HTTP response status codes indicate whether a specific HTTP request has been successfully completed. Responses are grouped in five classes: informational responses, successful responses, redirects, client errors, and servers errors.

Information responses

`100 Continue`

This interim response indicates that everything so far is OK and that the client should continue with the request or ignore it if it is already finished.

`101 Switching Protocol`

This code is sent in response to an `Upgrade:` request header by the client, and indicates that the protocol the server is switching too. It was introduced to allow migration to an incompatible protocol version, and is not in common use.

Successful responses

`200 OK`

The request has succeeded. The meaning of a success varies depending on the HTTP method:
GET: The resource has been fetched and is transmitted in the message body.
HEAD: The entity headers are in the message body.
POST: The resource describing the result of the action is transmitted in the message body.
TRACE: The message body contains the request message as received by the server

`201 Created`

The request has succeeded and a new resource has been created as a result of it. This is typically the response sent after a PUT request.

`202 Accepted`

The request has been received but not yet acted upon. It is non-committal, meaning that there is no way in HTTP to later send an asynchronous response indicating the outcome of processing the request. It is intended for cases where another process or server handles the request, or for batch processing.

`203 Non-Authoritative Information`

This response code means returned meta-information set is not exact set as available from the origin server, but collected from a local or a third party copy. Except this condition, 200 OK response should be preferred instead of this response.

`204 No Content`

There is no content to send for this request, but the headers may be useful. The user-agent may update its cached headers for this resource with the new ones.

`205 Reset Content`

This response code is sent after accomplishing request to tell user agent reset document view which sent this request.

`206 Partial Content`

This response code is used because of range header sent by the client to separate download into multiple streams.

Redirection messages

`300 Multiple Choice`

The request has more than one possible responses. User-agent or user should choose one of them. There is no standardized way to choose one of the responses.

`301 Moved Permanently`

This response code means that URI of requested resource has been changed. Probably, new URI would be given in the response.

`302 Found`

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

`303 See Other`

Server sent this response to directing client to get requested resource to another URI with an GET request.

`304 Not Modified`

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

`305 Use Proxy`

This means requested response must be accessed by a proxy. This response code is not largely supported because security reasons.

`306 unused`

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

`307 Temporary Redirect`

Server sent this response to directing client to get requested resource to another URI with same method that used prior request. This has the same semantic than 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.

`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.

Client error responses

`400 Bad Request`

This response means that server could not understand the request due to invalid syntax.

`401 Unauthorized`

Authentication is needed to get requested response. This is similar to 403, but in this case, authentication is possible.

`402 Payment Required`

This response code is reserved for future use. Initial aim for creating this code was using it for digital payment systems however this is not used currently.

`403 Forbidden`

Client does not have access rights to the content so server is rejecting to give proper response.

`404 Not Found`

Server can not find requested resource. This response code probably is most famous one due to its frequency to occur in web.

`405 Method Not Allowed`

The request method is known by the server but has been disabled and cannot be used. The two mandatory methods, `GET` and `HEAD`, must never be disabled and should not return this error code.

`406 Not Acceptable`

This response is sent when the web server, after performing server-driven content negotiation, doesn't find any content following the criteria given by the user agent.

`407 Proxy Authentication Required`

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

`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 or IE9, use HTTP preconnection mechanisms to speed up surfing (see bug 881804, which tracks the future implementation of such a mechanism in Firefox). Also note that some servers merely shut down the connection without sending this message.

`409 Conflict`

This response would be sent when a request conflict with current state of server.

`410 Gone`

This response would be sent when requested content has been deleted from server.

`411 Length Required`

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

`412 Precondition Failed`

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

`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.

`414 URI Too Long`

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

`415 Unsupported Media Type`

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

`416 Requested Range Not Satisfiable`

The range specified by the `Range` header field in the request can't be fulfilled; it's possible that the range is outside the size of the target URI's data.

`417 Expectation Failed`

This response code means the expectation indicated by the `Expect` request header field can't be met by the server.

`421 Misdirected Request`

The request was directed at a server that is not able to produce a response. This can be sent by a server that is not configured to produce responses for the combination of scheme and authority that are included in the request URI.

426 Upgrade Required

The server refuses to perform the request using the current protocol but might be willing to do so after the client upgrades to a different protocol. The server MUST send an Upgrade header field in a 426 response to indicate the required protocol(s) (Section 6.7 of [RFC7230]).

428 Precondition Required

The origin server requires the request to be conditional. Intended to prevent "the 'lost update' problem, where a client GETs a resource's state, modifies it, and PUTs it back to the server, when meanwhile a third party has modified the state on the server, leading to a conflict.

429 Too Many Requests

The user has sent too many requests in a given amount of time ("rate limiting").

431 Request Header Fields Too Large

The server is unwilling to process the request because its header fields are too large. The request MAY be resubmitted after reducing the size of the request header fields.

451 Unavailable For Legal Reasons

The user requests an illegal resource, such as a web page censored by a government.

Server error responses

500 Internal Server Error

The server has encountered a situation it doesn't know how to handle.

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`.

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.

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 responses 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.

504 Gateway Timeout

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

505 HTTP Version Not Supported

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

506 Variant Also Negotiates

The server has an internal configuration error: transparent content negotiation for the request results in a circular reference.

507 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.

511 Network Authentication Required

The 511 status code indicates that the client needs to authenticate to gain network access.