KNOW YOUR HTTP METHODS!

HTTP methods, also called verbs, are used to specify which *action* to apply to the *resource* specified by the Uniform Resource Identifier (URI). What exactly this resource represents, or the exact action performed on the resource, depends on the application server.

Method	Description	Safe?	Idempotent?	Specification
GET	Request a representation of the resource	yes	yes	HTTP 1.0
HEAD	Request only the headers for the resource	yes	yes	HTTP 1.0
POST	Process the request body with the resource	no	no	HTTP 1.0
PUT	Create or update a new resource with the contents of the request body	no	yes	HTTP 1.1
DELETE	Remove the specified resource	no	yes	HTTP 1.1
OPTIONS	Return the HTTP methods the specified resource supports	yes	yes	HTTP 1.1
TRACE	Echo the received request	yes	yes	HTTP 1.1
CONNECT	Convert the connection to a transparent tcp/ip tunnel, usually to allow SSL/TLS through an unencrypted HTTP proxy	—	—	HTTP 1.1
PATCH	Apply partial modifications to the resource	no	yes	RFC-5789

Safe Methods

Certain methods are specified to be *safe*, which means that executing them will not modify the resource or have other side effects on the overall state of the server. Unsafe methods may cause side effects, such as modifying a resource, sending an email or initiating the processing of a credit card.

Idempotent Methods

Some methods are *idempotent*, meaning that executing identical requests multiple times will have the same effect as executing only one request. For example, DELETE requests are indempotent because once a resource is deleted it can't be deleted again. Conversely, POST requests are not idempotent because a second POST request may send a second email or process the same credit card a second time. By definition, *safe* methods are also *idempotent*.