Running Python with MySQL using **Docker Compose**.

It will:

* Start a MySQL database container.
* Automatically create a table and insert data using an initialization script.
* Start a Python container that connects to the database and displays the records.

**📁 Folder Structure**

python-mysql-docker/

│

├── docker-compose.yml

├── init/

│ └── init.sql

├── app/

│ ├── main.py

│ └── requirements.txt

└── Dockerfile

**1. docker-compose.yml**

version: '3.8'

services:

db:

image: mysql:8.0

container\_name: mysql\_db

environment:

MYSQL\_ROOT\_PASSWORD: rootpassword

MYSQL\_DATABASE: myapp\_db

MYSQL\_USER: myuser

MYSQL\_PASSWORD: mypassword

ports:

- "3306:3306"

volumes:

- ./init:/docker-entrypoint-initdb.d

networks:

- mynetwork

python-app:

build: .

container\_name: python\_app

depends\_on:

- db

networks:

- mynetwork

networks:

mynetwork:

driver: bridge

**2. Dockerfile (for Python app)**

FROM python:3.11-slim

WORKDIR /app

COPY app/requirements.txt .

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

COPY app/ .

CMD ["python", "main.py"]

**3. init/init.sql — (Database & table initialization)**

CREATE TABLE IF NOT EXISTS users (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100) NOT NULL

);

INSERT INTO users (name) VALUES ('Alice'), ('Bob'), ('Charlie');

This script runs **automatically** when the MySQL container starts.

**4. app/requirements.txt**

mysql-connector-python

**5. app/main.py — (Python script to fetch and display records)**

import time

import mysql.connector

# Wait a bit to ensure the DB is ready

time.sleep(10)

# Connect to MySQL

conn = mysql.connector.connect(

host="db", # service name from docker-compose

user="myuser",

password="mypassword",

database="myapp\_db"

)

cursor = conn.cursor()

cursor.execute("SELECT \* FROM users")

print("Users in the database:")

for row in cursor.fetchall():

print(row)

cursor.close()

conn.close()

**6. Run the project**

In the project root folder:

**docker compose up --build**

You should see output similar to:

Users in the database:

(1, 'Alice')

(2, 'Bob')

(3, 'Charlie')

**To stop and clean up**

docker compose down -v