

# LU06.A04 - Erstellen einer RESTful API für die ToDo-Liste mit Flask



Deine Aufgabe ist es, eine RESTful API mit Flask zu erstellen, um eine einfache ToDo-Liste zu verwalten. Du erhältst bereits implementierte Klassen für die ToDo-Elemente und die dazugehörige DAO-Klasse.

## Kurzzusammenfassung

- Implementiere die Endpunkte für CRUD-Operationen (Create, Read, Update, Delete)
- Verwende JSON für den Datenaustausch
- Implementiere passende HTTP-Statuscodes

## Vorgehen

### 0. Vorbereitungen

1. Akzeptiere das GitHub Classroom Assignment
2. Klone dein persönliches Repository in die Entwicklungsumgebung

### 1. Implementiere CRUD-Endpunkte

- Erstelle die folgenden REST-Endpunkte für die CRUD-Operationen. Diese sollten die Methoden der TodoDao-Klasse verwenden.
1. **POST** /todos: Für das Hinzufügen eines neuen ToDo-Elements
  2. **GET** /todos: Für das Abrufen aller ToDo-Elemente
  3. **GET** /todos/<id>: Für das Abrufen eines einzelnen ToDo-Elements nach ID
  4. **PUT** /todos/<id>: Für das Aktualisieren eines vorhandenen ToDo-Elements
  5. **DELETE** /todos/<id>: Für das Löschen eines ToDo-Elements

```
# Beispiel für einen Endpunkt zum Hinzufügen eines neuen ToDo-Elements
@app.route('/todos', methods=['POST'])
def add_todo():
    # Dein Code hier
```

### 2. Datenaustausch mit JSON

- Verwende JSON für den Datenaustausch zwischen Client und Server. Du kannst jsonify aus dem Flask-Paket verwenden, um Python-Objekte in JSON umzuwandeln.

### 3. HTTP-Statuscodes

- Implementiere passende HTTP-Statuscodes für erfolgreiche und fehlerhafte Anfragen. Zum Beispiel:
  1. 201 Created für erfolgreich erstellte Ressourcen
  2. 404 Not Found wenn eine Ressource nicht gefunden wird
  3. 400 Bad Request für ungültige Anfragen

### 4. Teste deine API

Via API-Testing.http-File im Workspace

► Run in Postman

oder via:

### 5. Abgabe

Die Abgabe der Lösung erfolgt als Push in das persönliche GitHub-Repository.

⇒ GitHub Repo für externe Besucher

GitHub Repository <https://github.com/templates-python/m323-lu06-a04-restful>

Lernende am BZZ müssen den Link zum GitHub Classroom Assignment verwenden



© Kevin Maurizi

From:

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

Permanent link:

<https://wiki.bzz.ch/modul/m323/learningunits/lu06/aufgaben/restful>

Last update: **2024/03/28 14:07**

