

Atividade 05

Aluno: Gustavo Fontenele Coêlho

Curso: Técnico Integrado em Informática

Período: 4

Disciplina: Banco de Dados

Professor: Ricardo Duarte Taveira

Enunciado:

“Evidenciar a execução do Prompt-02 da aula do dia 13/01/2025.”

Prompt 2:

“Como um programador de Banco de Dados, crie um Tutorial para instalar um Banco de Dados Mysql no PlaywithDocker. Crie um Docker file que cria uma imagem e em seguida cria um banco de dados chamado BD_ALUNOS com a senha de root kid_aluno e cria as tabelas TB_ALUNOS e TB_PROFESSORES.”

1º Passo: Acesse o Play with Docker

1. Vá até o site do Play With Docker.
2. Clique em “+ADD NEW INSTANCE”.

2º Passo: Crie um Dockerfile

1. No terminal do Play with Docker, crie um novo diretório para o seu projeto:

```
#####  
#                                     WARNING!!!!                               #  
# This is a sandbox environment. Using personal credentials                   #  
# is HIGHLY! discouraged. Any consequences of doing so are                   #  
# completely the user's responsibilities.                                     #  
#                                                                              #  
# The PWD team.                                                                #  
#####  
[node1] (local) root@192.168.0.23 ~  
$ mkdir mysql-docker  
[node1] (local) root@192.168.0.23 ~  
$ cd mysql-docker
```

2. Crie um arquivo chamado “Dockerfile”:

```
$ touch Dockerfile  
[node1] (local) root@192.168.0.23 ~/mysql-docker  
$
```

3. Abra o “Dockerfile” em um editor de texto utilizando “vi” ou “nano”:

```
[node1] (local) root@192.168.0.23 ~/mysql-docker  
$ vi Dockerfile
```

4. Adicione o seguinte conteúdo ao “Dockerfile”:

```
1  
2 FROM mysql:latest  
3  
4 ENV MYSQL_ROOT_PASSWORD=kid_aluno  
5 ENV MYSQL_DATABASE=BD_ALUNOS  
6  
7 COPY init.sql /docker-entrypoint-initdb.d/  
8  
9 EXPOSE 3306
```

5. Salve e saia do editor, use “Control + O” e depois digite “:wq”.

3º Passo: Criar o Script de inicialização

1. Crie um arquivo chamado “init.sql”:

```
[node1] (local) root@192.168.0.23 ~/mysql-docker
$ touch init.sql
```

2. Abra o “init.sql” em um editor de texto usando “vi” ou “nano”:

```
[node1] (local) root@192.168.0.23 ~/mysql-docker
$ vi init.sql
```

3. Adicione o seguinte conteúdo ao “init.sql”:

```
1 CREATE TABLE TE_ALUNOS (
2   id INT AUTO_INCREMENT PRIMARY KEY,
3   nome VARCHAR(100) NOT NULL,
4   idade INT NOT NULL
5 );
6
7 CREATE TABLE TE_PROFESSORES (
8   id INT AUTO_INCREMENT PRIMARY KEY,
9   nome VARCHAR(100) NOT NULL,
10  disciplina VARCHAR(100) NOT NULL
11 );
```

4. Salve e saia do editor, use “Control + O” e depois digite “:wq”.

4º Passo: Construir a Imagem Docker

1. No terminal, execute o seguinte comando para construir a imagem Docker:

```
[node1] (local) root@192.168.0.23 ~/mysql-docker
$ docker build -t mysql-alunos .
[+] Building 24.0s (7/7) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 179B
=> [internal] load metadata for docker.io/library/mysql:latest
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build context
=> => transferring context: 295B
=> [1/2] FROM docker.io/library/mysql:latest@sha256:b842a59bfaf81ea435b65be7e041c280416df8d295457f64c7c1445484464123
=> resolve docker.io/library/mysql:latest@sha256:b842a59bfaf81ea435b65be7e041c280416df8d295457f64c7c1445484464123
=> sha256:b842a59bfaf81ea435b65be7e041c280416df8d295457f64c7c1445484464123 2.62kB / 2.62kB
=> sha256:16ec22ff04f9c09b6a62ae249a729f38a64999e971381612c3d3d6215e57cda2 886B / 886B
=> sha256:92a76add85c1e036676bc7e56d381012c9fc3b0979682a3e286a8f2e05611bc 2.86kB / 2.86kB
=> sha256:c013374826adcc2afe20b2b7507cc3d8b609511f9203916569fad90a89ee1359 340B / 340B
=> sha256:1d19e87a21f388780c1e2041d7a6fa8c3b05988de43968a7ad8419eae176d5 3.44kB / 3.44kB
=> sha256:8293a632ae568992a011de522bb4457aac1a2370c9bba3f6cccd5e4dcfe1 48.43MB / 48.43MB
=> sha256:023374826adcc2afe20b2b7507cc3d8b609511f9203916569fad90a89ee1359 340B / 340B
=> sha256:9f789b8d2675092d496f458e1e38afc344a107af46d8118c6e4ebb7f88202baa 983.00kB / 983.00kB
=> sha256:96f4da41c548514a2fbd6f4a2c2927b24b0e8fcf13beac055b97c6aec7528ecd 6.90MB / 6.90MB
=> sha256:fb087646189bd4ec8e107b18652be6e01f004546094a2caf9d8226156caa5600 2.60kB / 2.60kB
=> extracting sha256:1d19e87a21f388780c1e2041d7a6fa8c3b05988de43968a7ad8419eae176d5 3.44kB
=> sha256:8293a632ae568992a011de522bb4457aac1a2370c9bba3f6cccd5e4dcfe1 48.43MB
=> extracting sha256:9f789b8d2675092d496f458e1e38afc344a107af46d8118c6e4ebb7f88202baa 983.00kB
=> extracting sha256:96f4da41c548514a2fbd6f4a2c2927b24b0e8fcf13beac055b97c6aec7528ecd 6.90MB
=> extracting sha256:fb087646189bd4ec8e107b18652be6e01f004546094a2caf9d8226156caa5600 2.60kB
=> extracting sha256:023374826adcc2afe20b2b7507cc3d8b609511f9203916569fad90a89ee1359 340B
=> extracting sha256:16ec22ff04f9c09b6a62ae249a729f38a64999e971381612c3d3d6215e57cda2 886B
=> extracting sha256:9f789b8d2675092d496f458e1e38afc344a107af46d8118c6e4ebb7f88202baa 983.00kB
=> extracting sha256:96f4da41c548514a2fbd6f4a2c2927b24b0e8fcf13beac055b97c6aec7528ecd 6.90MB
=> extracting sha256:fb087646189bd4ec8e107b18652be6e01f004546094a2caf9d8226156caa5600 2.60kB
=> extracting sha256:023374826adcc2afe20b2b7507cc3d8b609511f9203916569fad90a89ee1359 340B
=> extracting sha256:16ec22ff04f9c09b6a62ae249a729f38a64999e971381612c3d3d6215e57cda2 886B
=> extracting sha256:c3947540c6bd1536774bcc401e880a1f75bbe4eb62735208e348480322d44b 325B / 325B
=> sha256:c38bed95fb4b13e368304cf9a901a2cd21d3961e4239310a59864495a19d9946 135.73MB / 135.73MB
=> sha256:712eb897f1e5b4df8b6789fd14ae145f1f8426f3cc79149621ccb4c9f09b572 5.39kB / 5.39kB
=> extracting sha256:16ec22ff04f9c09b6a62ae249a729f38a64999e971381612c3d3d6215e57cda2 886B
=> extracting sha256:9f789b8d2675092d496f458e1e38afc344a107af46d8118c6e4ebb7f88202baa 983.00kB
=> extracting sha256:96f4da41c548514a2fbd6f4a2c2927b24b0e8fcf13beac055b97c6aec7528ecd 6.90MB
=> extracting sha256:fb087646189bd4ec8e107b18652be6e01f004546094a2caf9d8226156caa5600 2.60kB
=> extracting sha256:023374826adcc2afe20b2b7507cc3d8b609511f9203916569fad90a89ee1359 340B
=> extracting sha256:16ec22ff04f9c09b6a62ae249a729f38a64999e971381612c3d3d6215e57cda2 886B
=> extracting sha256:c3947540c6bd1536774bcc401e880a1f75bbe4eb62735208e348480322d44b 325B
=> extracting sha256:c38bed95fb4b13e368304cf9a901a2cd21d3961e4239310a59864495a19d9946 135.73MB
=> extracting sha256:712eb897f1e5b4df8b6789fd14ae145f1f8426f3cc79149621ccb4c9f09b572 5.39kB
=> [2/2] COPY init.sql /docker-entrypoint-initdb.d/
=> exporting to image
=> exporting layers
=> writing image sha256:47fd062ade7443aab751750d99f157731afd2e585b759e112ed2680d941c022e
=> naming to docker.io/library/mysql-alunos
1 warning found (use docker --debug to expand):
- SecretRefedInArgOrEnv: Do not use ARG or ENV instructions for sensitive data (ENV "MYSQL_ROOT_PASSWORD") (line 4)
```

5º Passo: Executar o Contêiner

1. Após a construção da imagem, execute o contêiner com o seguinte comando:

```
[node1] (local) root@192.168.0.23 ~/mysql-docker
$ docker run --name mysql-alunos -d -p 3306:3306 mysql-alunos
7eecb27efc587a205e47fdc1b84c9ed349f20b8fa6361bd3d22801a14ab3772b
```

6º Passo: Acessar o MySQL

1. Para acessar o MySQL, você pode usar um cliente MySQL ou o próprio terminal do contêiner. Para acessar o terminal do contêiner, execute:

```
[node1] (local) root@192.168.0.23 ~/mysql-docker
$ docker exec -it mysql-alunos mysql -u root -p
```

2. Quando solicitado, insira a senha “kid_aluno”.

```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 9.2.0 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

3. Após o login, você pode verificar se o banco de dados e as tabelas foram criados com sucesso:

```
mysql> USE BD_ALUNOS
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SHOW TABLES
-> ;
+-----+
| Tables_in_BD_ALUNOS |
+-----+
| TB_ALUNOS            |
| TB_PROFESSORES       |
+-----+
2 rows in set (0.00 sec)
```