

AIDI2004 – AI in Enterprise systems

Assignment #3



Group 9

NAME: Dheeraj Reddy

STUDENT NUMBER: 100811270

NAME: Ayush Parolia

STUDENT NUMBER: 100799278

Step 1: Creating Database

```
# importing libraries
from flask import Flask, request, Response, jsonify
from flask_sqlalchemy import SQLAlchemy

# creating an instance of the flask app
app = Flask(__name__)

# Configure the database
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///database.db'
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False

from settings import *
import json

# Initializing our database
db = SQLAlchemy(app)

class Details(db.Model):
    __tablename__ = 'details'
    id = db.Column(db.Integer, primary_key=True)
    studentid = db.Column(db.Integer, nullable=False)
    firstname = db.Column(db.String(64), nullable=False)
    lastname = db.Column(db.String(64), nullable=False)
    dob = db.Column(db.String(64), nullable=False)
    amountdue = db.Column(db.Integer, nullable=False)

from details import db
db.create_all()
```

Step 2: Creating Basic Flask app

```
Cdheerajreddy2020@DESKTOP-GMQ5T57:/mnt/c/Users/Shravani/Downloads/Durham college/Sem-2
python3 app.py
* Serving Flask app "settings" (lazy loading)
* Environment: production
  WARNING: Do not use the development server in a production environment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 104-175-450
```

Step 3: For Create

```
@app.route('/students', methods=['POST'])
def add_student():
    '''Function to add new student to our database'''
    request_data = request.get_json()
    print(request_data);
    Details.add_student(request_data["studentid"], request_data["firstname"],
                        request_data["lastname"], request_data["dob"],
                        request_data["amountdue"])
    response = Response("Student added", 201, mimetype='application/json')
    return response
```

Step 4: Read

```
def get_student(_id):
    return [Details.json(Details.query.filter_by(id=_id).first())]
```

Step 5: Update

```
def update_student(_id, _studentid, _firstname, _lastname, _dob, _amountdue):
    student_to_update = Details.query.filter_by(id=_id).first()
    student_to_update.studentid = _studentid
    student_to_update.firstname = _firstname
    student_to_update.lastname = _lastname
    student_to_update.dob = _dob
    student_to_update.amountdue = _amountdue
    db.session.commit()
```

Step 6: For Delete

```
def delete_student(_id):
    Details.query.filter_by(id=_id).delete()
    db.session.commit()
```

Step 7 : Show all records

Overview GET http://127.0.0.1:5000/... POST http://127.0.0.1:5000/... PUT http://127.0.0.1:5000/... + ... No Environment

http://127.0.0.1:5000/students Save

GET http://127.0.0.1:5000/students Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (4) Test Results Status: 200 OK Time: 31 ms Size: 674 B Save Response

Pretty Raw Preview Visualize JSON

```
10  },
11  {
12    "amountdue": 3000,
13    "dob": "1991/06/01",
14    "firstname": "DHeeraj",
15    "id": 3,
16    "lastname": "Reddy",
17    "studentid": 100811270
18  },
19  {
20    "amountdue": 6000,
21    "dob": "1991/01/01",
22    "firstname": "Ayush",
23    "id": 4,
24    "lastname": "Parolia",
25    "studentid": 100799278
26  }
27 ]
28
```

Step 8 : Postman

http://127.0.0.1:5000/students Save

POST http://127.0.0.1:5000/students

Params Authorization Headers (8) Body Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL JSON

```
1  {
2    "studentid": 100811270,
3    "firstname": "DHeeraj",
4    "lastname": "Reddy",
5    "dob": "1991/06/01",
6    "amountdue": 3000
7  }
```

Body Cookies Headers (4) Test Results Status: 201 CREATED Time: 114 ms Size: 164 B S

Pretty Raw Preview Visualize JSON

1 Student added

http://127.0.0.1:5000/students/2

Save



GET http://127.0.0.1:5000/students/2

Send

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (4) Test Results

Status: 200 OK Time: 14 ms Size: 305 B Save Response

Pretty Raw Preview Visualize JSON



```
1 {}
2
3   "amountdue": 3000,
4   "dob": "1991/06/01",
5   "firstname": "DHeeraj",
6   "id": 2,
7   "lastname": "Reddy",
8   "studentid": 100811270
```

Bootcamp Runner Trash

http://127.0.0.1:5000/students/2

Save



</

DELETE http://127.0.0.1:5000/students/2

Send

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings

Cookies

none form-data x-www-form-urlencoded raw binary GraphQL Text

```
1 {
2   "studentid": 100811270,
3   "firstname": "DHeeraj",
4   "lastname": "Reddy",
5   "dob": "1991/05/01",
6   "amountdue": 5000
7 }
```

Body Cookies Headers (4) Test Results

Status: 200 OK Time: 78 ms Size: 161 B Save Response

Pretty Raw Preview Visualize JSON



```
1 Student Deleted
```

Step 9: Github repo link : https://github.com/dheerajreddy2020/CRUD_in_Flask

The screenshot shows a GitHub repository interface. At the top, there's a header with 'master' branch selected, '1 branch', and '0 tags'. Buttons for 'Go to file', 'Add file', and 'Code' are visible. Below this, a commit history table lists files: __pycache__, app.py, database.db, details.py, and settings.py, all marked as 'Initial Commit' and '39 minutes ago'. A sidebar on the right contains sections for 'About' (CRUD metho), 'Releases' (No releases publi, Create a new rele), 'Packages' (No packages pub, Publish your first), and 'Languages' (Python 100.0%). A blue banner at the bottom encourages adding a README.

File	Commit	Time
__pycache__	Initial Commit	39 minutes ago
app.py	Initial Commit	39 minutes ago
database.db	Initial Commit	39 minutes ago
details.py	Initial Commit	39 minutes ago
settings.py	Initial Commit	39 minutes ago

Step 10 : Collaboration Plan

Task	Responsible	Time
Flask analysis	Dheeraj	2 Hrs
Flask app creation	Dheeraj, Ayush	4 Hrs
Postman testing	Ayush	3 Hrs
Document Creation	Dheeraj, Ayush	1 Hr