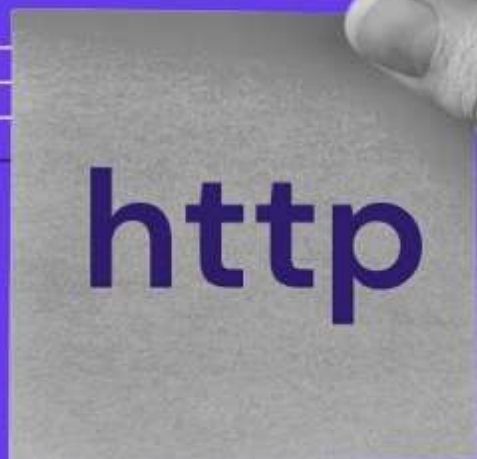


HTTP

Status Codes Cheat Sheet



1XX
2XX
3XX



HTTP Status Codes Cheat Sheet

HTTP status codes are organized into five categories.

1XX Informational:

This class is invisible to the client and indicates that the server received the request and will continue it. It consists of four status codes:

100 (continue):

The server has received the initial request and will send the final answer once the client sends the entire submission.

101 (switching protocols):

The server understood the Upgrade header request and will point out which application protocol to use if it's advantageous.

102 (Processing):

The server sends this status when the request takes longer to complete. It aims to reassure the client that it's processing the submission.

103 (Early Hints):

It allows the user to preload resources while the server prepares for a response.

2XX Success :

This class means the server has completed the request and sent back the expected response to the client. It consists of ten status codes.

200 (ok):

The server has processed the request. However, the actual meaning depends on the method used, such as HEAD, GET, POST, OPTIONS, TRACE, or PUT. It is an ideal status code, showing your page is working properly for visitors and search engine crawlers.

201 (created):

The server has completed the request by creating one or more new resources shown in the Location header or Uniform Resource Identifier (URI).

202 (accepted):

The server has accepted the request but is not finished yet. This response allows a host to process another submission without the user staying connected until the end.

203 (non-authoritative information):

The request was successful, but a transforming proxy modified the meta information. Hence, the server collects the actual data from a third-party or local copy.

204 (No content):

The server has fulfilled the request, but there is no additional content to show. This response allows action inputs without changing the current page. 205 reset content. The server has completed the request and the user should reset the document view to its original state.

206 (Partial Content):

The client sends a Range header request asking the server to load only part of a resource. This response includes a Content-Range header or multipart Content-Type, ETag and/or Content-Location, and Expires.

207 (Multi-Status):

It provides the statuses of multiple independent operations. 208 already reported. It is used inside a DAV:propstat element to avoid repeatedly enumerating the internal members of multiple bindings to the same collection.

226 IM used:

The server has fulfilled a GET request, and the response shows one or more instance-manipulation results.

3XX Redirection

This class means the user gets redirected as the resource resides in a new location. It consists of nine status codes:

300 multiple choices:

The request has more than one possible response, and the user agent should choose one of them.

301 moved permanently:

The target resource has been moved to a new URL permanently. Regarding SEO, search engine bots will pass link equity to the new address.

302 found (previously: moved temporarily):

The requested resource has been moved to a new URI temporarily.

303 see other:

The server will direct the client to a different resource at another URI with the GET method.

304 not modified:

It informs the client that the response has not been modified. Hence, the user can still use the resource's cached version.

305 unused (previously: use proxy):

It informed the client to use a proxy to access the requested resource. However, it was deprecated due to security concerns.

306 unused (previously: switch proxy):

It informed the client that subsequent requests should use the specified proxy. However, it is no longer used.

307 temporary redirect:

The server will use the previous method to direct the client to the requested resource on another URI.

308 permanent redirect:

The resource has been moved to another URI permanently, and any future references to it should use the address enclosed in the response

4XX Client Error

This class indicates that the server can't reach the page due to a client-side error. Here are the most common status codes:

400 bad requests:

The server couldn't process the request due to a client error, such as bad syntax or deceptive routing.

401 unauthorized:

The server couldn't fulfill the request as it requires valid authentication credentials.

402 payment required:

Initially, it would be part of digital payment systems. However, the plan failed, and it's currently reserved for future use.

403 forbidden:

The server understood the request but refused to fulfill it because the client didn't have enough permissions to access the resource.

404 not found:

The server couldn't find the requested resource or was unwilling to disclose it. This may harm your SEO efforts as search engines won't index the content.

405 method not allowed:

The server supports the request method, but the resource doesn't.

406 not acceptable:

The server couldn't find any content following the criteria given by the user via the Accept header.

407 proxy authentication required:

A proxy must authenticate the client before continuing.

408 request timeout:

The server got timed out while waiting for the entire request.

409 conflict:

The server can't fulfill the request due to a conflict with the target resource.

410 gone. The server has deleted the requested resource permanently.

411 length required:

The server rejected the request because it requires a defined Content-Length.

412 precondition failed:

The server doesn't meet the expectations stated in the request header field.

413 payload too large:

The payload was too large and the server declined to process it.

414 URI too long:

The request-URI was too long and the server refused to interpret it.

415 unsupported media type:

The request uses a media type that the server doesn't support.

416 range not satisfiable:

The server can't fulfill the range specified in the Range header field.

417 expectation failed:

The server couldn't meet the requirements in the Expect request-header field.

418 unused (previously: I'm a teapot):

The server refuses to brew coffee using a teapot. This response is an April Fools joke from 1998.

421 misdirected request:

The client sends a request to a server that can't produce a response.

422 unprocessable entity:

The server couldn't process the request since it contained semantic errors.

423 locked:

The resource is locked.

424 failed dependency:

The request failed because the previous action also failed.

425 too early:

The server refused to process a request that might be replayed.

426 upgrade required:

The server refuses to process the request unless the client switches to a different protocol.

428 precondition required:

The server requires the request to be conditional.

429 too many requests:

The client has sent too many requests at a time.

431 request header fields too large:

The server couldn't process the request because the header fields were too large.

451 unavailable for legal reasons:

The client has requested a resource that the server can't legally provide.

5XX Server Error

This class means the server encountered an error while processing the request. When it comes to SEO, search engines will lower the site's crawl budget and remove indexed URLs that persistently return this response. Here are some of the status codes:

500 internal server error:

An unexpected error occurred, making the server fail to complete the request.

501 not implemented:

The server doesn't support the functionality required to fulfill the request.

502 bad gateway:

The server got an invalid response from an inbound host while trying to complete the request.

503 service unavailable:

The server is unavailable and can't process the request. This may happen because it is down or under scheduled maintenance.

504 gateway timeout:

The server got timed out while waiting for a response.

505 HTTP version not supported:

The server doesn't support the HTTP version used in the request.

506 variant also negotiates:

The server encountered an internal configuration error.

507 insufficient storage:

The request method can't work on the resource because the server is unable to manage the necessary representation to fulfill the request.

508 loop detected:

The server detected an infinite loop while processing the request.

510 unused (previously: not extended):

The server requires further extensions to fulfill the request. However, this experimental code is no longer used since the Internet Engineering Task Force (IETF) terminated it.

511 network authentication required:

Clients must authenticate themselves before getting network access.

Did you find it Useful?

Leave a **comment!**



Alamin CodePapa

@CodePapa360

FOLLOW FOR MORE

Like



Comment



Repost

