HTTP Status Codes

For great REST services the correct usage of the correct HTTP status code in a response is vital.

1xx – Informational

This class of status code indicates a provisional response, consisting only of the Status-Line and optional headers, and is terminated by an empty line

100 – Continue
101 – Switching Protocols
102 – Processing

2xx – Successful

This class of status code indicates that the client's request was successfully received, understood, and accepted.

200 – OK
201 – Created
202 – Accepted
203 – Non-Authoritative Information
204 – No Content
205 – Reset Content
206 – Partial Content
207 – Multi-Status

3xx – Redirection

This class of status code indicates that further action needs to be taken by the user agent in order to fulfill the request.

300 – Multiple Choices
301 – Moved Permanently
302 – Found
303 – See Other
304 – Not Modified
305 – Use Proxy
307 – Temporary Redirect

4xx – Client Error

The 4xx class of status code is intended for cases in which the client seems to have erred.

400 – Bad Request
401 – Unauthorised
402 – Payment Required
403 – Forbidden
404 – Not Found
405 – Method Not Allowed
406 – Not Acceptable
407 – Proxy Authentication Required
408 – Request Timeout
409 – Conflict
410 – Gone
411 – Length Required
412 – Precondition Failed
413 – Request Entity Too Large
414 – Request URI Too Long
415 – Unsupported Media Type
416 – Requested Range Not Satisfiable
417 – Expectation Failed
422 – Unprocessable Entity
423 – Locked
424 – Failed Dependency
425 – Unordered Collection
426 – Upgrade Required

5xx – Server Error

Response status codes beginning with the digit "5" indicate cases in which the server is aware that it has erred or is incapable of performing the request.

500 – Internal Server Error
501 – Not Implemented
502 – Bad Gateway
503 – Service Unavailable
504 – Gateway Timeout
505 – HTTP Version Not Supported
506 – Variant Also Negotiates
507 – Insufficient Storage
510 – Not Extended

Examples of using HTTP Status Codes in REST

201 – When doing a POST to create a new resource it is best to return 201 and not 200.
204 – When deleting a resources it is best to return 204, which indicates it succeeded but there is no body to return.
301 – If a 301 is returned the client should update any cached URI's to point to the new URI.
302 – This is often used for temporary redirect's, however 303 and 307 are better choices.
409 – This provides a great way to deal with conflicts caused by multiple updates.
501 – This implies that the feature will be implemented in the future.

Special Cases

306 – This status code is no longer used. It used to be for switch proxy.
418 – This status code from RFC 2324. However RFC 2324 was submitted as an April Fools' Joke. The message is *I am a teapot.*

Key	Description
Black	HTTP version 1.0
Blue	HTTP version 1.1
Aqua	Extension RFC 2295
Green	Extension RFC 2518
Yellow	Extension RFC 2774
Orange	Extension RFC 2817
Purple	Extension RFC 3648
Red	Extension RFC 4918

REST is a style of development for exposing data in such a way that it is easy to consume, easy to produce and makes use of HTTP. REST is not a replacement for SOAP or WS* but an alternative to solve certain problems better.

Architecture