LU05a - Hypertext Transfer Protocol

Das **H**yper**t**ext **T**ransfer **P**rotocol (kurz HTTP) wird hauptsächlich eingesetzt, um Webseiten aus dem World Wide Web zu einem Webbrowser zu übertragen. Dabei sendet der Client (Webbrowser) eine Anfrage (engl. Request) an den Webserver, dieser sendet eine Antwort (engl. Response) zurück.

Lesen Sie https://de.wikipedia.org/wiki/Hypertext_Transfer_Protocol und selfhtml - HTTP um einen Überblick über HTTP zu erhalten.

HTTP-Methoden

Die Angabe einer HTTP-Methode bestimmt den Aufbau des Requests. Ausserdem zeigt sie dem Server an, welche Art von Verarbeitung der Client wünscht.

Die wichtigsten Methoden für den Einstieg sind GET, POST (und PUT).

GET

Die Methode GET zeigt an, dass der Client Daten lesen möchte. Er erwartet vom Server eine Datei als Antwort. Ein Request mit der Methode GET enthält nur Anfrage- und Header-Zeilen, keinen Body.

Mit dieser Methode lesen wir z.B. HTML-Dateien oder Bilder.

POST

Um Daten an den Server zu senden, verwendet der Client die POST-Methode. Der Client erwartet, dass der Server die mitgesendeten Daten verarbeitet, zum Beispiel in eine Datenbank speichert. Nebst Anfrage- und Headerzeilen enthält ein POST-Request einen Body mit den Daten.

Die Benutzereingaben in Formularen werden mit POST übermittelt.

PUT-Methode

In manchen Webapplikationen wird neben POST auch die PUT-Methode verwendet. Sowohl POST als auch PUT senden Daten an den Client zur Verarbeitung. Der Unterschied ist, ob es sich um ein neues Datenobjekt oder ein bestehendes Datenobjekt handelt. Zur Verdeutlichung nehmen wir eine Webseite um Bücher zu erfassen bzw. zu bearbeiten.

- Der Benutzer erfasst ein neues Buch ⇒ POST. Der Server soll ein neues Buch mit den übermittelten Daten einfügen.

- Der Benutzer ändert die Angaben zu einem bestehenden Buch ⇒ PUT.

In vielen Webseiten wird ein Formular immer mit POST übermittelt. Dann muss der Server selber herausfinden, ob es sich um ein neues oder bestehendes Datenobjekt handelt.

HTTP Statuscode

Die wichtigste Angabe in der Response des Servers ist der HTTP Statuscode. Dieser Statuscode informiert den Client ob die Anfrage erfolgreich bearbeitet wurde. Die vollständige Liste finden Sie unter https://developer.mozilla.org/en-US/docs/Web/HTTP/Status.

Hier sind nur ein paar der wichtigsten Statuscodes:

Status	Bedeutung	Erklärung
200	OK	Die Anfrage wurde erfolgreich verarbeitet.
302	Moved Permanently	Die Zieladresse hat sich verändert, in Zukunft die neue Adresse verwenden.
304	Not modified	Die Ressource hat sich seit dem letzten Zugriff nicht verändert. Der Client kann die Version aus seinem Cache verwenden.
404	Not found	Die gewünschte Ressource wurde nicht gefunden.
500	Internal Server Error	Auf dem Server ist ein Fehler aufgetreten.

---

m293-LU05, m293-XnY

Kevin Maurizi, Marcel Suter

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/modul/m293/learningunits/lu05/http

Last update: **2024/05/21 07:03**