**Contact Manager API**

In this application, the functions which are used are: add\_contact, show\_contacts, and delete\_contacts, which uses the HTTP methods POST, GET, and DELETE.

**Functions:**

create\_database() : Creates the SQLite database and the contacts table if they do not already exist.

add\_contact(contact: Contact) : Inserts a new contact into the database.

get\_contacts() : Retrieves all contacts from the database.

delete\_contact(name: str) : Deletes a contact from the database.

**Dependencies:**

FastAPI : A hight performance web framework for building APIs with Python.

Pydantics : data validation and management library for Python apps.

SQlite3 : Lightweight serverless database engine.

**Code Breakdown**:



1) Here we install the neccessary dependencies, then create an instance(app) and defines a pydantic model named 'Contact' with two fields name and number.

2) Function Management: As discussed above we have used 4 functions named create\_database, add\_contact, get\_contacts and delete\_contact.

3) **API Endpoints**:

@app.post("/add\_contact/")

Defines a post endpoint for adding a contact

@app.get("/get\_contacts/")

Defines a get endpoint for retrieving all the contacts, if no contacts found it raises an HTTPException.

@app.delete("/delete\_contact/{name}")

Defines a delete endpoint for deleting a contact, if the contact is not found it raises an HTTPException.

4) Script Startup

if \_\_name\_\_ == "\_\_main\_\_":

create\_database()

This is where the loop starts and it creates a database if it does not exist when starting the app.

**Virtual Environment Setup**:

To create a virtual env:

python -m venv name(venv name)

To activate the virtual env:

.\name\Scripts\activate

Install fastapi:

pip install fastapi

Install uvicorn:

pip install "uvicorn[standard]"

uvicorn name:app --reload