✅ A custom network  
✅ A named volume for data persistence  
✅ Environment variables (like root password and database name)  
✅ Initialization script that runs automatically when the container starts.

mysql-docker/

│── Dockerfile

│── init/

│ └── init.sql

│── docker-compose.yml

Dockerfile

|  |
| --- |
| # Start from the official MySQL image  FROM mysql:8.0  # Set environment variables  # These can be overridden at runtime if needed  ENV MYSQL\_ROOT\_PASSWORD=rootpassword  ENV MYSQL\_DATABASE=myapp\_db  ENV MYSQL\_USER=myuser  ENV MYSQL\_PASSWORD=mypassword  # Copy initialization scripts  # Any .sql or .sh files in this folder will run automatically on container startup  COPY init/ /docker-entrypoint-initdb.d/  # Expose default MySQL port  EXPOSE 3306  # Default entrypoint provided by MySQL image will handle DB setup  ENTRYPOINT ["docker-entrypoint.sh"]  CMD ["mysqld"] |

Init.sql

|  |
| --- |
| -- This script will run automatically when the container is created  CREATE TABLE IF NOT EXISTS users (  id INT AUTO\_INCREMENT PRIMARY KEY,  name VARCHAR(100) NOT NULL,  email VARCHAR(100) NOT NULL UNIQUE  );  INSERT INTO users (name, email) VALUES ('Alice', 'alice@example.com');  INSERT INTO users (name, email) VALUES ('Bob', 'bob@example.com'); |

docker build -t my-mysql-image .

docker volume create mysql\_data

docker volume ls

docker run -d \

--name mysql\_container \

-p 3306:3306 \

-v mysql\_data:/var/lib/mysql \

my-mysql-image

Test MySQL Connection

docker exec -it mysql\_container mysql -u root -p

SHOW DATABASES;

USE myapp\_db;

SHOW TABLES;

SELECT \* FROM users;

Docker Compose

|  |
| --- |
| version: "3.9"  services:  mysql:  build: .  container\_name: mysql\_container  restart: always  environment:  MYSQL\_ROOT\_PASSWORD: rootpassword  MYSQL\_DATABASE: myapp\_db  MYSQL\_USER: myuser  MYSQL\_PASSWORD: mypassword  ports:  - "3306:3306"  volumes:  - mysql\_data:/var/lib/mysql # persistent volume  - ./init:/docker-entrypoint-initdb.d # mount initialization scripts  networks:  - mynet  volumes:  mysql\_data:  networks:  mynet:  driver: bridge |

Commands:

# Build the image

docker-compose build

# Create and run the container

docker-compose up -d

# Check running containers

docker ps

# Connect to the MySQL container

docker exec -it mysql\_container mysql -u root -p