## LU08.A01 - Resultate auswerten



Ergänze die Requests zum Testen der Bookshelf-Applikation in **Bruno** mit Tests.

Um die Ausführung und Auswertung der Requests zu automatisieren, ergänzen wir diese durch **Tests**. Jeder Request soll **1-2 Tests** erhalten:

- Prüfen, ob die Response den erwarteten HTTP-Statuscode enthält.
- Wo sinnvoll: Prüfen, ob die **Daten in der Response** korrekt sind.



Der Request zum **Lesen aller Bücher** erwartet als Antwort ein **Array mit 3 Büchern**. Sieh in den Bruno-Docs nach, wie man JSON-Responses prüft (Status, Typ, Länge, Felder).

### **Hinweise**

Beispiele für Tests (in der Request-Datei unter **Tests**):

#### Statuscode prüfen

```
test("Status ist 200", () => {
  expect(res.status).to.equal(200);
});
```

#### JSON-Body prüfen (Array mit 3 Einträgen)

```
test("Liste hat 3 Bücher", () => {
  const body = res.json();
  expect(Array.isArray(body)).to.equal(true);
  expect(body.length).to.equal(3);
});
```

#### **Einzelnes Buch prüfen (Felder & Werte-Beispiele)**

```
test("Buch hat Pflichtfelder", () => {
  const b = res.json();
  expect(b).to.have.property("book_uuid");
  expect(b).to.have.property("title");
  expect(b).to.have.property("author");
  expect(b).to.have.property("price");
  expect(b).to.have.property("format");
```

 $update: \\ 2025/09/18 \\ modul: m450: learning units: lu08: aufgaben: tests bruno https://wiki.bzz.ch/modul/m450/learning units/lu08/aufgaben/tests bruno? rev=1758181313$ 

```
expect(b).to.have.property("isbn");
});
```

Weitere Beispiele: Bruno Docs: Tests

# **Abgabe**

Exportiere dein Bruno-Projekt (Ordner mit allen ``.bru``-Dateien und Environments) als ZIP und lade die **ZIP-Datei** hier in Moodle hoch.

#### M450-LU08



No sa Marcel Suter (angepasst)

From:

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

Permanent link:

https://wiki.bzz.ch/modul/m450/learningunits/lu08/aufgaben/testsbruno?rev=175818131

Last update: 2025/09/18 09:41



https://wiki.bzz.ch/ Printed on 2025/11/27 12:16