

Aula-13-01-2025 PWD Criação BD MySql

Ricardo Duarte Taveira

13 de jan. (editado: 13 de jan.)

Turma: P4 de Informática

Feito por: Pedro Henrique Alves Barbosa e Riquelmy da Silva Ricarte

Prompt-02

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.

```
#####  
#                                     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@192.168.0.8 ~  
$ mkdir mysql-docker  
[node1] (local) root@192.168.0.8 ~  
$ cd mysql-docker  
[node1] (local) root@192.168.0.8 ~/mysql-docker  
$ touch Dockerfile  
[node1] (local) root@192.168.0.8 ~/mysql-docker  
$ nano Dockerfile  
bash: nano: command not found  
[node1] (local) root@192.168.0.8 ~/mysql-docker  
$ vi Dockerfile  
[node1] (local) root@192.168.0.8 ~/mysql-docker  
$ touch init.sql  
[node1] (local) root@192.168.0.8 ~/mysql-docker  
$ vi init.sql  
[node1] (local) root@192.168.0.8 ~/mysql-docker  
$ docker build -t mysql-alunos .  
[+] Building 23.4s (7/7) FINISHED
```

```
$ docker build -t mysql-alunos .
[+] Building 23.4s (7/7) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile              0.0s
=> => transferring dockerfile: 176B                             0.0s
=> [internal] load metadata for docker.io/library/mysql:latest  2.1s
=> [internal] load .dockerignore                                0.0s
=> => transferring context: 2B                                    0.0s
=> [internal] load build context                                0.0s
=> => transferring context: 329B                                   0.0s
=> [1/2] FROM docker.io/library/mysql:latest@sha256:0255b469f0135a0236d672d60e3154ae2f4538b1467 21.1s
=> => resolve docker.io/library/mysql:latest@sha256:0255b469f0135a0236d672d60e3154ae2f4538b1467 0.0s
=> => sha256:8689c5f93826d949b43a4b7613f9216499694a3082bdc372094a2910082aa391 2.86kB / 2.86kB 0.0s
=> => sha256:56a8c14e14044b8ec7fffb4dd165c8dbe10d4c6ba3d9e754f0c906f52a0b5b4fb 6.52kB / 6.52kB 0.0s
=> => sha256:0255b469f0135a0236d672d60e3154ae2f4538b146744966d96440318cc822c6 2.62kB / 2.62kB 0.0s
=> => sha256:2c0a233485c3a7b6cab556a9a9c2916ca9a3afc8c46097ddfbe0af4fe120a50c 49.10MB / 49.10MB 1.0s
=> => sha256:cb5a6a8519b2b1d789b65fe763238531f3456522a13e5e8d6a644ccd9a95e527 885B / 885B 0.4s
=> => sha256:570d30cf82c5e8cd5bde2c25ddc2d5c25e9a38583f09f4411ffce694e16943 983.00kB / 983.00kB 0.4s
=> => sha256:a841bfff36f3c6803a0ea0497a2f21ed75fcc7614e9f0fdaf1130861d2052f3f1 6.90MB / 6.90MB 0.8s
=> => sha256:80ba30c577823a1075c778315bd221b405236429f27ad62b3d359d59e55c640d 2.60kB / 2.60kB 0.7s
=> => sha256:5e49e1f26961569a4b36de1ed482bccbb9249ca5b29d008222bb19d302cca30 340B / 340B 1.1s
=> => sha256:ced670fc7f1c77db06886a4b52dc4051543eac07c0f58d481c62f0993e93e365 48.21MB / 48.21MB 1.9s
=> => sha256:0b9dc7ad7f03decfdeb7260a38115fa61c9d00a8bcaadc00fd9034290cbc8b7f 325B / 325B 1.3s
=> => extracting sha256:2c0a233485c3a7b6cab556a9a9c2916ca9a3afc8c46097ddfbe0af4fe120a50c 2.9s
=> => sha256:cd0d5df9937b5e0d0634c76339346dfac141def2ee29367bf9858ea335714477 69.09MB / 69.09MB 2.3s
=> => sha256:1f87d67b89c62ad5e3961e9f730df83c0628049afd874dcfb0df2cd3e7dc5faf 5.32kB / 5.32kB 1.7s
=> => extracting sha256:cb5a6a8519b2b1d789b65fe763238531f3456522a13e5e8d6a644ccd9a95e527 0.0s
=> => extracting sha256:570d30cf82c5e8cd5bde2c25ddc2d5c25e9a38583f09f4411ffce694e1694355 0.1s
=> => extracting sha256:a841bfff36f3c6803a0ea0497a2f21ed75fcc7614e9f0fdaf1130861d2052f3f1 0.5s
=> => extracting sha256:80ba30c577823a1075c778315bd221b405236429f27ad62b3d359d59e55c640d 0.0s
=> => extracting sha256:5e49e1f26961569a4b36de1ed482bccbb9249ca5b29d008222bb19d302cca30 0.0s
=> => extracting sha256:ced670fc7f1c77db06886a4b52dc4051543eac07c0f58d481c62f0993e93e365 2.1s
=> => extracting sha256:0b9dc7ad7f03decfdeb7260a38115fa61c9d00a8bcaadc00fd9034290cbc8b7f 0.0s
=> => extracting sha256:cd0d5df9937b5e0d0634c76339346dfac141def2ee29367bf9858ea335714477 13.9s
=> => extracting sha256:1f87d67b89c62ad5e3961e9f730df83c0628049afd874dcfb0df2cd3e7dc5faf 0.0s
=> [2/2] COPY init.sql /docker-entrypoint-initdb.d/             0.1s
=> exporting to image                                             0.0s
```

```
1 warning found (use docker --debug to expand):
- SecretsUsedInArgOrEnv: Do not use ARG or ENV instructions for sensitive data (ENV "MYSQL_ROOT_PASSWORD") (line 2)
[node1] (local) root@192.168.0.8 ~/mysql-docker
$ docker run --name mysql-alunos -d -p 3306:3306 mysql-alunos
8a41217874dc37527993740c5c72cd6cb57a88284e817012146a21551bca0e9f
[node1] (local) root@192.168.0.8 ~/mysql-docker
$ docker exec -it mysql-alunos mysql -u root -p
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)
[node1] (local) root@192.168.0.8 ~/mysql-docker
$ docker exec -it mysql-alunos mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 9.1.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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.

mysql> show databases;
```

```

mysql> show databases;
+-----+
| Database |
+-----+
| BD_ALUNOS |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.01 sec)

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)

mysql> cat Dockerfile
-> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'cat Dockerfile' at line 1
mysql> .quit
-> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '.quit' at line 1
mysql> exit
Bye
[node1] (local) root@192.168.0.8 ~/mysql-docker
$ cat Dockerfile
FROM mysql:latest
ENV MYSQL_ROOT_PASSWORD=kid_aluno
ENV MYSQL_DATABASE=BD_ALUNOS
COPY init.sql /docker-entrypoint-initdb.d/
EXPOSE 3306

[node1] (local) root@192.168.0.8 ~/mysql-docker
$ cat init_sql
cat: can't open 'init_sql': No such file or directory
[node1] (local) root@192.168.0.8 ~/mysql-docker
$ cat init.sql
CREATE TABLE IF NOT EXISTS TB_ALUNOS (
  id INT AUTO INCREMENT PRIMARY KEY,
  nome VARCHAR(100) NOT NULL,
  idade INT NOT NULL
);
CREATE TABLE IF NOT EXISTS TB_PROFESSORES (
  id INT AUTO INCREMENT PRIMARY KEY,

```