

Luccas Zulliane Marchetti da Silva

Camila Lima de Oliveira

Alexandra Evangelista

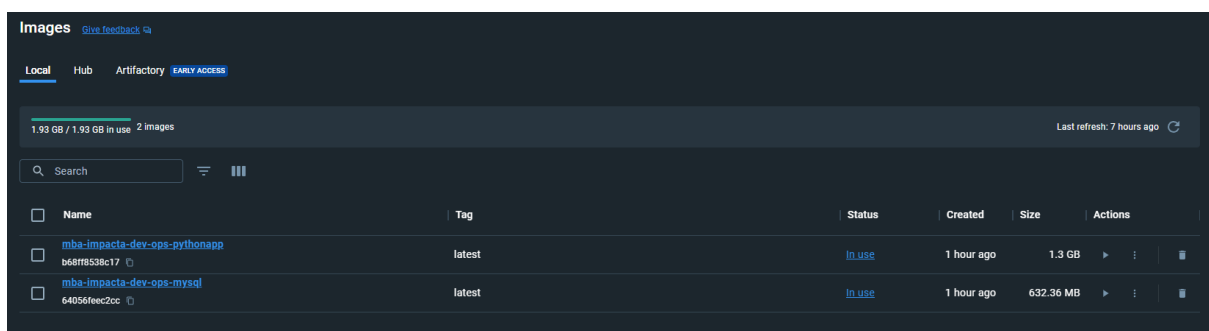
Felipe Cannas

Diego Carvalho

Link do repositório: <https://github.com/luccas-zulliane-impacta/mba-impacta-dev-ops.git>

Execução do docker-compose

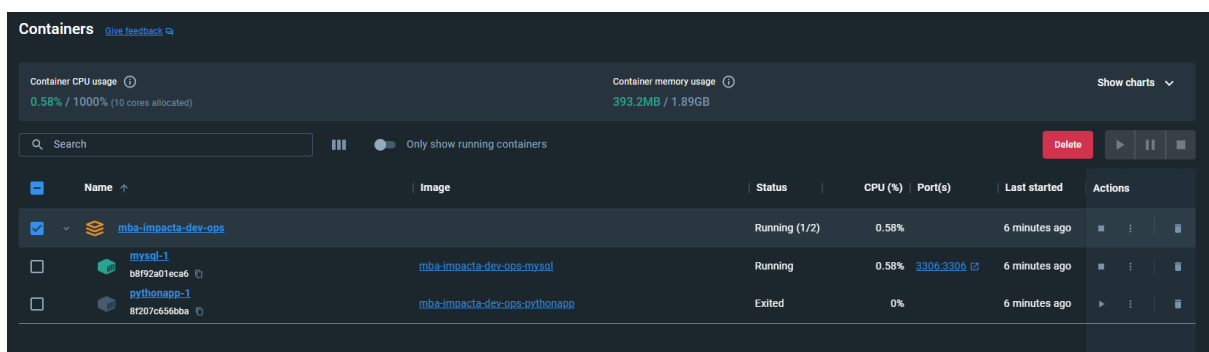
A execução do arquivo docker-compose permite verificar as seguintes imagens:



Name	Tag	Status	Created	Size	Actions
mba-impacta-dev-ops-pythonapp b68f8538c17	latest	In use	1 hour ago	1.3 GB	▶ ⋮ 🗑
mba-impacta-dev-ops-mysql 64056fec2cc	latest	In use	1 hour ago	632.36 MB	▶ ⋮ 🗑

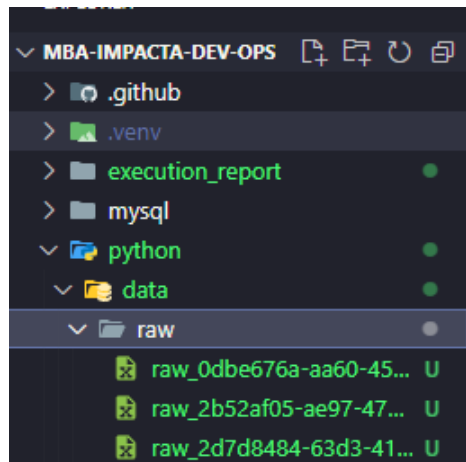
Há, portanto, a imagem da aplicação em python e a imagem do microserviço do banco de dados relacional em MySQL. As imagens foram criadas a partir dos arquivos Dockerfile específicos.

Os contêineres em execução podem ser verificados na imagem que segue:



Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
mba-impacta-dev-ops	mba-impacta-dev-ops	Running (1/2)	0.58%		6 minutes ago	■ ⋮ 🗑
mysql-1 b8f92a01eca6	mba-impacta-dev-ops-mysql	Running	0.58%	3306:3306	6 minutes ago	■ ⋮ 🗑
pythonapp-1 8f207c656bba	mba-impacta-dev-ops-pythonapp	Exited	0%		6 minutes ago	▶ ⋮ 🗑

O arquivo docker-compose ainda foi criado permitindo o monitoramento de volumes do tipo *bind-mount* i.e., os diretórios de logs de execução de código e de camada raw de dados que são atualizados com seus arquivos específicos também podem ser obtidos na máquina local de onde foi criada a imagem do microserviço em python. A imagem abaixo permite verificar o monitoramento local, fora do contêiner, dos arquivos gerados na camada raw após o processo de obtenção de dados via API.



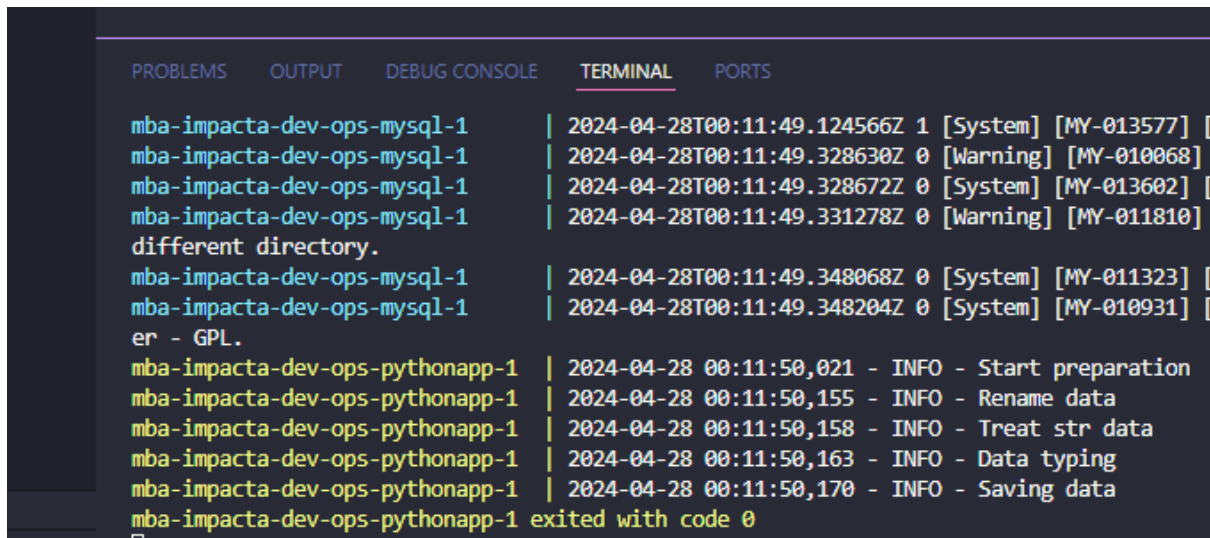
Semelhantemente, os logs de execução de código são armazenados na pasta mostrada abaixo, cuja alimentação da rodada atual do código dentro do contêiner.



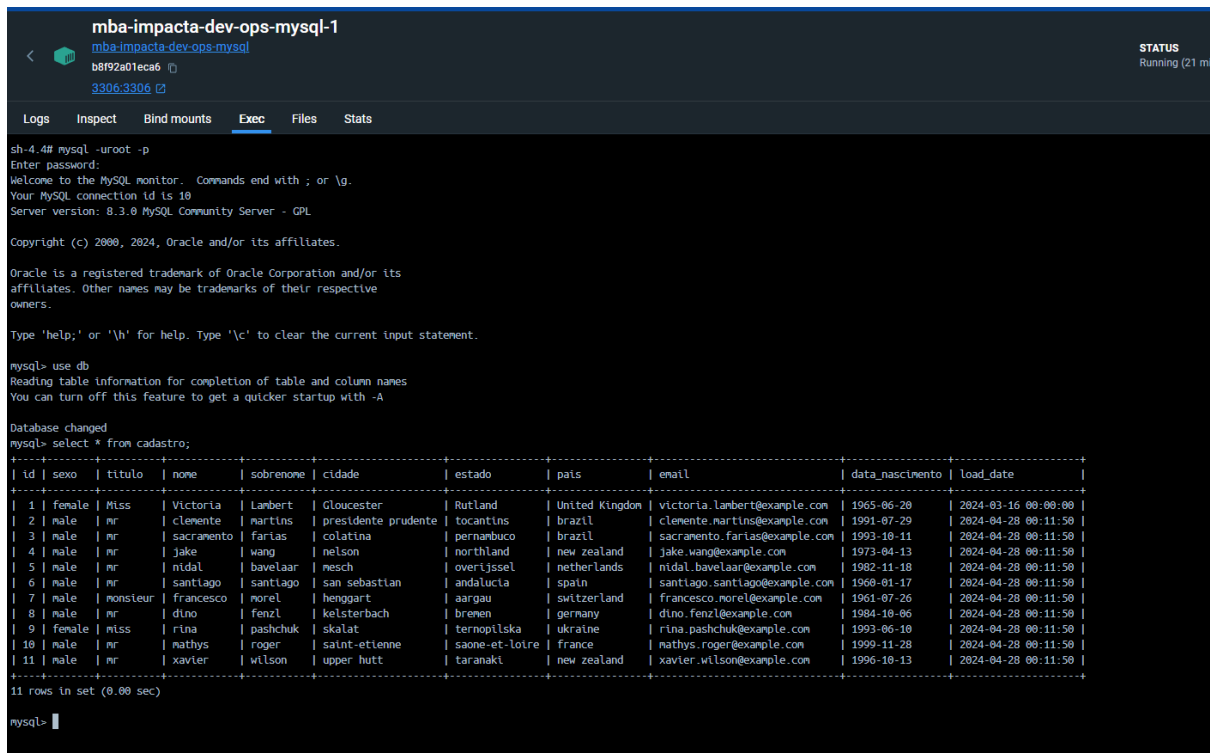
A imagem abaixo permite verificar parte dos logs de execução do pipeline de ETL gerados entre os testes de rodada consecutivos:

```
157 2024-04-28 00:11:48,519 - INFO - Start ingestion
158 2024-04-28 00:11:50,021 - INFO - Start preparation
159 2024-04-28 00:11:50,155 - INFO - Rename data
160 2024-04-28 00:11:50,158 - INFO - Treat str data
161 2024-04-28 00:11:50,163 - INFO - Data typing
162 2024-04-28 00:11:50,170 - INFO - Saving data
163
```

Os logs do pipeline também são trazidos ao terminal de execução via streaming, como mostrado na imagem abaixo:



Os dados gerados no banco relacional podem ser obtidos via comando na aplicação do Docker Desktop como mostrado na imagem abaixo:



Git actions

O push na branch main permite disparar o workflow que irá validar o código com critério de nota na qualidade de escrita e documentação inline em docstrings e também fará o push da imagem dos microserviços da aplicação em python e do banco relacional MySQL no Docker Hub.

O workflow de avaliação do código usando *pylint* permite obter os resultados da imagem abaixo:

Summary

Jobs

build (3.9)

build (3.10)

push

Run details

Usage

Workflow file

build (3.9)

succeeded now in 29s

> Set up job

> Run actions/checkout@v3

> Set up Python 3.9

> Install dependencies

> Code Review

1 ▶ Run pylint ingestion.py --fail-under=5.0

7

8 -----

9 Your code has been rated at 10.00/10

> Print secret

1 ▶ Run echo *** | sed 's/./& /g'

7 https://randomuser.me/api/?results=10

> Post Set up Python 3.9

1 Post job cleanup.

> Post Run actions/checkout@v3

1 Post job cleanup.

2 /usr/bin/git version

3 git version 2.43.2

4 Temporarily overriding HOME='/home/runner/work/_temp/f97a21b8-f928-47c3-a2f6-17a8eed678ba' before making global git config

5 Adding repository directory to the temporary git global config as a safe directory

6 /usr/bin/git config --global --add safe.directory /home/runner/work/mba-impacta-dev-ops/mba-impacta-dev-ops

7 /usr/bin/git config --local --name-only --get-regexp core.sshCommand

8 /usr/bin/git submodule foreach --recursive sh -c "git config --local --name-only --get-regexp 'core.sshCommand' && git c

9 /usr/bin/git config --local --name-only --get-regexp http.https://github.com/.extraheader

10 http.https://github.com/.extraheader

11 /usr/bin/git config --local --unset-all http.https://github.com/.extraheader

12 /usr/bin/git submodule foreach --recursive sh -c "git config --local --name-only --get-regexp 'http.https://github.com/.c

13 ;"

> Complete job

O job push permite que as credenciais armazenadas no github como secrets sejam usadas para logar no DockerHub e adicionalmente façam o push das imagens nos repositórios específicos:

← Build and Push python Image to AWS ECR

✓

feat: Final push. Work complete. #11

Summary

Jobs

✓ build (3.9)

✓ build (3.10)

✓

push

Run details

Usage

Workflow file

push

succeeded 4 minutes ago in 1m 4s

> ✓ Set up job

> ✓ Check out the repo

> ✓ Log in to Docker Hub

> ✓ Set up Docker Buildx

> ✓ Build and push Python app Docker image

> ✓ Build and push MySQL Docker image

> ✓ Post Build and push MySQL Docker image

> ✓ Post Build and push Python app Docker image

> ✓ Post Set up Docker Buildx

> ✓ Post Log in to Docker Hub

> ✓ Post Check out the repo

> ✓ Complete job

dockerhub

Explore

Repositories

Organizations

🔍

Search Docker Hub

luccaszullianesilva

Search by repository name 🔍

All Content

Create repository

luccaszullianesilva / mysql

Contains: Image • Last pushed: 3 minutes ago

🚫 Security unknown

☆ 0

📦 15

🌐 Public

luccaszullianesilva / pythonapp

Contains: Image • Last pushed: 3 minutes ago

🚫 Security unknown

☆ 0

📦 2

🌐 Public

