504 (Gateway Timeout) :

This error response is given when the server is acting as a gateway and cannot get a response in time.

5.6 505 (HTTP version not supported) :

The HTTP version used in the request is not supported by the server.

5.7 506 (Variant also negotiates) :



The server has an internal configuration error: the chosen variant resource is configured to engage in transparent content negotiation itself, and is therefore not a proper end point in the negotiation process.

5.8 507 (Insufficient Storage) :

The method could not be performed on the resource because the server is unable to store the representation needed to successfully complete the request.

5.9 508 (Loop detected) :

The server detected an infinite loop while processing the request.

5.10 510 (Not extended) :

Further extensions to the request are required for the server to fulfill it.

5.11 511 (Network Authentication required) :

Indicates that the client needs to authenticate to gain network access.