

FIAP GRADUAÇÃO

SISTEMAS DE INFORMAÇÃO

MICROSERVICE AND WEB ENGINEERING

Prof^a. Aparecida Castello Rosa
profaparecida.rosa@fiap.com.br

■ Agenda

- Terminar *Dockerizando* projeto produto-mvc
- Docker Compose
- Docker Compose – MySql
- Perfil dev

Objetivos

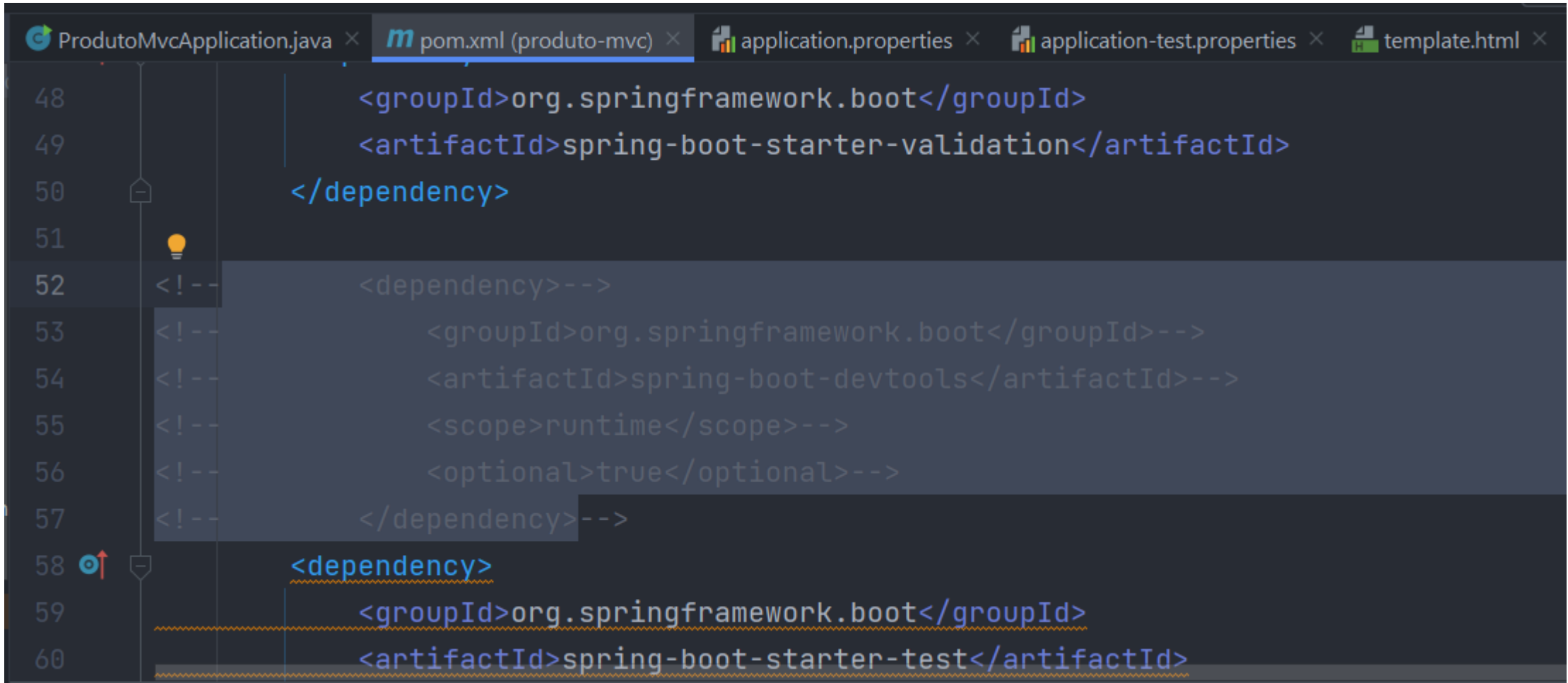
- Criar *Dockerfile* para projeto produto-mvc.
- Conhecer docker-compose
- docker-compose para MySql
- Criar perfil dev para projeto produto-mvc
- Projeto produto-mvc com MySql

Dockerizando projeto produto-mvc

Exercício

- Baixar projeto produto-mvc – aula 23

Alterar pom.xml



```
48     <groupId>org.springframework.boot</groupId>
49     <artifactId>spring-boot-starter-validation</artifactId>
50 </dependency>
51
52 <!-- <dependency>-->
53 <!-- <groupId>org.springframework.boot</groupId>-->
54 <!-- <artifactId>spring-boot-devtools</artifactId>-->
55 <!-- <scope>runtime</scope>-->
56 <!-- <optional>true</optional>-->
57 <!-- </dependency>-->
58 <dependency>
59     <groupId>org.springframework.boot</groupId>
60     <artifactId>spring-boot-starter-test</artifactId>
```

Dockerfile

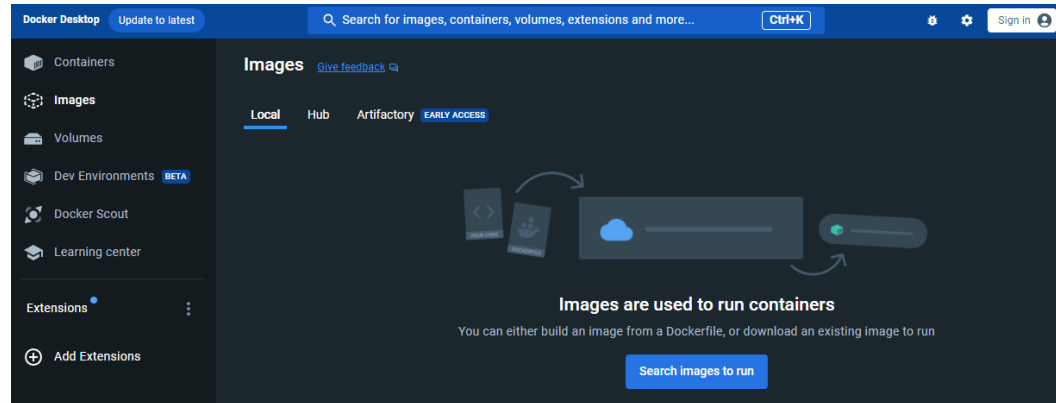
```
Dockerfile x
1  # Define your base image
2  >> FROM openjdk:17-jdk-alpine
3
4  # Define the working directory by using the WORKDIR instruction.
5  WORKDIR /app
6
7  # Copy both the Maven wrapper script and your project's pom.xml file
8  # into the current working directory /app within the Docker container.
9  COPY .mvn/ .mvn
10 COPY mvnw pom.xml ./
11
12 # Execute a command within the container.
13 # Download all dependencies for your project without building the final JAR file.
14 RUN ./mvnw dependency:go-offline
15
16 # Copy the src directory from your project on the host machine
17 # to the /app directory within the container.
18 COPY src ./src
19
20 EXPOSE 8080
21
22 # Set the default command to be executed when the container starts.
23 CMD ["/mvnw", "spring-boot:run"]
```

<https://docs.docker.com/guides/docker-concepts/building-images/multi-stage-builds/>

Docker

```
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
------------	-----	----------	---------	------



```
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> ls
```

Diretório: C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26

Mode		LastWriteTime	Length	Name
d----		14/05/2024 11:47		.idea
d----		14/05/2024 11:44		.mvn
d----		14/05/2024 11:44		src
d----		14/05/2024 11:44		target
-a----		11/02/2024 15:22	395	.gitignore
-a----		14/05/2024 14:25	819	<u>Dockerfile</u>
-a----		11/02/2024 15:22	1396	HELP.md
-a----		11/02/2024 15:22	11290	mvnw
-a----		11/02/2024 15:22	7592	mvnw.cmd
-a----		14/05/2024 11:50	2826	pom.xml

```
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> |
```

AULA 27 – Dockerfile produto-mvc - Docker Compose - MySql

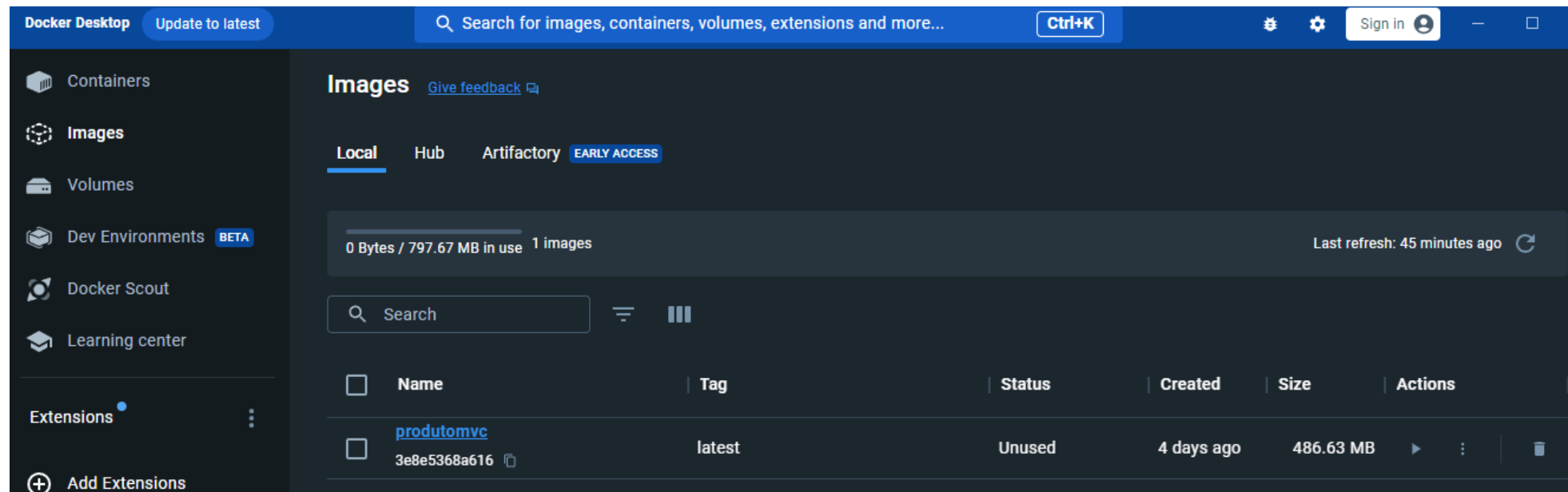
Gerar a *imagem* do projeto produto-mvc

```
$ docker build -t produtomvc .
```

```
Windows PowerShell
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker build -t produtomvc .
[+] Building 2.7s (11/11) FINISHED
=> [internal] load build definition from Dockerfile                                0.0s
=> => transferring dockerfile: 858B                                              0.0s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                  0.0s
=> [internal] load metadata for docker.io/library/openjdk:17-jdk-alpine        2.4s
=> [1/6] FROM docker.io/library/openjdk:17-jdk-alpine@sha256:4b6abae565492dbe9e7a894137c966a7485154238902f2f25e9dbd9784383d81 0.1s
=> => resolve docker.io/library/openjdk:17-jdk-alpine@sha256:4b6abae565492dbe9e7a894137c966a7485154238902f2f25e9dbd9784383d81 0.1s
=> [internal] load build context                                                0.1s
=> => transferring context: 8.05kB                                              0.0s
=> CACHED [2/6] WORKDIR /app                                                    0.0s
=> CACHED [3/6] COPY .mvn/ .mvn                                                 0.0s
=> CACHED [4/6] COPY mvnw pom.xml ./                                           0.0s
=> CACHED [5/6] RUN ./mvnw dependency:go-offline                               0.0s
=> CACHED [6/6] COPY src ./src                                                  0.0s
=> exporting to image                                                          0.0s
=> => exporting layers                                                         0.0s
=> => writing image sha256:3e8e5368a616fd9e68a5e1220b91b39538c3428369b7ee812b624c3635adc4ea 0.0s
=> => naming to docker.io/library/produtomvc                                  0.0s

What's Next?
 1. Sign in to your Docker account → docker login
 2. View a summary of image vulnerabilities and recommendations → docker scout quickview
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26>
```

```
Windows PowerShell
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26>
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker images
REPOSITORY    TAG       IMAGE ID      CREATED       SIZE
producomvc    latest   3e8e5368a616  4 days ago   487MB
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> |
```



Construir o *container* da *imagem*

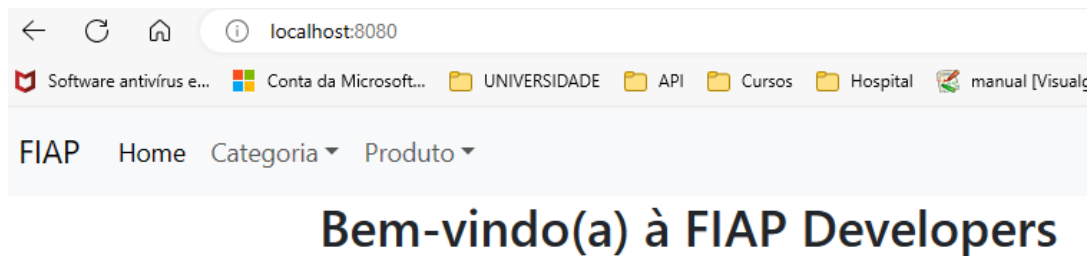
```
$ docker run -dp 8080:8080 produtomvc:latest
```

Listar *containers*

```
$ docker ps
```

```
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26>
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker run -dp 8080:8080 produtomvc:latest
6c53379e4c3783bb53de615b4b29ca0ed69dd15c791c302e52b983530f5597d2
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker ps
CONTAINER ID   IMAGE             COMMAND                  CREATED        STATUS        PORTS                               NAMES
6c53379e4c37   produtomvc:latest "/mvnw spring-boot:..." 5 seconds ago  Up 4 seconds  0.0.0.0:8080->8080/tcp             awesome_ride
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> |
```

Testar aplicação



Cadastro de Categoria

Nome:

Teste

Salvar

Lista de Produtos

Nome	Descrição	Categoria	Lojas	Valor	Ações	
Mouse Microsoft	Mouse sem fio	Mouse	Americanas , Magazine Luiza	250.0	Excluir	Editar
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	Smartphone	Americanas , Submarino	1799.0	Excluir	Editar
Smart TV	Smart TV LG LED 65 polegadas	Smart TV	Fast Shop	3999.0	Excluir	Editar
Teclado Microsof	Teclado sem fio	Teclado	Fast Shop	278.5	Excluir	Editar
Apple iPhone 15	Apple iPhone 15, 128G, Preto	Smartphone	Fast Shop	4999.0	Excluir	Editar

Lista de Categorias

Nome	Ações	
Smartphone	Excluir	Editar
Smart TV	Excluir	Editar
Notebook	Excluir	Editar
Tablet	Excluir	Editar
Mouse	Excluir	Editar
Teclado	Excluir	Editar
Teste	Excluir	Editar

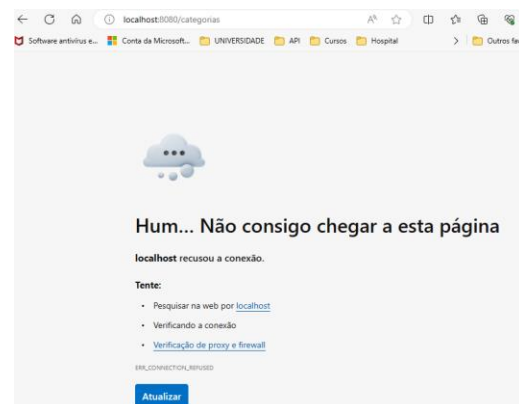
■ Testar aplicação

Vamos parar o *container*

```
$ docker stop 6c5
```

Testar a aplicação.

```
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
6c53379e4c37   produtomvc:latest  "./mvnw spring-boot:..."  5 seconds ago  Up 4 seconds  0.0.0.0:8080->8080/tcp    awesome_ride
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker stop 6c5
6c5
```



Vamos iniciar o **container** e testar a aplicação.

```
# listar todos os containers
docker ps -a
# iniciar o container
docker start 6c5
#listar os containers em execução
docker ps
```

```
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS          PORTS          NAMES
6c53379e4c37   produtomvc:latest  "/mvnw spring-boot:..."  5 minutes ago  Exited (143) 2 minutes ago          awesome_ride
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker start 6c5
6c5
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS          PORTS          NAMES
6c53379e4c37   produtomvc:latest  "/mvnw spring-boot:..."  5 minutes ago  Up 4 seconds   0.0.0.0:8080->8080/tcp  awesome_ride
PS C:\Cida\ws_fiap_2024_01_intellij\Aula_26_ProdutoMVC_Dockerfile_codigo\produto-mvc_aula_26> |
```

Testar aplicação

FIAP Home Categoria ▾ Produto ▾

Lista de Categorias

Nome	Ações
Smartphone	Excluir Editar
Smart TV	Excluir Editar
Notebook	Excluir Editar
Tablet	Excluir Editar
Mouse	Excluir Editar
Teclado	Excluir Editar



Docker Compose

Exercício

I Docker Compose

O **Docker Compose** é uma ferramenta poderosa para orquestrar e gerenciar aplicativos compostos por vários *containers* Docker.

Por exemplo, vamos imaginar que temos que executar diversos *containers* interligados manualmente, um por um, definindo configurações e links entre eles. Com o **Docker Compose** podemos simplificar e automatizar esse processo.

O **Docker Compose** é uma forma escrever em um único arquivo todos os detalhes do ambiente de sua aplicação.

Até agora, usamos o *dockerfile* apenas para criar imagens, seja da aplicação, do BD ou do *webserver*, mas sempre de forma unitária, pois temos um *dockerfile* para cada *tipo* de *container*.

Com o **Docker Compose** estamos nos referindo sobre o ambiente inteiro. Por exemplo, no **Docker Compose** definimos quais os *services* que desejamos criar e quais as características de cada *service* (quantidade de *containers* debaixo daquele *service*, *volumes*, *network*, etc.).

A sintaxe dos *compose files* é o YML.

<https://docs.docker.com/compose/>

Docker Compose

Funcionalidades:

- **Definição de serviços com YAML:** arquivo **YAML** que define a configuração de cada **container**, incluindo *imagem*, *volumes*, *portas*, *redes* e *dependências*.
- **Links automáticos:** O **Compose** lida automaticamente com a criação de links entre **containers**, facilitando a comunicação entre eles.
- **Monitoramento de logs:** Visualizar os *logs* de cada **container** em tempo real ou salvar em arquivos para análise posterior.
- **Escalabilidade:** Aumentar ou diminuir o número de instâncias de um **container** com facilidade, adaptando o aplicativo à demanda.
- **Desenvolvimento e testes simplificados:** Ideal para ambientes de desenvolvimento e testes, no qual é crucial ter um ambiente replicável e consistente.
- **Implantação em produção:** O **Compose** também pode ser utilizado para implantar aplicativos em produção, gerenciando múltiplos **containers** em um servidor.

I Docker Compose

Benefícios:

- **Agilidade:** Automatizar a criação e o gerenciamento dos **containers**, economizando tempo e esforço.
- **Simplicidade:** Definição dos serviços em um único arquivo **YAML**, tornando a configuração e o provisionamento mais intuitivos.
- **Escalabilidade:** Adaptar o aplicativo às mudanças na demanda com facilidade, escalando verticalmente ou horizontalmente.
- **Consistência:** Garantir ambientes replicáveis em diferentes máquinas, para desenvolvimento, testes e produção.
- **Organização:** Manter os **containers** e seus ambientes organizados e bem documentados.

Docker Compose - comandos

Subir **container** com **compose** (com o arquivo `docker-compose.yml` no local onde será executado)

```
docker-compose up
```

Subir **container** com **compose** em *modo deamon* (com o arquivo `docker-compose.yml` no local onde será executado)

```
docker-compose up -d
```

Listar serviços do **compose** (com o arquivo `docker-compose.yml` no local onde será executado)

```
docker-compose ps
```

I Docker Compose - comandos

Parar serviços do *compose*

```
docker-compose stop
```

Iniciar serviços do *compose*

```
docker-compose start
```

Remover serviços do *compose*

```
docker-compose down
```

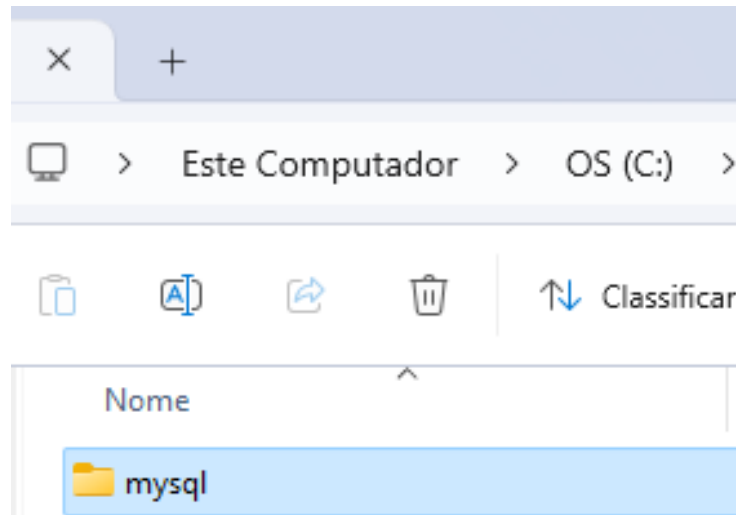
I Docker Compose

```
Windows PowerShell
Windows PowerShell
PS C:\> docker -v
Docker version 24.0.6, build ed223bc
PS C:\> docker-compose -v
Docker Compose version v2.23.0-desktop.1
PS C:\> docker compose version
Docker Compose version v2.23.0-desktop.1
PS C:\> |
```


Docker Compose para MySql

Criar uma pasta para o arquivo do Docker Compose para o MySql.

Criar o arquivo `docker-compose.yml` dentro da pasta.



Docker Compose para MySql

```
C: > mysql > 🐳 docker-compose.yml
1  version: '3.8'
2
3  services:
4    mysql:
5      container_name: mysql_container
6      image: mysql:8.0
7      command: --default-authentication-plugin=mysql_native_password
8      # restart: always
9      environment:
10     | MYSQL_ROOT_PASSWORD: root
11     ports:
12     | - "3306:3306"
13     volumes:
14     | - mysql-data:/var/lib/mysql
15
16  volumes:
17    mysql-data:
```

■ Docker Compose para MySql

```
Windows PowerShell
Windows PowerShell x + v
PS C:\mysql> ls

Diretório: C:\mysql

Mode                LastWriteTime         Length Name
----                -
-a-----         18/05/2024   20:29           348 docker-compose.yml

PS C:\mysql> |
```

Docker Compose para MySql

```
PS C:\mysql>
PS C:\mysql> docker-compose up -d
[+] Running 12/12
  ✓mysqlldb 11 layers [#####] 0B/0B Pulled
    ✓fcbdc4090331 Pull complete
    ✓a845363d0003 Pull complete
    ✓e8058c6f9bdd Pull complete
    ✓2ec357e5ab0c Pull complete
    ✓5afcf1382f34 Pull complete
    ✓7e6fb392497b Pull complete
    ✓0287ae8ee890 Pull complete
    ✓faa49449b97e Pull complete
    ✓ddd0836d7ff3 Pull complete
    ✓c25cb9740165 Pull complete
    ✓8d993d969e2b Pull complete
[+] Building 0.0s (0/0)
[+] Running 3/3
  ✓Network mysql_default Created
  ✓Volume "mysql_mysql-data" Created
  ✓Container mysql_container Started
PS C:\mysql> |
```

Docker Compose para MySql

```
Windows PowerShell
PS C:\mysql> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
NAME
de06e46d481b   mysql:8.0  "docker-entrypoint.s..." 3 minutes ago  Up 3 minutes  0.0.0.0:3306->3306/tcp, 33060/tcp
mysql_container
PS C:\mysql>
PS C:\mysql> docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
produtomvc    latest   3e8e5368a616   4 days ago    487MB
mysql         8.0      c5f0c876cdb1   2 weeks ago    567MB
PS C:\mysql> |
```

Docker Compose para MySql

```
Windows PowerShell
PS C:\mysql>
PS C:\mysql> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
NAMES
de06e46d481b   mysql:8.0     "docker-entrypoint.s..." 4 minutes ago  Up 4 minutes  0.0.0.0:3306->3306/tcp, 33060/tcp
mysql_container
PS C:\mysql> docker container exec -it mysql_container bash
bash-5.1# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.37 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> |
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.37 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;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| performance_schema      |
| sys                     |
+-----+
4 rows in set (0.00 sec)

mysql> |
```

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.00 sec)

mysql> exit
Bye
bash-5.1# exit
exit
PS C:\mysql> |
```



```
Windows PowerShell
PS C:\mysql> docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
de06e46d481b       mysql:8.0          "docker-entrypoint.s..." 10 minutes ago      Up 9 minutes        0.0.0.0:3306->3306/tcp, 33060/tcp    mysql_container
PS C:\mysql> docker stop de0
de0
PS C:\mysql> docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
PS C:\mysql> docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
de06e46d481b       mysql:8.0          "docker-entrypoint.s..." 10 minutes ago      Exited (0) 9 seconds ago                  mysql_container
6c53379e4c37       produtomvc:latest  "./mvnw spring-boot:..." 2 hours ago         Exited (143) 8 minutes ago                awesome_ride
PS C:\mysql>
```

Preparando o projeto produto-mvc para o MySQL

- Baixar projeto produto-mvc – aula 23

Adicionar dependência MySql

Dependencies

MySQL Driver SQL
MySQL JDBC driver.

demo.zip

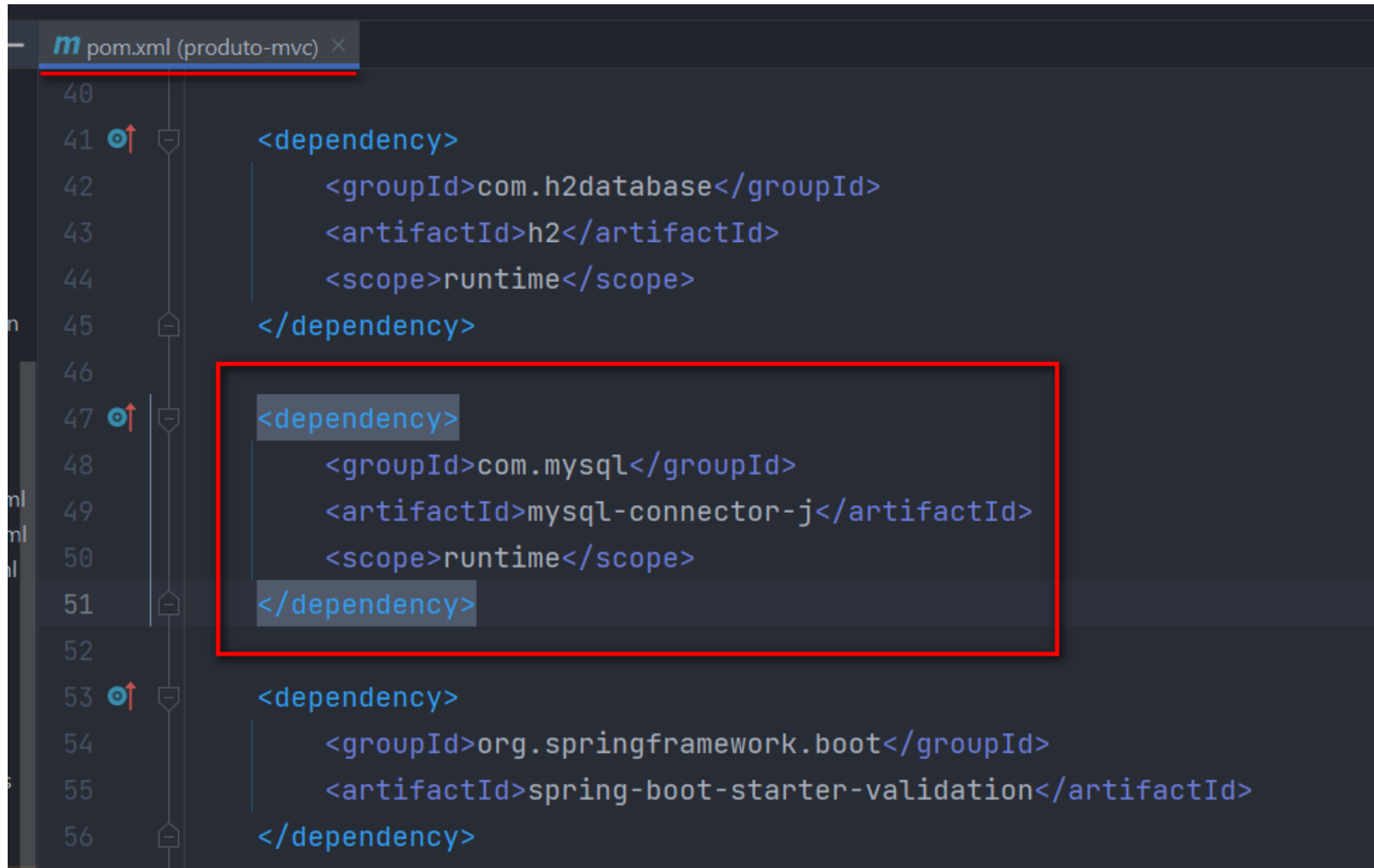
DOWNLOAD COPY

- .gitignore
- .mvn
- HELP.md
- mvnw
- mvnw.cmd
- pom.xml**
- src

```
20 <dependency>
21   <groupId>org.springframework.boot</groupId>
22   <artifactId>spring-boot-starter</artifactId>
23 </dependency>
24
25 <dependency>
26   <groupId>com.mysql</groupId>
27   <artifactId>mysql-connector-j</artifactId>
28   <scope>runtime</scope>
29 </dependency>
30 <dependency>
31   <groupId>org.springframework.boot</groupId>
32   <artifactId>spring-boot-starter-test</artifactId>
```

DOWNLOAD CTRL + ↵ CLOSE ESC

Alterar pom.xml

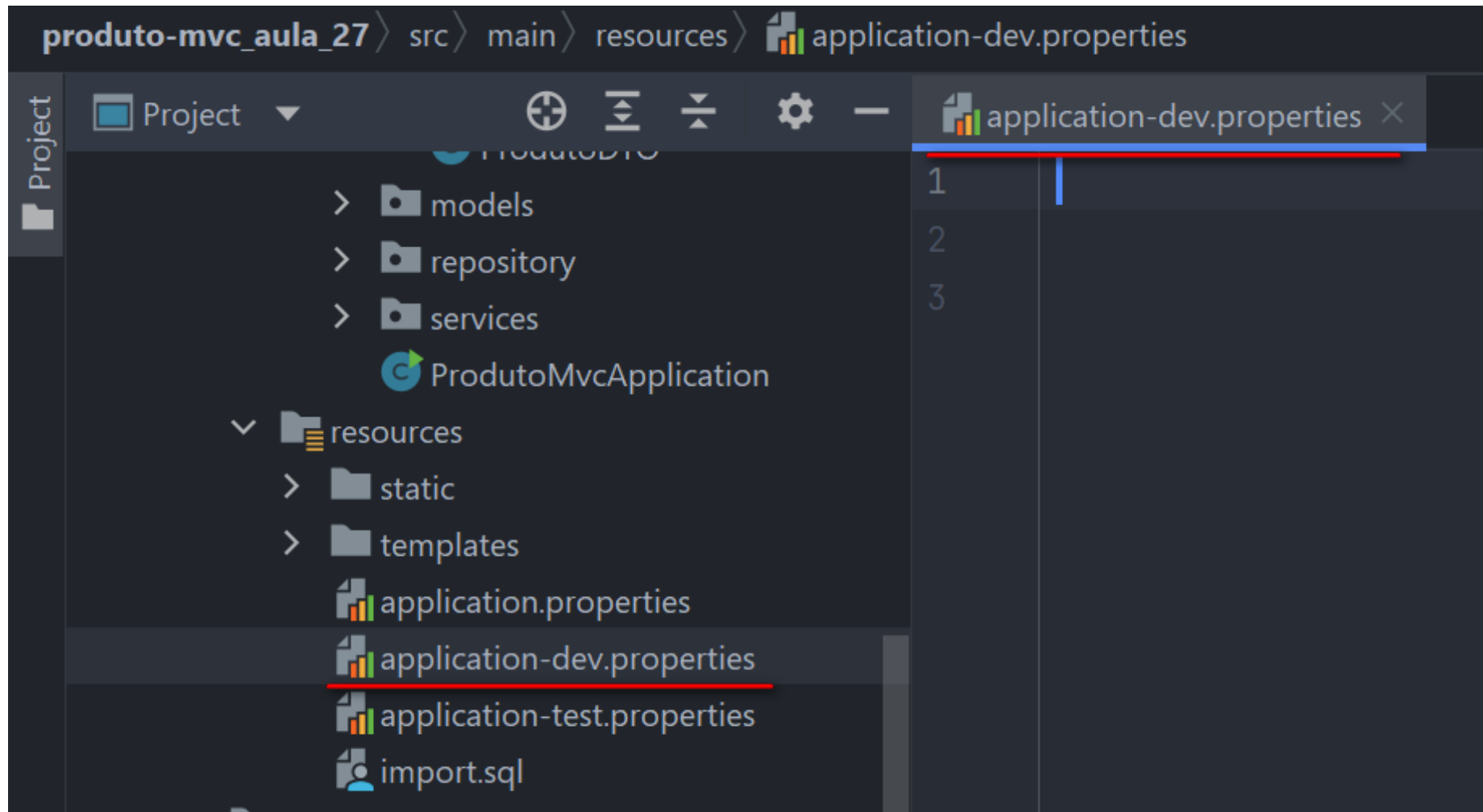


The screenshot shows a code editor with a file named 'pom.xml (produto-mvc)'. The XML content is as follows:

```
40
41 <dependency>
42     <groupId>com.h2database</groupId>
43     <artifactId>h2</artifactId>
44     <scope>runtime</scope>
45 </dependency>
46
47 <dependency>
48     <groupId>com.mysql</groupId>
49     <artifactId>mysql-connector-j</artifactId>
50     <scope>runtime</scope>
51 </dependency>
52
53 <dependency>
54     <groupId>org.springframework.boot</groupId>
55     <artifactId>spring-boot-starter-validation</artifactId>
56 </dependency>
```

The dependency block starting at line 47 (MySQL connector) is highlighted with a red rectangular box.

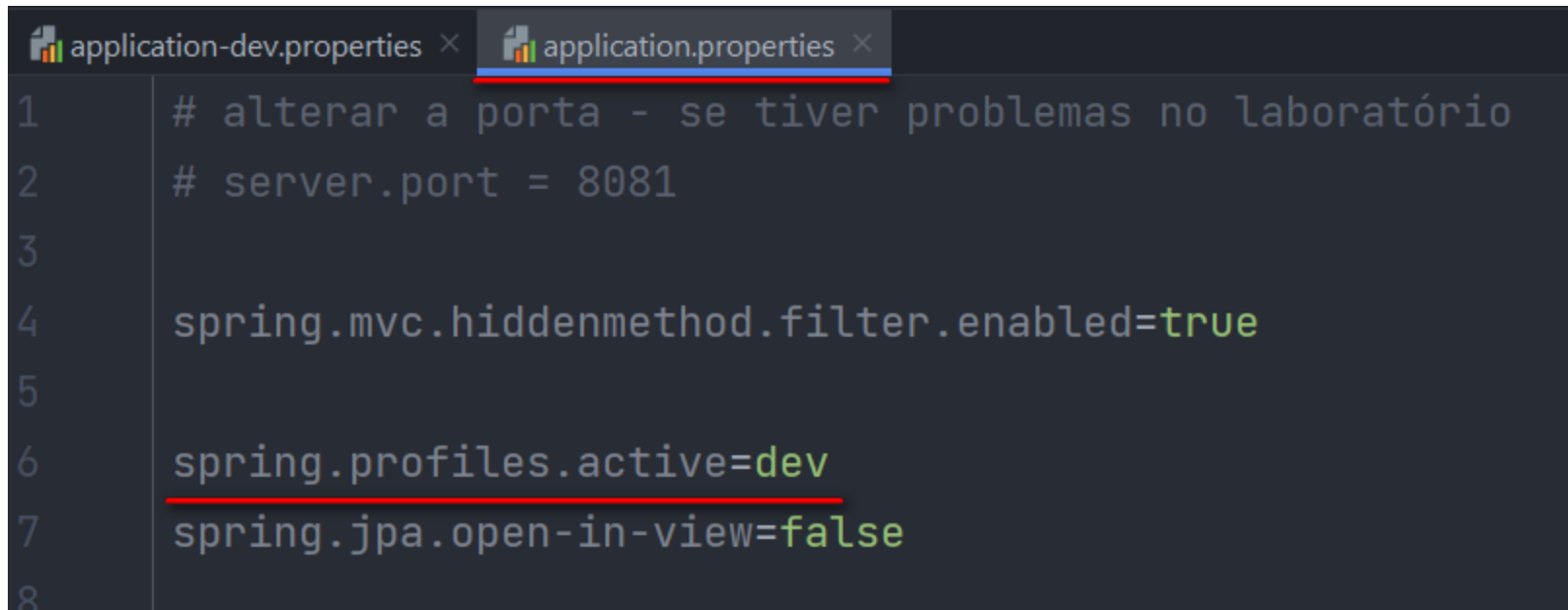
■ Criar o perfil dev - application-dev.properties



application-dev.properties

```
application-dev.properties x
1  # Homologação - MySQL Local
2
3  # Configurações para conexão com o banco
4  spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
5  spring.datasource.url=jdbc:mysql://localhost:3306/produtomvc?createDatabaseIfNotExist=true
6  spring.datasource.username=root
7  spring.datasource.password=root
8
9  spring.jpa.generate-ddl=true
10
11 # Informa ação DDL inicial do Hibernate
12 # create, update, create-drop, validate, none
13 spring.jpa.hibernate.ddl-auto=create
14
15 # Formata o sql exibido
16 spring.jpa.properties.hibernate.format_sql=true
17
18 # Exibe / oculta o SQL executado
19 spring.jpa.show-sql=true
```

Alterar application.properties



The screenshot shows a code editor with two tabs: 'application-dev.properties' and 'application.properties'. The 'application.properties' tab is active and contains the following content:

```
1 # alterar a porta - se tiver problemas no laboratório
2 # server.port = 8081
3
4 spring.mvc.hiddenmethod.filter.enabled=true
5
6 spring.profiles.active=dev
7 spring.jpa.open-in-view=false
8
```


Docker Compose

Docker Compose

```

PS C:\mysql> ls

Diretório: C:\mysql

Mode                LastWriteTime         Length Name
----                -
-a-----         18/05/2024   20:29           348 docker-compose.yml

PS C:\mysql> docker-compose ps
NAME          IMAGE          COMMAND          SERVICE    CREATED   STATUS    PORTS
PS C:\mysql> docker-compose ps -a
NAME          IMAGE          COMMAND          SERVICE    CREATED   STATUS    PORTS
mysql_container  mysql:8.0    "docker-entrypoint.s..."  mysqldb    19 hours ago  Exited (0) 19 hours ago
PS C:\mysql> docker-compose up -d
[+] Building 0.0s (0/0)
[+] Running 1/1
✓ Container mysql_container Started
PS C:\mysql> docker ps
CONTAINER ID   IMAGE          COMMAND          CREATED        STATUS        PORTS
NAMES
de06e46d481b  mysql:8.0    "docker-entrypoint.s..."  19 hours ago  Up 5 seconds  0.0.0.0:3306->3306/tcp, 33060/t
cp  mysql_container

```

MySQL - Docker

```
PS C:\mysql> docker container exec -it mysql_container bash
bash-5.1# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.37 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.

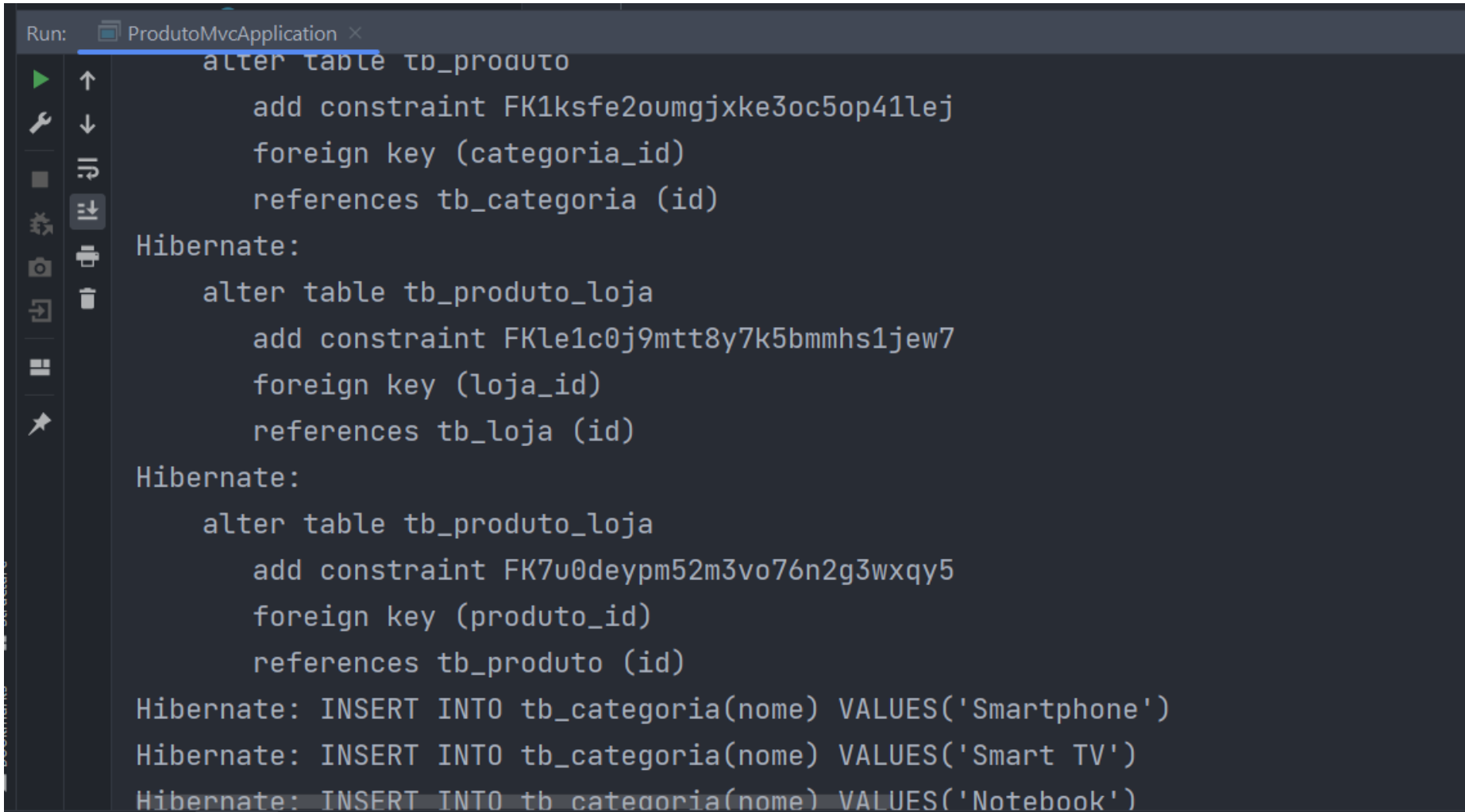
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
```

Database
information_schema
mysql
performance_schema
sys

```
4 rows in set (0.01 sec)
```

■ Executar a aplicação



```
Run: ProdutoMvcApplication x
alter table tb_produto
  add constraint FK1ksfe2oumgjxke3oc5op41lej
  foreign key (categoria_id)
  references tb_categoria (id)
Hibernate:
  alter table tb_produto_loja
    add constraint FKle1c0j9mtt8y7k5bmmhs1jew7
    foreign key (loja_id)
    references tb_loja (id)
Hibernate:
  alter table tb_produto_loja
    add constraint FK7u0deypm52m3vo76n2g3wxqy5
    foreign key (produto_id)
    references tb_produto (id)
Hibernate: INSERT INTO tb_categoria(nome) VALUES('Smartphone')
Hibernate: INSERT INTO tb_categoria(nome) VALUES('Smart TV')
Hibernate: INSERT INTO tb_categoria(nome) VALUES('Notebook')
```

```
Windows PowerShell
Windows PowerShell x + v

4 rows in set (0.01 sec)

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

mysql> use produtomvc
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_produtomvc |
+-----+
| tb_categoria |
| tb_loja |
| tb_produto |
| tb_produto_loja |
+-----+
4 rows in set (0.00 sec)
```

```
mysql> select * from tb_categoria;
+----+-----+
| id | nome |
+----+-----+
| 1 | Smartphone |
| 2 | Smart TV |
| 3 | Notebook |
| 4 | Tablet |
| 5 | Mouse |
| 6 | Teclado |
+----+-----+
6 rows in set (0.00 sec)

mysql> select * from tb_loja;
+----+-----+
| id | nome |
+----+-----+
| 1 | Americanas |
| 2 | Fast Shop |
| 3 | Magazine Luiza |
| 4 | Submarino |
+----+-----+
4 rows in set (0.00 sec)
```

```
mysql> select * from tb_produto;
```

valor	categoria_id	id	descricao	nome
250	5	1	Mouse sem fio	Mouse Microsoft
1799	1	2	Samsung Galaxy A54 5G	Smartphone Samsung Galaxy A54 5G
3999	2	3	Smart TV LG LED 65 polegadas	Smart TV
278.5	6	4	Teclado sem fio	Teclado Microsof
4999	1	5	Apple iPhone 15, 128G, Preto	Apple iPhone 15

```
5 rows in set (0.00 sec)
```

```
mysql> select * from tb_produto_loja;
```

loja_id	produto_id
1	1
3	1
1	2
4	2
2	3
2	4
2	5

```
7 rows in set (0.00 sec)
```

Testar aplicação

Cadastro de Categoria

Nome:

Teste MySql

Salvar

Lista de Categorias

Nome	Ações	
Smartphone	Excluir	Editar
Smart TV	Excluir	Editar
Notebook	Excluir	Editar
Tablet	Excluir	Editar
Mouse	Excluir	Editar
Teclado	Excluir	Editar
Teste MySql	Excluir	Editar

```
mysql> select * from tb_categoria;
```

id	nome
1	Smartphone
2	Smart TV
3	Notebook
4	Tablet
5	Mouse
6	Teclado
7	Teste MySQL

7 rows in set (0.00 sec)

Spring Boot Database Initialization

<https://docs.spring.io/spring-boot/docs/1.1.0.M1/reference/html/howto-database-initialization.html>

59. Database initialization

An SQL database can be initialized in different ways depending on what your stack is. Or of course you can do it manually as long as the database is a separate process.

59.1 Initialize a database using JPA

JPA has features for DDL generation, and these can be set up to run on startup against the database. This is controlled through two external properties:

`spring.jpa.generate-ddl` (boolean) switches the feature on and off and is vendor independent.

`spring.jpa.hibernate.ddl-auto` (enum) is a Hibernate feature that controls the behavior in a more fine-grained way. See below for more detail.

59.2 Initialize a database using Hibernate

You can set `spring.jpa.hibernate.ddl-auto` explicitly and the standard Hibernate property values are `none`, `validate`, `update`, `create-drop`.

Hibernate - DDL

O Hibernate permite a criação do banco de dados inteiro se alterarmos a propriedade `spring.jpa.hibernate.ddl-auto`, os valores aceitos são:

Propriedade	Espedificação
validate	Faz uma validação sempre que a aplicação inicia e verifica se o banco coincide com as suas classes de entidade. Valida o schema, não faz mudanças no banco.
update	Sempre que a aplicação iniciar ele verifica se suas classes entidades estão de acordo com o banco. Caso não esteja, vai ser feito um <i>update</i> no banco adicionando novas colunas na tabela dessa entidade. Vale lembrar que o update não remove ou renomeia colunas. Por isso, não é recomendado de forma alguma utilizar o Spring Data para gerenciar a evolução do seu banco, para isso existem ferramentas próprias, como, por exemplo, o <i>FlyWay</i> .
create	Sempre que iniciar sua aplicação o Spring Data vai apagar tudo e recriar novamente, ou seja, cria o schema e apaga a estrutura e dados anteriores .
create-drop	Semelhante ao create , mas sempre que a aplicação é parada ele apaga tudo que foi criado, ou seja, cria o schema e depois apaga tudo ao terminar a sessão.
none	Basicamente é nenhuma das opções acima, ou seja, significa que você não quer que o Spring Data faça alterações no seu banco. Essa opção é a recomendada para produção , para que não ocorra problemas do Spring modificar seu banco.

OBS.: Não colocar **create**, **create-drop** e **update** em **PRODUÇÃO !!!**

Alterar application-dev.properties

```
application-dev.properties x application.properties x
1  # Homologação - MySQL Local
2
3  # Configurações para conexão com o banco
4  spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
5  spring.datasource.url=jdbc:mysql://localhost:3306/produtomvc?createDatabaseIfNotExist=true
6  spring.datasource.username=root
7  spring.datasource.password=root
8
9  spring.jpa.generate-ddl=true
10
11  # Informa ação DDL inicial do Hibernate
12  # create, update, create-drop, validate, none
13  spring.jpa.hibernate.ddl-auto=none
14
15  # Formata o sql exibido
16  spring.jpa.properties.hibernate.format_sql=true
17
18  # Exibe / oculta o SQL executado
19  spring.jpa.show-sql=true
```

Executar aplicação e testar

Categoria ▾ Produto ▾

Lista de Categorias

Nome	Ações	
Smartphone	Excluir	Editar
Smart TV	Excluir	Editar
Notebook	Excluir	Editar
Tablet	Excluir	Editar
Mouse	Excluir	Editar
Teclado	Excluir	Editar
Teste MySql	Excluir	Editar

■ Executar aplicação e testar

Cadastro de Produtos

Nome:

Descrição:

Categoria:

Lojas:
Fast Shop
Magazine Luiza
Submarino

Valor:

Salvar

■ Executar aplicação e testar

Lista de Produtos

Nome	Descrição	Categoria	Lojas	Valor	Ações	
Mouse Microsoft	Mouse sem fio	Mouse	Americanas , Magazine Luiza	250.0	Excluir	Editar
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	Smartphone	Americanas , Submarino	1799.0	Excluir	Editar
Smart TV	Smart TV LG LED 65 polegadas	Smart TV	Fast Shop	3999.0	Excluir	Editar
Teclado Microsof	Teclado sem fio	Teclado	Fast Shop	278.5	Excluir	Editar
Apple iPhone 15	Apple iPhone 15, 128G, Preto	Smartphone	Fast Shop	4999.0	Excluir	Editar
Teste	Teste MySql	Teste MySql	Submarino	23.0	Excluir	Editar

Docker

Testes

- 1 - Parar a aplicação.
- 2 - Remover o **container** do **MySQL**.
- 3 - Executar o `docker-compose` novamente.
- 4 - Subir o **container**.
- 5 - Executar a aplicação e verificar se os dados continuam persistidos no DB.

```
mysql> exit
Bye
bash-5.1# exit
exit
PS C:\mysql>
```



```
Windows PowerShell
PS C:\mysql> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
NAMES
de06e46d481b   mysql:8.0      "docker-entrypoint.s..." 20 hours ago   Up 41 minutes  0.0.0.0:3306->3306/tcp, 33060/
tcp           mysql_container
PS C:\mysql> docker rm -f de0
de0
PS C:\mysql> docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
NAMES
6c53379e4c37   produtomvc:latest  "./mvnw spring-boot:..." 22 hours ago   Exited (143) 20 hours ago
awesome_ride
PS C:\mysql> |
```

```
Windows PowerShell
PS C:\mysql> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
de06e46d481b   mysql:8.0      "docker-entrypoint.s..." 20 hours ago   Up 41 minutes  0.0.0.0:3306->3306/tcp, 33060/
tcp           mysql_container
PS C:\mysql> docker rm -f de0
de0
PS C:\mysql> docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
6c53379e4c37   produtomvc:latest  "./mvnw spring-boot:..." 22 hours ago   Exited (143) 20 hours ago
awesome_ride
PS C:\mysql> |
```

```
Windows PowerShell
PS C:\mysql> ls

Diretório: C:\mysql

Mode                LastWriteTime         Length Name
----                -
-a-----         18/05/2024   20:29           348 docker-compose.yml

PS C:\mysql> docker-compose up -d
[+] Building 0.0s (0/0)                                docker:default
[+] Running 1/1
✓ Container mysql_container Started                      0.1s
PS C:\mysql> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS
ba66df2a1a3b   mysql:8.0 "docker-entrypoint.s..." 17 seconds ago Up 17 seconds 0.0.0.0:3306->3306/tcp, 3306
0/tcp        mysql_container
PS C:\mysql> |
```

```
PS C:\mysql> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
ba66df2a1a3b   mysql:8.0      "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:3306->3306/tcp, 33060/tcp
mysql_container
PS C:\mysql> docker container exec -it mysql_container bash
bash-5.1# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 8.0.37 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;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| produtomvc |
| sys |
+-----+
5 rows in set (0.00 sec)

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

```
mysql> use produtomvc
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_produtomvc |
+-----+
| tb_categoria          |
| tb_loja               |
| tb_produto            |
| tb_produto_loja       |
+-----+
4 rows in set (0.00 sec)

mysql> select * from tb_categoria;
+----+-----+
| id | nome          |
+----+-----+
| 1  | Smartphone    |
| 2  | Smart TV     |
| 3  | Notebook     |
| 4  | Tablet       |
| 5  | Mouse        |
| 6  | Teclado      |
| 8  | Teste MySQL  |
+----+-----+
7 rows in set (0.00 sec)
```

Testar aplicação

Lista de Categorias

Nome

Smartphone

Smart TV

Notebook

Tablet

Mouse

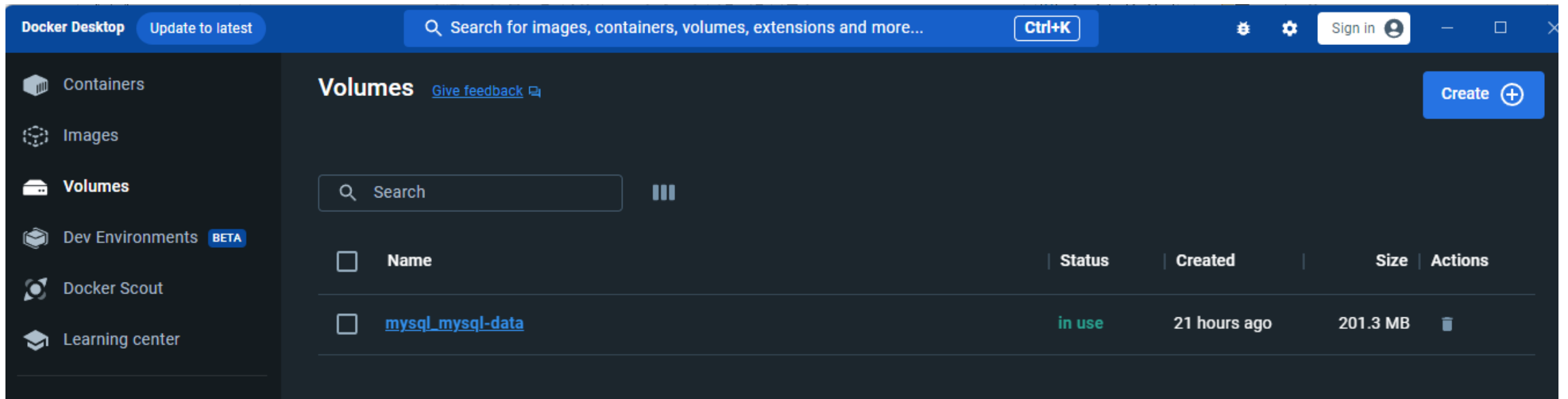
Teclado

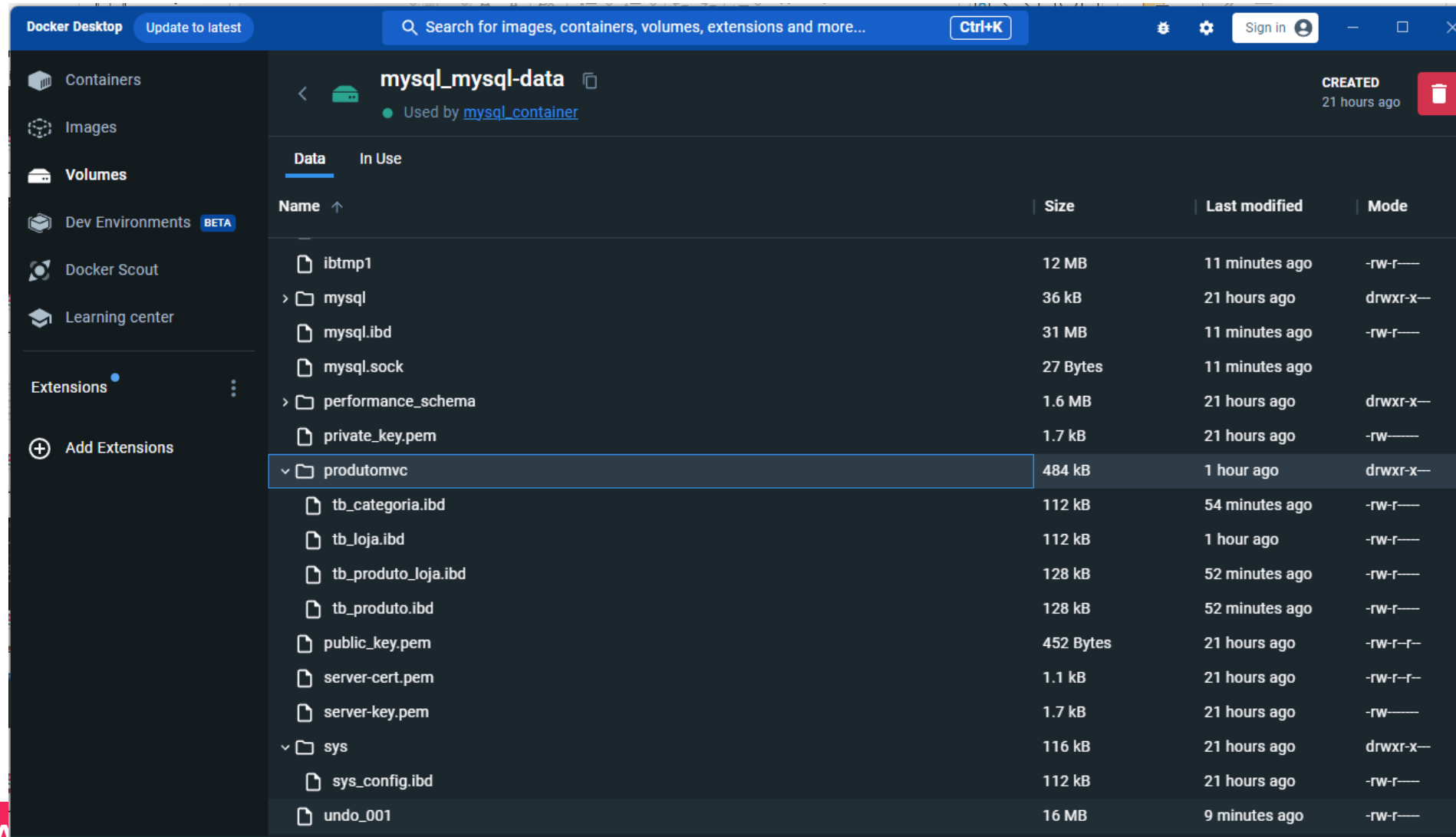
Teste MySql

Lista de Produtos

Nome	Descrição	Categoria	Lojas	Valor	Ações	
Mouse Microsoft	Mouse sem fio	Mouse	Americanas , Magazine Luiza	250.0	Excluir	Editar
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	Smartphone	Americanas , Submarino	1799.0	Excluir	Editar
Smart TV	Smart TV LG LED 65 polegadas	Smart TV	Fast Shop	3999.0	Excluir	Editar
Teclado Microsof	Teclado sem fio	Teclado	Fast Shop	278.5	Excluir	Editar
Apple iPhone 15	Apple iPhone 15, 128G, Preto	Smartphone	Fast Shop	4999.0	Excluir	Editar
Teste	Teste MySql	Teste MySql	Submarino	23.0	Excluir	Editar

```
Windows PowerShell
PS C:\mysql> docker volume ls
DRIVER      VOLUME NAME
local       mysql_mysql-data
PS C:\mysql> docker volume inspect mysql_mysql-data
[
  {
    "CreatedAt": "2024-05-18T23:32:27Z",
    "Driver": "local",
    "Labels": {
      "com.docker.compose.project": "mysql",
      "com.docker.compose.version": "2.23.0",
      "com.docker.compose.volume": "mysql-data"
    },
    "Mountpoint": "/var/lib/docker/volumes/mysql_mysql-data/_data",
    "Name": "mysql_mysql-data",
    "Options": null,
    "Scope": "local"
  }
]
PS C:\mysql> |
```







Copyright © 2024
Prof^a. Aparecida de Fátima Castello Rosa

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).