COURSEWARE

HTTP-Requests

Contents

Overview

A JavaScript **HTTP request** is a server side object that represents a *request* to the server.

Tutorial

Request Methods

HTTP defines a set of **request methods** to indicate the desired action to be performed for a given resource. Each of them implement a different semantic, but common features are shared by a group of them (e.g. a request method can be *safe*, *idempotent*, or *cacheable*).

GET

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

HEAD

The `HEAD` method asks for a response identical to that of a `GET` request, but without the response body.

POST

The `POST` method is used to submit an entity to the specified resource, often causing a change in state or side effects on the server.

PUT

The `PUT` method replaces all current representations of the target resource with the request payload.

DELETE

The `DELETE` method deletes the specified resource.

CONNECT

The `CONNECT` method establishes a tunnel to the server identified by the target resource.

OPTIONS

The `OPTIONS` method is used to describe the communication options for the target resource.

TRACE

The `TRACE` method performs a message loop-back test along the path to the target resource.

PATCH

The `PATCH` method is used to apply partial modifications to a resource.

HTTP Status Codes

HTTP Response status codes indicate whether a specific HTTP Request has been successfully completed. Responses are grouped in 5 classes:

- *Informational responses* (100 - 199)
- *Successful responses* (200 - 299)
- *Redirects* (300 - 399)
- *Client Errors* (400 - 499)
- *Server Errors* (500 - 599)

Informational Responses

100 - Continue

This interim response indicates that everything is OK so far, and that the client should continue the request, or ignore the response if the request is finished.

Successful responses

200 - OK

This shows that the request was successfully carried out. All the requested data was located on the web server and transferred to the client. Internet users do not usually see this code.

Redirects

301 - Moved Permanently

301 means that the data requested from the client cannot be found under the given address since it has been moved permanently.

Since the current location of the request is delivered in the status report, the browser can request the new address straight away.

The user is then forwarded to the new address and the old address is no longer valid. The 301 code also goes unnoticed because the URL in the address bar simply changes.

302 - Moved Temporarily

Unlike the 301 code, which is a permanent redirection, the 302 informs the user that the requested data has temporarily been moved.

With a 302 code, the remaining information is specified so that an automatic redirection takes place. The old address remains valid.

Client Errors

403 - Forbidden

403 tells the client that the requested data is access-protected and that the request cannot be performed due to the client not having the authority.

An automatically generated HTML page will let the user know about the access problem.

404 - Not found

404 means that the requested website information was not found on the server. It could be that the address no longer exists or the contents were moved to a new address without notice.

Users that receive a 404 message should check whether the address was correctly written in the address bar. Any links to non-existing pages are known as *dead links*.

Server Errors

500 - Internal Server Error

The 500 server response functions as a collection status code for unexpected server errors. If an error occurs on the server's part, which prevents the requested data from being retrieved, this HTTP status code will automatically be issued.

This should be analysed by the website owner so that repairs can be carried out on the server software.

For further information on each of the status codes, click [this link if you are a dog person](#) and [this link if you are a cat person](#).

Exercises

There are no exercises for this module.