G1F Praktischer Nachweis

Ich kann eine Anbindung an eine NoSQL Datenbank implementieren. (z. B. API)

Fragenstellung und Lernziele

In dieser Übung geht es darum, eine minimalistische API in Python und Flask zu erstellen, die eine Verbindung zu einer MongoDB herstellt. Dabei sollen folgende Lernziele erreicht werden:

- Importieren von CSV-Daten in eine NoSQL-Datenbank
- Implementieren und Testen von REST-Endpunkten (GET und POST)

Umsetzung

Container Setup

Docker-compose Datei erstellen:

```yaml
services:
  mongodb:
    image: mongo:8.0
    container_name: books-mongo
    command: ["mongod", "--auth"]
    environment:
      MONGO_INITDB_ROOT_USERNAME: admin
      MONGO_INITDB_ROOT_PASSWORD: secret123
    ports:
      - "27018:27017"
    volumes:
      - ./mongo-data:/data/db
      - ./books.csv:/tmp/books.csv:ro
    restart: unless-stopped

  flask-app:
    build: ./flask-app
    container_name: flask-books-api
    ports:
      - "5000:5000"
    environment:
      FLASK_APP: app.py
      FLASK_RUN_HOST: 0.0.0.0
      FLASK_RUN_PORT: 5000
      MONGO_URI: mongodb://admin:secret123@mongodb:27017
      DB_NAME: librarydb
```

```
        depends_on:
          - mongodb
```

CSV-Daten:

```
isbn,title,author,year
9780140449136,Meditations,Marcus Aurelius,2006
9780261103573,The Hobbit,J.R.R. Tolkien,1937
9780553386790,Thinking\, Fast and Slow,Daniel Kahneman,2011
9780307269997,The Road,Cormac McCarthy,2006
```

Import CSV-Daten in MongoDB:

```
docker exec -it books-mongo mongoimport `
  --username admin `
  --password secret123 `
  --authenticationDatabase admin `
  --db librarydb `
  --collection books `
  --type csv `
  --headerline `
  --file /tmp/books.csv
```

Flask API

Wir wollen folgende Endpunkte implementieren:

```
GET /books - liefert alle Dokumente

GET /books/:id – liefert ein Dokument per ObjectId

POST /books – legt ein neues Dokument an
```

Projektsetup

Layout

```
.
├── docker-compose.yml
├── books.csv
└── flask-app
    ├── Dockerfile
    ├── requirements.txt
    └── app.py
```

Requirements.txt

```
Flask
pymongo
python-dotenv
```

Dockerfile

```dockerfile
FROM python:3.13-slim
WORKDIR /app

COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt

COPY . .

CMD ["flask", "run"]
```

API-Code:

```python
import os
from flask import Flask, request, jsonify
from pymongo import MongoClient
from bson.objectid import ObjectId
from dotenv import load_dotenv

load_dotenv()

mongo_uri = os.getenv('MONGO_URI')
db_name = os.getenv('DB_NAME', 'librarydb')

if not mongo_uri:
    raise RuntimeError('MONGO_URI not set')

client = MongoClient(mongo_uri)
db = client[db_name]
books_collection = db['books']

app = Flask(__name__)

@app.route('/books', methods=['GET'])
def get_books():
    books = list(books_collection.find())
    for book in books:
        book['id'] = str(book['_id'])
        del book['_id']
    return jsonify(books), 200
```

```python
@app.route('/books/<id>', methods=['GET'])
def get_book(id):
    if not ObjectId.is_valid(id):
        return jsonify({'error': 'Invalid ID'}), 400
    book = books_collection.find_one({'_id': ObjectId(id)})
    if not book:
        return jsonify({'error': 'Not found'}), 404
    book['id'] = str(book['_id'])
    del book['_id']
    return jsonify(book), 200

@app.route('/books', methods=['POST'])
def create_book():
    data = request.json
    title = data.get('title')
    author = data.get('author')
    isbn = data.get('isbn')
    year = data.get('year')
    if not title or not author or not isbn or not isinstance(year, int):
        return jsonify({'error': 'Missing fields or wrong types'}), 400
    result = books_collection.insert_one({
        'isbn': isbn,
        'title': title,
        'author': author,
        'year': year
    })
    return jsonify({'insertedId': str(result.inserted_id)}), 201

@app.errorhandler(Exception)
def handle_exception(e):
    return jsonify({'error': 'Internal server error'}), 500

if __name__ == '__main__':
    app.run(
        host=os.getenv('FLASK_RUN_HOST', '0.0.0.0'),
        port=int(os.getenv('FLASK_RUN_PORT', 5000))
    )
```

Starten der Flask-App

```
docker compose up -d
```

		g1f_flask	-	-	-
	●	books-mongo	6b4fc8973f91	mongo:8.0	27018:27017

flask-books-api   90583e00a031   g11_flask-flask-app   5000:5000

HTTP-Requests testen

GET /books

```
curl -X GET http://localhost:5000/books
```

```
# curl http://localhost:5000/books | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   483  100   483    0      0  10332      0 --:--:-- --:--:-- --:--:-- 10500
[
  {
    "author": "J.R.R. Tolkien",
    "id": "6853f2d7a6afcb2da5c44d5e",
    "isbn": 9780261103573,
    "title": "The Hobbit",
    "year": 1937
  },
  {
    "author": "Marcus Aurelius",
    "id": "6853f2d7a6afcb2da5c44d5f",
    "isbn": 9780140449136,
    "title": "Meditations",
    "year": 2006
  },
  {
    "author": "Cormac McCarthy",
    "id": "6853f2d7a6afcb2da5c44d60",
    "isbn": 9780307269997,
    "title": "The Road",
    "year": 2006
  },
  {
    "author": "Fast and Slow",
    "field4": 2011,
    "id": "6853f2d7a6afcb2da5c44d61",
    "isbn": 9780553386790,
    "title": "Thinking",
    "year": "Daniel Kahneman"
  }
]
```

POST /books

```
curl -X POST "http://localhost:5000/books" `
    -H "Content-Type: application/json" `
    -d '{
        "isbn": "9780307269997",
        "title": "The Road",
        "author": "Cormac McCarthy",
        "year": 2006
    }'
```

```
⊗ Oliver ▪ ~/../school/m165/g1f_flask
# curl.exe -X POST "http://localhost:5000/books" `
>       -H "Content-Type: application/json" `
>       -d '{
>           "isbn": "9780307269997",
>           "title": "The Road",
>           "author": "Cormac McCarthy",
>           "year": 2006
>       }'
{"insertedId":"6853f36103ffd2a09161f3c2"}
```

GET /books/:id

```
curl -X GET http://localhost:5000/books/<id>
```

```
⊗ Oliver ▪ ~/../school/m165/g1f_flask
# curl http://localhost:5000/books/6853f36103ffd2a09161f3c2 | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   115  100   115    0     0   3908      0 --:--:-- --:--:-- --:--:--  3965
{
  "author": "Cormac McCarthy",
  "id": "6853f36103ffd2a09161f3c2",
  "isbn": "9780307269997",
  "title": "The Road",
  "year": 2006
}
```