

HTTP Status Codes

Bob Singh

Web servers return HTTP status codes, to indicate the status of a request. When a Web server responds to a request from a browser, the response consists of a status line, response headers, a blank line and the document. For example,

```
HTTP/1.1 200 OK
Content-Type: text/plain
```

```
Hello World
```

In the above example, the status line consists of the HTTP version (1.1), status code (200 in the above example) and a short message (OK in the above example). Headers are optional except for Content-Type, which specifies the MIME type of the document. The message in the status line is directly associated with the status code. The status code should be set before sending any document content to the client. You can use setStatus method of HttpServletResponse for doing this. The setStatus method takes an int (the status code) as an argument. So, you can call this method by

```
response.setStatus(int.....);
```

HTTP status codes fall into five categories:

- 100-199
Codes in the 100's are informational, indicating that the client should respond with some other action.
- 200-299
Indicate that the request was successful.
- 300-399
Used for files that have moved and usually indicate a Location header indicating the new address.
- 400-499
Indicate an error by the client.
- 500-599
Signify an error by the server.

The constants in HttpServletResponse that represent the various codes are derived from the standard messages associated with the codes. In servlets, you refer to the

status codes by means of these constants. For example, you would say, `response.setStatus(response.SC_NO_CONTENT)` rather than `response.setStatus(204)`.

Status Code	Meaning
100 (Continue)	The 100(SC_CONTINUE) status code tells the browser to continue sending a request to the server.
101 (Switching Protocols)	101(SC_SWITCHING_PROTOCOLS) status indicates that the server will comply with the Upgrade header. The server sends this response, when the client asks to switch from HTTP/1.0 to HTTP/1.1.
200 (OK)	A value of 200(SC_OK) means that everything is successful. The document follows for GET and POST requests. This status is default for servlets.
201 (Created)	A status code of 201(SC_CREATED) indicates that the server created a new document in response to the request; the Location header will give its URL.
202 (Accepted)	202(SC_ACCEPTED) tells the client that the request is being acted upon, but processing is not yet complete.
203 (Non-Authoritative Information)	203(SC_NON_AUTHORITATIVE_INFORMATION) status signifies that the document is being returned normally, but some of the response headers might be incorrect.
204 (No Content)	204(SC_NO_CONTENT) tells that the browser should continue to display the previous document because no new document is available.
205 (Reset Content)	205(SC_RESET_CONTENT) means that there is no new document, but the browser should reset the document view. It is used to force browsers to clear form fields.
206 (Partial Content)	206(SC_PARTIAL_CONTENT) is sent, when the server fulfills a partial request that includes a Range header.
300 (Multiple Choices)	300(SC_MULTIPLE_CHOICES) tells that the requested document can be found in several places. These locations are listed in the returned document
301 (Moved Permanently)	301(SC_MOVED_PERMANENTLY) status indicates that the requested document is elsewhere; the new URL for this document is given in the Location header. Browsers should automatically follow the link to the new URL.
302 (Found)	302(SC_FOUND) is similar to 301 except that the URL given by the Location header should be interpreted as a temporary replacement, not a permanent one.
303 (See Other)	303(SC_SEE_OTHER) status is similar to 301, 302 except

	that if the original request was POST, the new document should be retrieved with a GET.
304 (Not Modified)	When a client has a cached document, it can perform a conditional request by supplying an If-Modified-Since header to indicate that it only wants the document, if it has been changed since the specified date. 304 (SC_NOT_MODIFIED) means that the cached version is up-to-date and the client should use it. Otherwise, the server should return the requested document with the normal 200 status code.
305 (Use Proxy)	305(SC_USE_PROXY) signifies that the requested document should be retrieved via the proxy listed in the Location header.
307 (Temporary Redirect)	The rules for handling 307 status code is similar to 302. It redirects GET and POST requests in 303 responses; redirects GET, but not POST requests in 307 responses.
400 (Bad Request)	400(SC_BAD_REQUEST) status indicates bad syntax in the client request
401 (Unauthorized)	401(SC_UNAUTHORIZED) signifies that the client tried to access a password-protected page without proper identifying information in the Authorization header.
403 (Forbidden)	403(SC_FORBIDDEN) means that the server refuses to supply the resource, regardless of authorization. It is a result of a bad file or directory permissions on the server.
404 (Not found)	404(SC_NOT_FOUND) status tells the client that no resource could be found at that address.
405 (Method Not Allowed)	405(SC_METHOD_NOT_ALLOWED) value indicates that the request method (GET, POST, PUT, DELETE etc.) was not allowed for this particular resource.
406 (Not Acceptable)	406(SC_NOT_ACCEPTABLE) indicates that the requested resource has a MIME type incompatible with the types specified by the client in Accept header.
407 (Proxy Authentication Required)	407(SC_PROXY_AUTHENTICATION_REQUIRED) value is similar to 401, but it is used by proxy servers. It indicates that the client must authenticate itself with the proxy server.
408 (Request Timeout)	408(SC_REQUEST_TIMEOUT) code means that the client took too long to finish sending the request.
409 (Conflict)	409(SC_CONFLICT) status code is used for situations such as an attempt to upload an incorrect version of a file.
410 (Gone)	410(SC_GONE) tells the client that the requested document is gone and no forwarding address is known. It differs from 404 in the sense that the

	document is permanently gone (not just unavailable for unknown reasons as with 404).
411 (Length required)	411(SC_LENGTH_REQUIRED) signifies that the server cannot process the request, unless the client sends a Content-Length header (indicating the amount of data being sent to the server).
412 (Precondition Failed)	412(SC_PRECONDITION_FAILED) status indicates that some precondition specified in the header was false.
413 (Request Entity Too Large)	413(SC_REQUEST_ENTITY_TOO_LARGE) status is used, when the URI is too long. URI means the part of the URL, after the host and the port in the URL.
415 (Unsupported Media Type)	415(SC_UNSUPPORTED_MEDIA_TYPE) means that the request had an attached document of a type that the server doesn't know how to handle.
416 (Requested range not satisfiable)	416(SC_REQUESTED_RANGE_NOT_SATISFIABLE) means that the client included an unsatisfiable Range header in the request.
417 (Expectation Failed)	416(SC_EXPECTATION_FAILED) tells the browser that it won't accept the document.
500 (Internal Server Error)	500 (SC_INTERNAL_SERVER_ERROR) is the generic "Server is confused" status code. This results from servlets that crash or return improperly formatted headers or CGI programs.
501 (Not Implemented)	501(SC_NOT_IMPLEMENTED) status notifies the client that the server doesn't support the functionality to fulfill the request.
502 (Bad Gateway)	502(SC_BAD_GATEWAY) is used by servers that act as proxies or gateways. It indicates that the initial server got a bad response from the remote server.
503 (Service Unavailable)	503(SC_SERVICE_UNAVAILABLE) signifies that the server cannot respond because of maintenance or overloading.
504 (Gateway Timeout)	504(SC_GATEWAY_TIMEOUT) is used by servers that act as proxies or gateways. It indicates that the initial server didn't get a timely response from the remote server.
505 (HTTP Version Not Supported)	505(SC_HTTP_VERSION_NOT_SUPPORTED) status code means that the server doesn't support the HTTP version named in the request line.