**Employee Attendance tracking System**



*First we'll start by creating a SQlite database to store employees information, in this program, we'll create a table named "employees" with column for employee ID, name and status (e.g., present or absent).*

*This function connects to the SQlite database and creates the "employees" table if it doesn't already exist.*

**Creating Endpoints:**

*We'll define endpoints for handling CRUD (create, read, update, delete) operations on employees.*

1. *GET/employees/{emp\_id}: Retrieves details of an employee by ID.*
2. *POST/employees/: Adds a new employee*
3. *PUT/employees/{emp\_id}: Updates details of existing employee*
4. *DELETE/employees/{emp\_id}: Deletes an employee.*

**Dependencies:**

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

*SQlite3: Lightweight server-less database engine.*

**Virtual Environment Setup(if dockerfile is not created):**

*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*

**Dockerfile Breakup:**

*FROM python:3.9-slim*

*This line specifies the base image for the Dockerfile, it is the official image from Docker Hub. This base image allows to run the Attendance tracker application within a light weight container environment.*

*WORKDIR/app*

*This line creates the directory inside the container and all the commands will be executed in this directory.*

*RUN pip install --no cache-dir fastapi uvicorn*

*This line installs the python packages fastapi and uvicorn using the pip inside the docker container. These packages are used for building web app using fastapi. --no-cache-dir, this is used for avoid caching the downloaded package files which keeps the Docker image smaller.*

*COPY..*

*This line copies all the local directory files to the /app directory in the Docker container.*

*EXPOSE 8000*

*This line exposes port 8000 on the Docker container and it will listen on port 8000 at runtime.*

*CMD ["uvicorn", "attendancetracker:app", "--host", "0.0.0.0", "--port", "8000"]*

*This command will execute when the Docker container starts, it launches the uvicorn server, 'attendancetracker:app' this tells the uvicorn to app object from the file named attendancetracker.py host port tells uvicorn to listen for incoming connection from any IP address and port 8000 specifies the port number on which uvicorn should listen for connections*.

**Running the application:**

*To run the application, build the docker image and run the container:*

*docker build -t attendancetracker .*

*docker run -p 8000:8000 attendancetracker*

*The application will be accessible at* '[http://localhost:8000](http://localhost:8000/)'

*You can also access the Swagger UI from* '<http://localhost:8000/docs>'