# LU03.A04/L04 - Webformulare: JSON-Daten lesen, bearbeiten, einfügen und löschen

1. Bestandteile eine node-Servers kennen.
2. Basis-Server ausführen können.
3. Daten anzeigen, verändern und schreiben können.

## Rahmenbedingungen

- Sozialform: individual
- Hilfsmittel: OpenBooks
- Zeit: 30 Minutes
- Erwartetes Resultat: Node-Server, der verschiedene abgefragte Daten am Monitor/Browser bearbeitet und ausgibt.

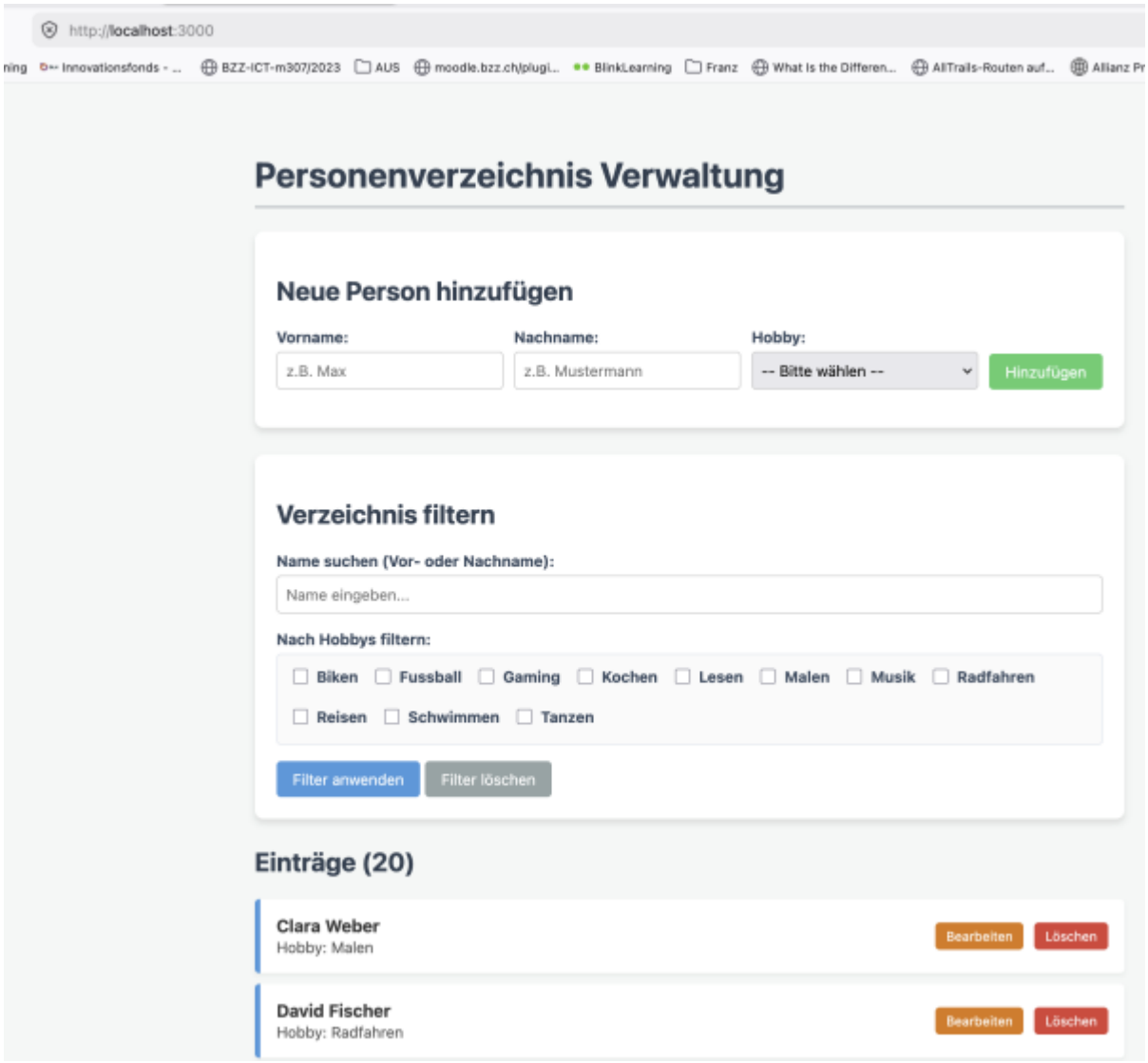
## Ausgangslage

Um Rahmen des Web-Entwicklungsprojekts „Personenverzeichnis“ soll eine bestehende Node.js-Applikation mit Express und EJS (Embedded JavaScript) finalisiert und in Betrieb genommen werden. Bisher blockieren fehlerhafte Dateipfade, unvollständige Routen-Variablen sowie Caching-Probleme im Firefox-Browser das fehlerfreie Hinzufügen, Bearbeiten und Löschen von Datensätzen. Ihre Aufgabe ist es, die Ordnerstruktur zu bereinigen, die Server-Logik zu vervollständigen und die Benutzeroberfläche mit einem dynamischen Hobby-Dropdown sowie optimierten Buttons auszustatten. Der lokale Prototyp muss anschliessend auf Port 3000 fehlerfrei und stabil laufen.

## Auftrag

Installieren Sie die Ihnen zur Verfügung gestellten Files wie unten abgebildet und lassen Sie die *Mini-Applikation* laufen. Legen Sie Personen an, filtern diese und löschen auch Datensätze.

Wenn alles korrekt gemacht wurde, sieht Ihre Applkation ungefähr wie folgt aus:




# Benötigte File und Ablage-Struktur

Die nachfolgenden Files müssen wir folgt im VS Studio Code abgelegt werden: Benötigten Files als Zipfile

Stellen Sie also sicher, dass diese den nachfolgenden Aufbau haben.

```
▼ a04 (Projekt-Wurzelverzeichnis)
  ▼ public
    ▼ css
        style.css
  ▼ views
      index.ejs
    data.json
    server.js
```

# Arbeitsschritte

### 1. Order & Files

- Laden Sie die Files herunter
- Importieren Sie diese ins VS Studio Code

### 2. Projektabhängigkeiten einrichten

Bevor das Skript starten kann, müssen die benötigten Node-Pakete installiert werden. Gehen Sie also in Ihr Projektverzeichnis und tippen den nachfolgenden Befehl in die Kommandozeile.

**Hinweis:** Unter Windows muss der gewählte Terminal ein **Commandline** oder **Command Prompt** sein.

```
npm init -y
```

**3. Express installieren **

Die benötigten Module installieren: Express (für den Server) und EJS (für die Templates) müssen im Projekt registriert werden:

```
npm install express ejs
```

** 4. Schreibberechtigung der JSON-Datei setzten**

Damit wir die JSON-Datei auch beschreiben können, müssen wir hier die Berechtigung anpassen. Gehen Sie im Terminal an die Stelle, an dem die JSON-Datei gespeichert wurde und geben die nachfolgenden Codezeilen ein.

** 4.1 Winddows-Powershell**

```
# Entfernt das Schreibschutz-Attribut (R = Read-Only) von der Datei
attrib -r data.json
```

### 4.2 MacOS

```
# Gibt dem Besitzer (User) Lese- und Schreibrechte (rw-)
chmod 644 data.json

# Falls der ganze Ordner blockiert ist, rekursiv für alle Dateien freigeben:
chmod -R 755
```

# Lösungen

Gibt keine, da die Aufgabe darin besteht die Applikation zu laufen zu bringen.

Volkan Demir

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

Permanent link:
**https://wiki.bzz.ch/de/modul/m307/learningunits/lu03/aufgaben/04?rev=1779968807**

Last update: **2026/05/28 13:46**