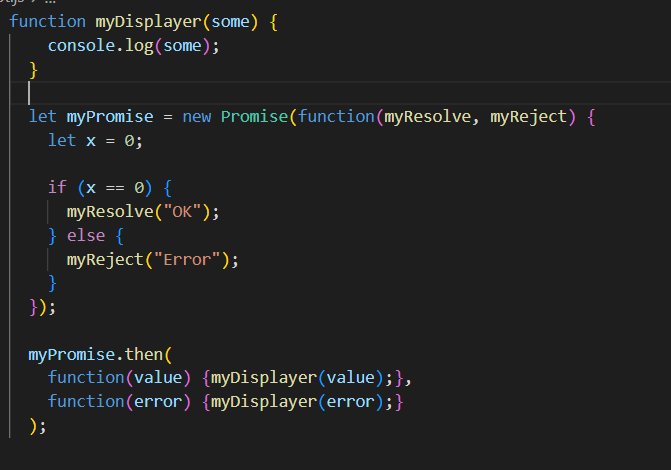
**JavaScript Promises**

A promise is an object representing eventual completion of an asynchronous event.

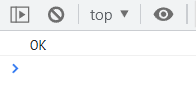
A promise is i one of the following states

1. Pending - means initial state, neither fulfilled or rejected.
2. fulfilled: meaning that the operation was completed successfully.
3. rejected: meaning that the operation failed.

Input:



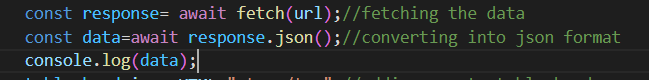
Output:



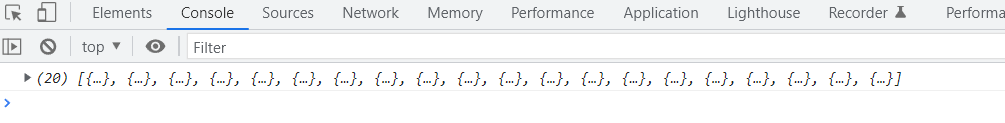
**JavaScript Fetch**

The fetch() method starts the process of fetching a resource from a server. The fetch() method returns a Promise that resolves to a Response object.

Input

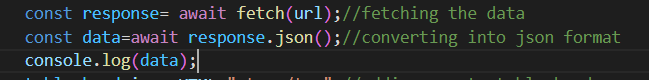


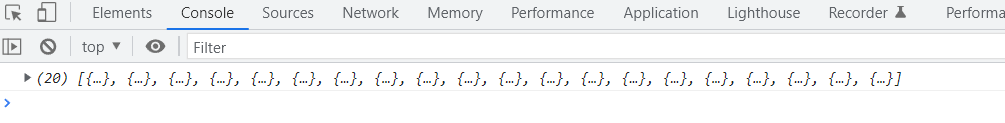
Output



HTTP methods

1. GET: The get method requests a representation of the specified resource. Requests using GET should only retrieve data.





1. Post :

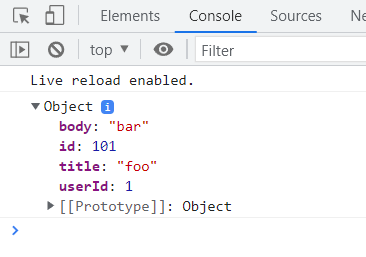
POST is used to send data to a server to create/update a resource.

The data sent to the server with POST is stored in the request body of the HTTP request.

Input:



Output:



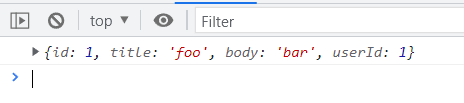
1. put

PUT request is made to a particular resource. If the Request-URI refers to an already existing resource, an update operation will happen, otherwise create operation should happen if Request-URI is a valid resource URI (assuming the POST-request-URIclient is allowed to determine resource identifier).

input:



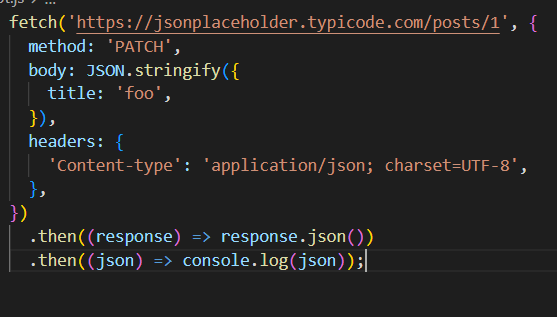
Output:



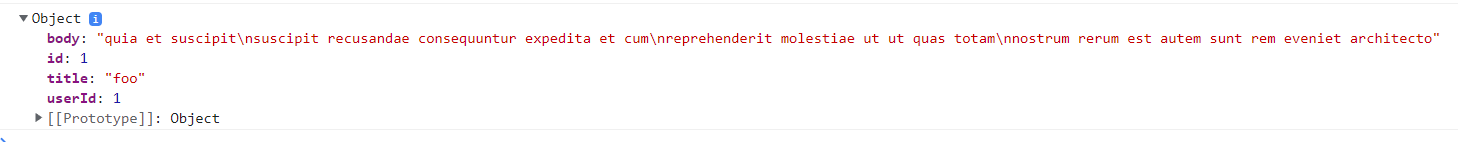
1. post

A PATCH request is considered a set of instructions on how to modify a resource. Contrast this with put; which is a complete representation of a resource.

Input

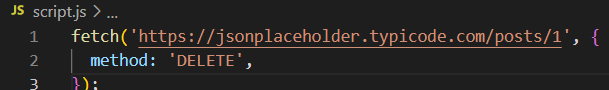


Output



1. Delete

Used to delete a specified resource from the server.



**HTTP status codes**

1. 2xx- these codes represent success.

1. 200-OK -the request was fulfilled.
2. 201- created - following a POST command this represents success.
3. 204- NO Response- Server has received the request but there is no information to send back, and the client should stay in the same document view. This is mainly to allow input for scripts without changing the document at the same time.

2. 3xx- redirect

1. 301-Moved permanently- The data requested has been assigned a new URI, the change is permanent.
2. 302-found temporarily - It is the same as the 301, indicating that the client should redirect. Instead, you're declaring that it is a temporary redirect and that they should come back to looking up this URL at some point.

3. 4xx- client error

1. 400-Bad request- The request had bad syntax,such as invalid input or missing data or was inherently impossible to be satisfied.
2. 401-unauthorized- 401 response is sent when the client has not sent any authentication credentials.
3. 403-Forbidden- when they have credentials but they are wrong or insufficient.
4. 404-Not Found- when the server cannot find anything matching the url given.

4. 5xx-Server error

1. 500-internal server error- This is the code that the error handler sends when there is an unintentional error in your code. You can be very sure it is to do with a server-side coding error or config when throwing this error.
2. 503-gateway timeout-This is equivalent to Internal Error 500, but in the case of a server which is in turn accessing some other service, this indicates that the respose from the other service did not return within a time that the gateway was prepared to wait.