LU08b - Variablen einsetzen

Variablen in Bruno erleichtern das Anpassen von Requests an unterschiedliche Bedingungen.

Variablen für Pfade

Im Verlauf der Entwicklung wird eine Applikation zunächst in der Entwicklungsumgebung ausgeführt. Für die Requests in Bruno bedeutet dies in der Regel, dass unsere API unter einem Pfad wie `http://localhost:5000/…` erreichbar ist. Später werden wir die Applikation auf unser Testsystem deployen. Der Pfad im Testsystem könnte z.B. `https://test.bzz.ch/…` sein.

Ich möchte meine API-Tests in beiden Umgebungen nutzen können. Welche Möglichkeiten habe ich?

- Ich passe jedesmal den Pfad in allen Requests an.
- Ich erstelle Kopien aller Requests – einmal mit dem Pfad des Testsystems und einmal mit dem Pfad der Entwicklungsumgebung.
- Ich verwende eine Umgebungsvariable und passe jeweils nur den Wert dieser Variablen an.

Um eine solche Umgebungsvariable einzusetzen, benötigen wir nur wenige Schritte:

1. Definiere die Variable in einer Environment-Datei (z. B. `dev.json` oder `test.json`).
2. Lege den Wert fest (z. B. `http://localhost:5000` oder `https://test.bzz.ch`).
3. Füge die Variable in den Pfad deiner Requests ein, z. B. `{{baseUrl}}/book/list`.

Siehe Bruno Docs: Environment Variables.

Variablen für Daten

Mit Variablen können wir auch Werte aus einer Response speichern und in einem anderen Request wiederverwenden.

Beispiele

Erstellen und Lesen von Daten

1. Ein Request speichert einen neuen Kunden. Als Antwort liefert die API die Kundennummer.
2. Ich speichere die Kundennummer in einer Variablen.
3. Ein weiterer Request verwendet diese Variable, um den vorher erstellten Kunden zu lesen.

Authentifikation

Bei vielen APIs brauchen wir ein Token, welches unsere Identität und Berechtigungen enthält. Dieses Token erhalten wir, indem wir einen Login-Request senden.

1. Ein Request sendet die Login-Daten an die API. Als Antwort erhalte ich ein Authentifizierungstoken.
2. Ich speichere das Token in einer Variablen.
3. Bei jedem weiteren Request sende ich das gespeicherte Token mit.

Umsetzung

Ich zeige hier ein Beispiel, wie ich das mit Bruno umsetze.

1. Login student

Der Request sendet die Login-Daten für einen Studenten. Die Response ist eine JSON-Struktur, die das Token enthält.

Script im Bereich „Tests“

```
test("Status test", () => {
  // HTTP-Status prüfen
  expect(res.status).to.equal(200);

  // JSON-Daten aus der Response holen
  const data = res.json();

  // Token im Environment speichern
  bru.setEnvVar("token", data.token, { persist: true });
});
```

2. Lies alle Anmeldungen zu einem Studenten

Dieser Request liest alle Anmeldungen zum angemeldeten Studenten. Dazu muss das Token aus dem ersten Request mitgesendet werden. Dieses Token wird im Header des Requests mitgesendet und nicht in der URL.

Token in „headers“

- Key: `Authorization`
- Value: `Bearer {{token}}`

Hinweise & Doku

1. Tests/Assertions in Bruno: Bruno Docs: Tests
2. JavaScript-API Referenz (z. B. bru.setEnvVar): Bruno Docs: JavaScript API Reference

---

M450-LU08

Marcel Suter (angepasst)

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

Permanent link:
https://wiki.bzz.ch/modul/m450/learningunits/lu08/variablenbruno?rev=1758181011

Last update: **2025/09/18 09:36**