HTTP METHODS

The primary or most commonly-used HTTP methods are POST, GET, PUT, PATCH, and DELETE. These methods correspond to create, read, update, and delete (or CRUD) operations, respectively. There are a number of other methods, too, but they are utilized less frequently.

GET

The HTTP GET method is used to *read* (or retrieve) a representation of a resource. In case of success (or non-error), GET returns a representation in JSON and an HTTP response status code of 200 (OK). In an error case, it most often returns a 404 (NOT FOUND) or 400 (BAD REQUEST).

The GET method is used to retrieve information from the given server using a given URI. Requests using GET should only retrieve data and should have no other effect on the data.

POST

The POST method is most often utilized to *create* new resources. In particular, it is used to create subordinate resources. That is subordinate to some other (e.g. parent) resource. In other words, when creating a new resource, POST to the parent and the service takes care of associating the new resource with the parent, assigning an ID (new resource URI), etc.

On successful creation, HTTP response code 201 is returned.

A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms.

PATCH

PATCH is used to *modify* resources. The PATCH request only needs to contain the changes to the resource, not the complete resource.

In other words, the body should contain a set of instructions describing how a resource currently residing on the server should be modified to produce a new version.

DELETE

DELETE is quite easy to understand. It is used to *delete* a resource identified by filters or ID.

On successful deletion, the HTTP response status code 204 (No Content) returns with no response body.

Important

If you DELETE a resource, it is removed. Repeatedly calling DELETE on that resource will often return a 404 (NOT FOUND) status code since it was already removed and, therefore, is no longer findable.

Removes all current representations of the target resource given by a URI.

HEAD
Same as GET, but transfers the status line and header section only.

PUT
Replaces all current representations of the target resource with the uploaded content.

CONNECT
Establishes a tunnel to the server identified by a given URI.

OPTIONS
Describes the communication options for the target resource.

TRACE
Performs a message loop-back test along the path to the target resource.