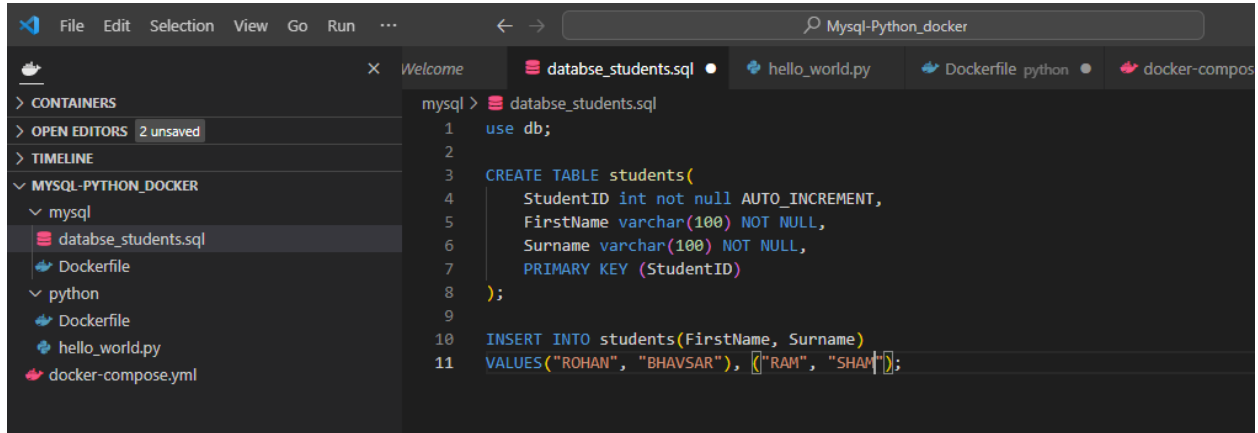


DOCKER COMPOSE

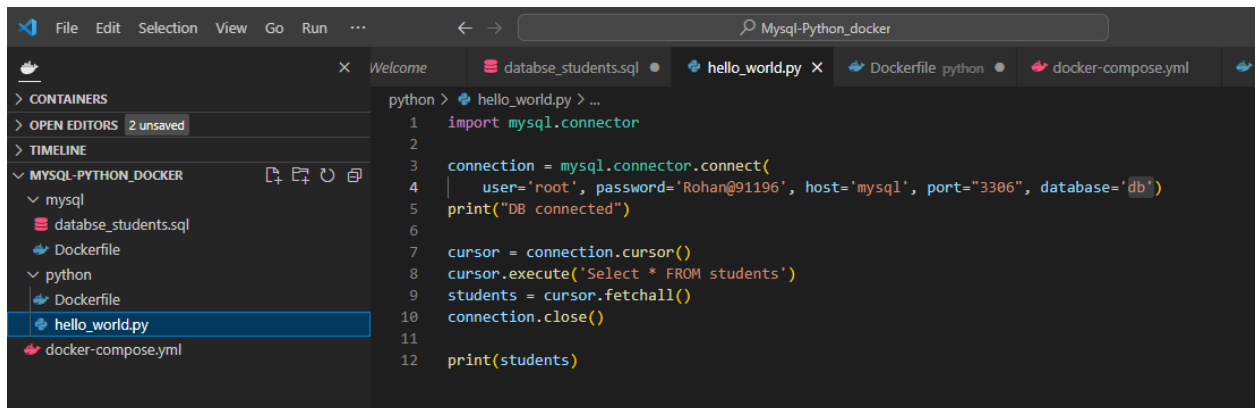
STEP 1: Creating MYSQL database and table



The screenshot shows the VS Code editor interface with the 'Mysql-Python_docker' workspace. The left sidebar displays the Explorer view with a tree structure: CONTAINERS, OPEN EDITORS (2 unsaved), TIMELINE, and MYSQL-PYTHON_DOCKER. Under MYSQL-PYTHON_DOCKER, there are subfolders for 'mysql' and 'python'. The 'mysql' folder contains 'database_students.sql', 'Dockerfile', and 'python' folder. The 'python' folder contains 'Dockerfile', 'hello_world.py', and 'docker-compose.yml'. The main editor area shows the 'database_students.sql' file with the following SQL code:

```
mysql > database_students.sql
1 use db;
2
3 CREATE TABLE students(
4     StudentID int not null AUTO_INCREMENT,
5     FirstName varchar(100) NOT NULL,
6     Surname varchar(100) NOT NULL,
7     PRIMARY KEY (StudentID)
8 );
9
10 INSERT INTO students(FirstName, Surname)
11 VALUES("ROHAN", "BHAVASAR"), ("RAM", "SHAM");
```

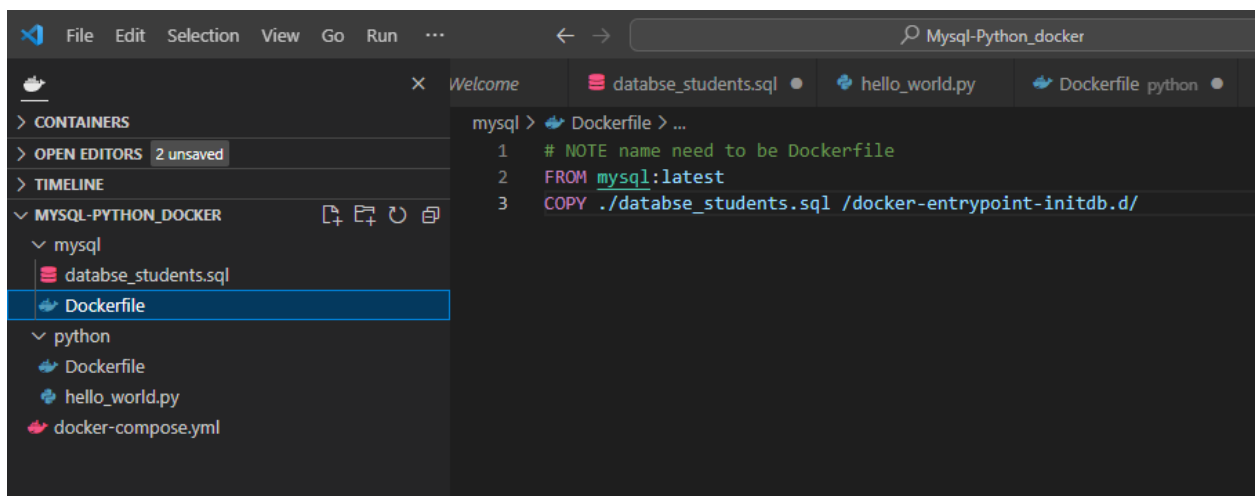
STEP 2: CREATING PYTHON FILE



The screenshot shows the VS Code editor interface with the 'Mysql-Python_docker' workspace. The left sidebar displays the Explorer view with a tree structure: CONTAINERS, OPEN EDITORS (2 unsaved), TIMELINE, and MYSQL-PYTHON_DOCKER. Under MYSQL-PYTHON_DOCKER, there are subfolders for 'mysql' and 'python'. The 'mysql' folder contains 'database_students.sql', 'Dockerfile', and 'python' folder. The 'python' folder contains 'Dockerfile', 'hello_world.py', and 'docker-compose.yml'. The main editor area shows the 'hello_world.py' file with the following Python code:

```
python > hello_world.py > ...
1 import mysql.connector
2
3 connection = mysql.connector.connect(
4     user='root', password='Rohan@91196', host='mysql', port="3306", database='db')
5 print("DB connected")
6
7 cursor = connection.cursor()
8 cursor.execute('Select * FROM students')
9 students = cursor.fetchall()
10 connection.close()
11
12 print(students)
```

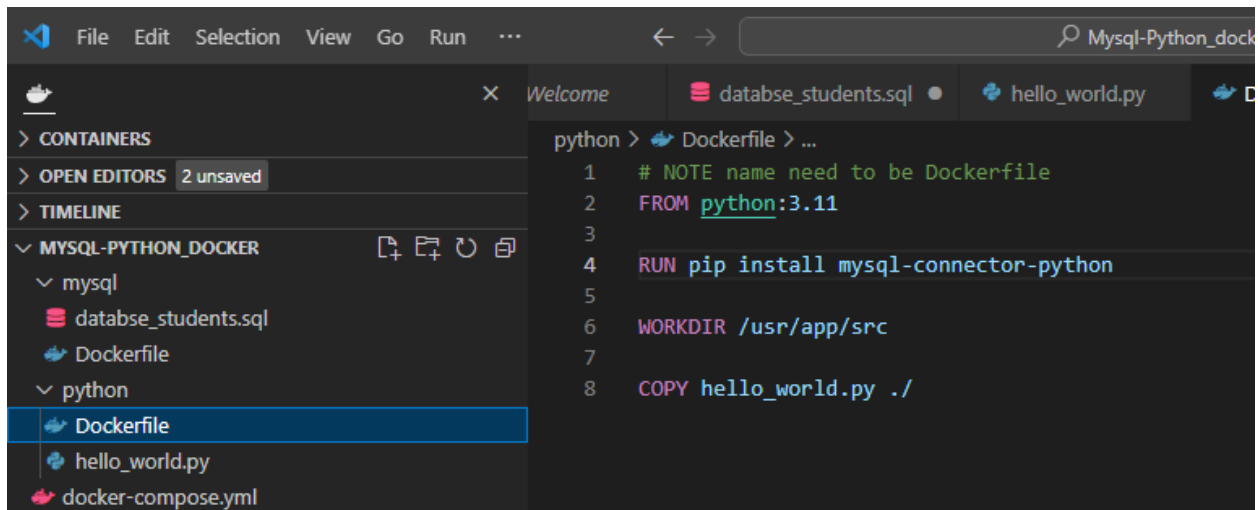
Step 3: Creating docker file for mysql container



The screenshot shows the VS Code editor interface with the 'Mysql-Python_docker' workspace. The left sidebar displays the Explorer view with a tree structure: CONTAINERS, OPEN EDITORS (2 unsaved), TIMELINE, and MYSQL-PYTHON_DOCKER. Under MYSQL-PYTHON_DOCKER, there are subfolders for 'mysql' and 'python'. The 'mysql' folder contains 'database_students.sql', 'Dockerfile', and 'python' folder. The 'python' folder contains 'Dockerfile', 'hello_world.py', and 'docker-compose.yml'. The main editor area shows the 'Dockerfile' file with the following Dockerfile code:

```
mysql > Dockerfile > ...
1 # NOTE name need to be Dockerfile
2 FROM mysql:latest
3 COPY ./database_students.sql /docker-entrypoint-initdb.d/
```

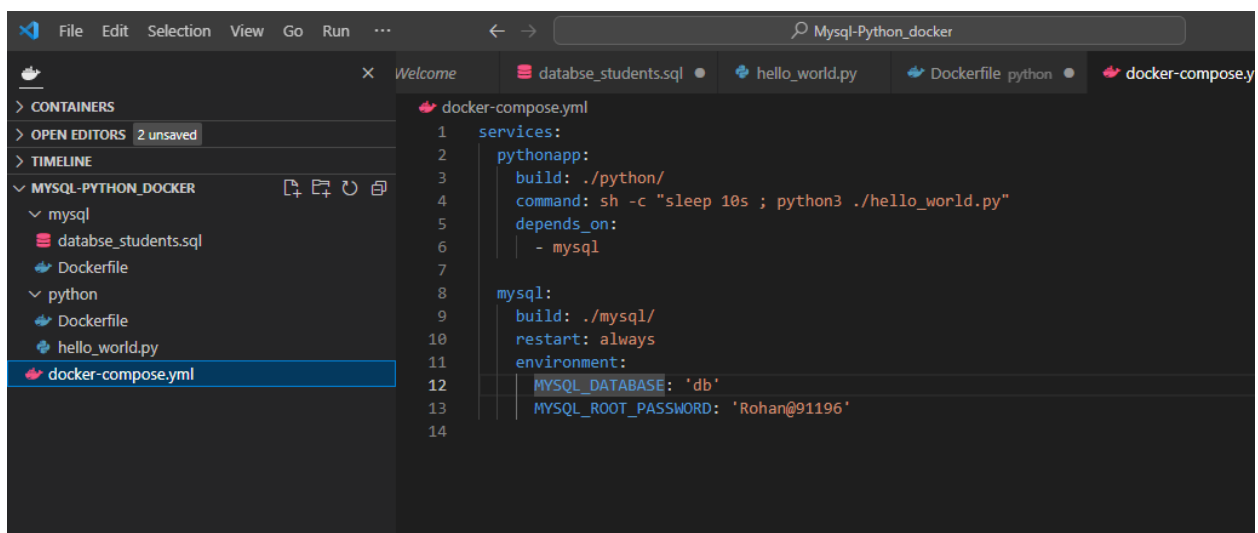
Step 4: Creating docker file for python app



The screenshot shows the Visual Studio Code editor with a project named 'Mysql-Python_docker'. The left sidebar shows the 'CONTAINERS' view with a tree structure: 'MYSQL-PYTHON_DOCKER' > 'mysql' > 'databse_students.sql', 'Dockerfile', and 'python' > 'Dockerfile' (selected). The main editor shows the content of the selected 'Dockerfile' for the 'python' container. The file contains the following Dockerfile instructions:

```
python > Dockerfile > ...
1  # NOTE name need to be Dockerfile
2  FROM python:3.11
3
4  RUN pip install mysql-connector-python
5
6  WORKDIR /usr/app/src
7
8  COPY hello_world.py ./
```

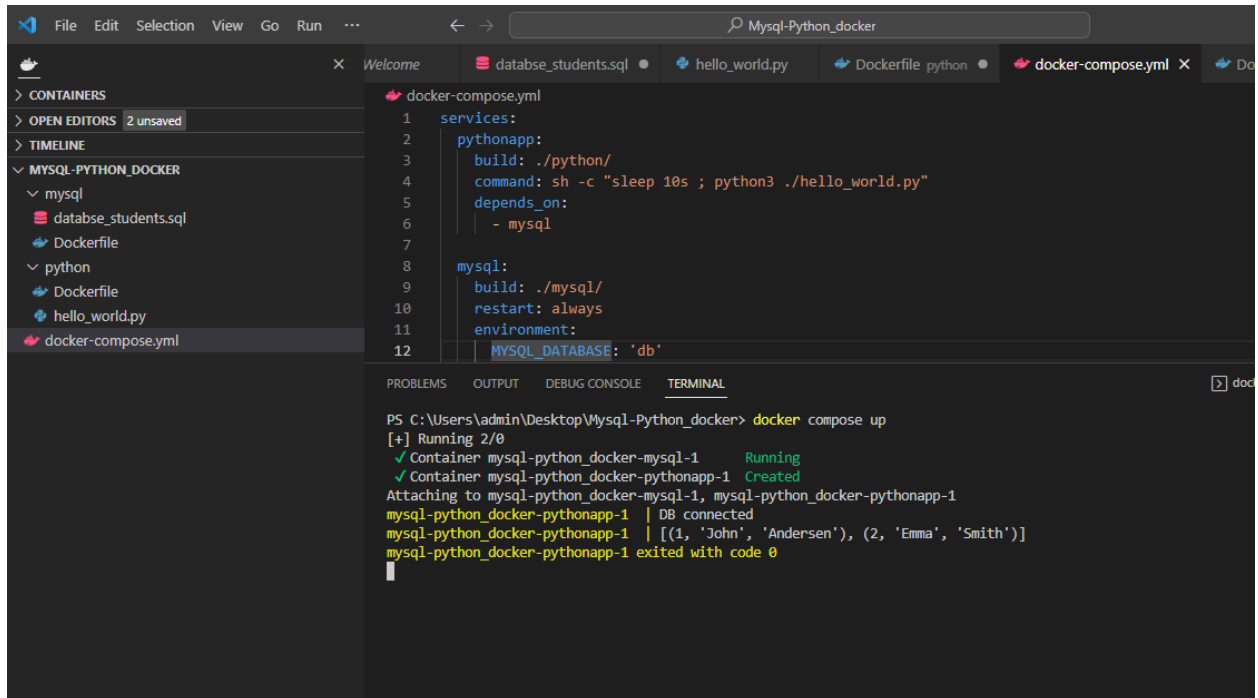
Step 5: creating docker compose .yaml file



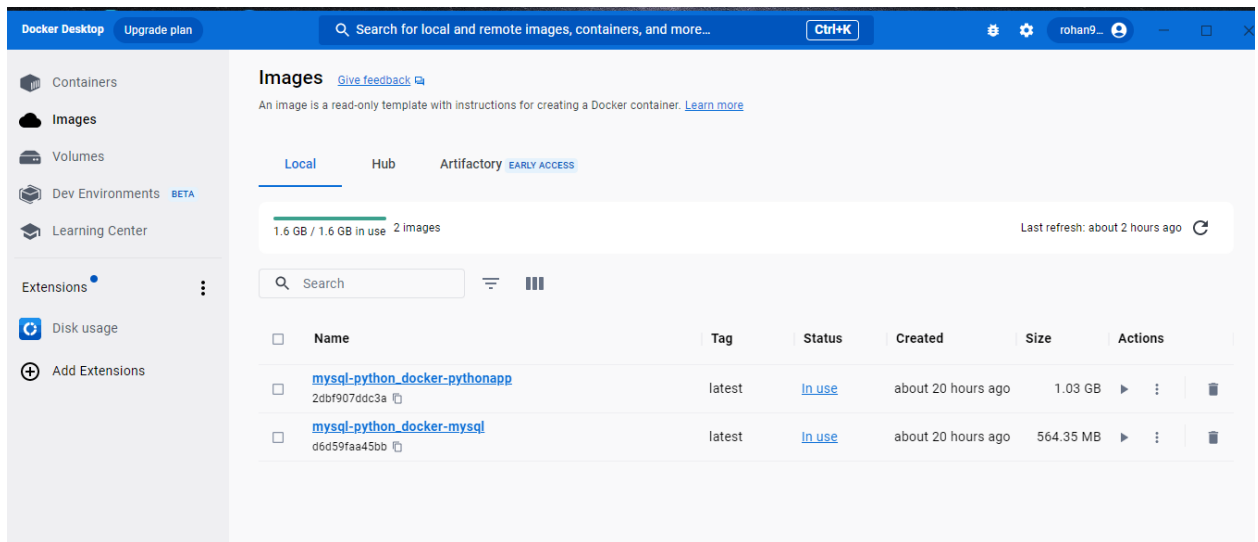
The screenshot shows the Visual Studio Code editor with the same project. The left sidebar shows the 'CONTAINERS' view with the tree structure: 'MYSQL-PYTHON_DOCKER' > 'mysql' > 'databse_students.sql', 'Dockerfile', and 'python' > 'Dockerfile', 'hello_world.py', and 'docker-compose.yml' (selected). The main editor shows the content of the selected 'docker-compose.yml' file. The file contains the following Docker Compose configuration:

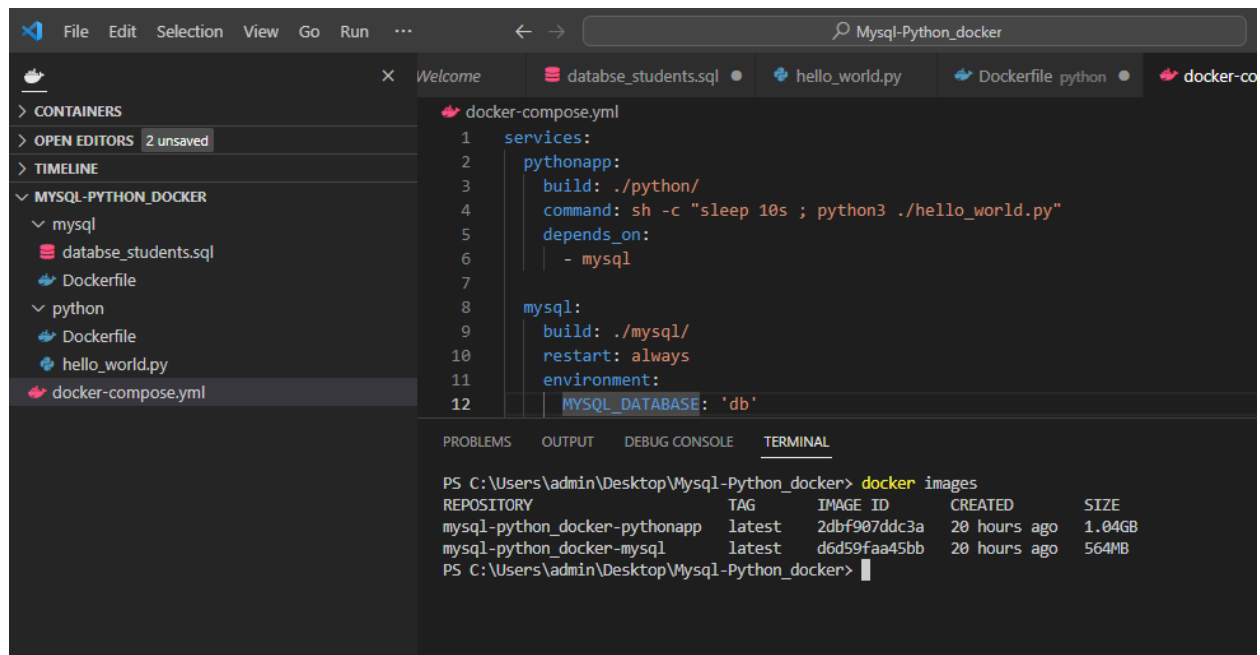
```
docker-compose.yml
1  services:
2    pythonapp:
3      build: ./python/
4      command: sh -c "sleep 10s ; python3 ./hello_world.py"
5      depends_on:
6        - mysql
7
8    mysql:
9      build: ./mysql/
10     restart: always
11     environment:
12       MYSQL_DATABASE: 'db'
13       MYSQL_ROOT_PASSWORD: 'Rohan@91196'
14
```

Step 6: Running docker compose up commands



Step 7: Created images





Step 8: docker containers

