

Tutorial - Projeto Human Resources Analytics

Preparação do ambiente

1. Faça download do Anaconda no site:
<https://www.anaconda.com/products/individual#Downloads>
2. Instalar o Docker Desktop no Windows ou no Linux
<https://www.docker.com/get-started>
3. Faça download do Visual Studio code
<https://code.visualstudio.com/download>

Crie um diretório na sua máquina para armazenar scripts e outros artefatos, exemplo:

C:\projeto

/home/<seunome>/projeto

Instalação e Configuração do Mysql Server

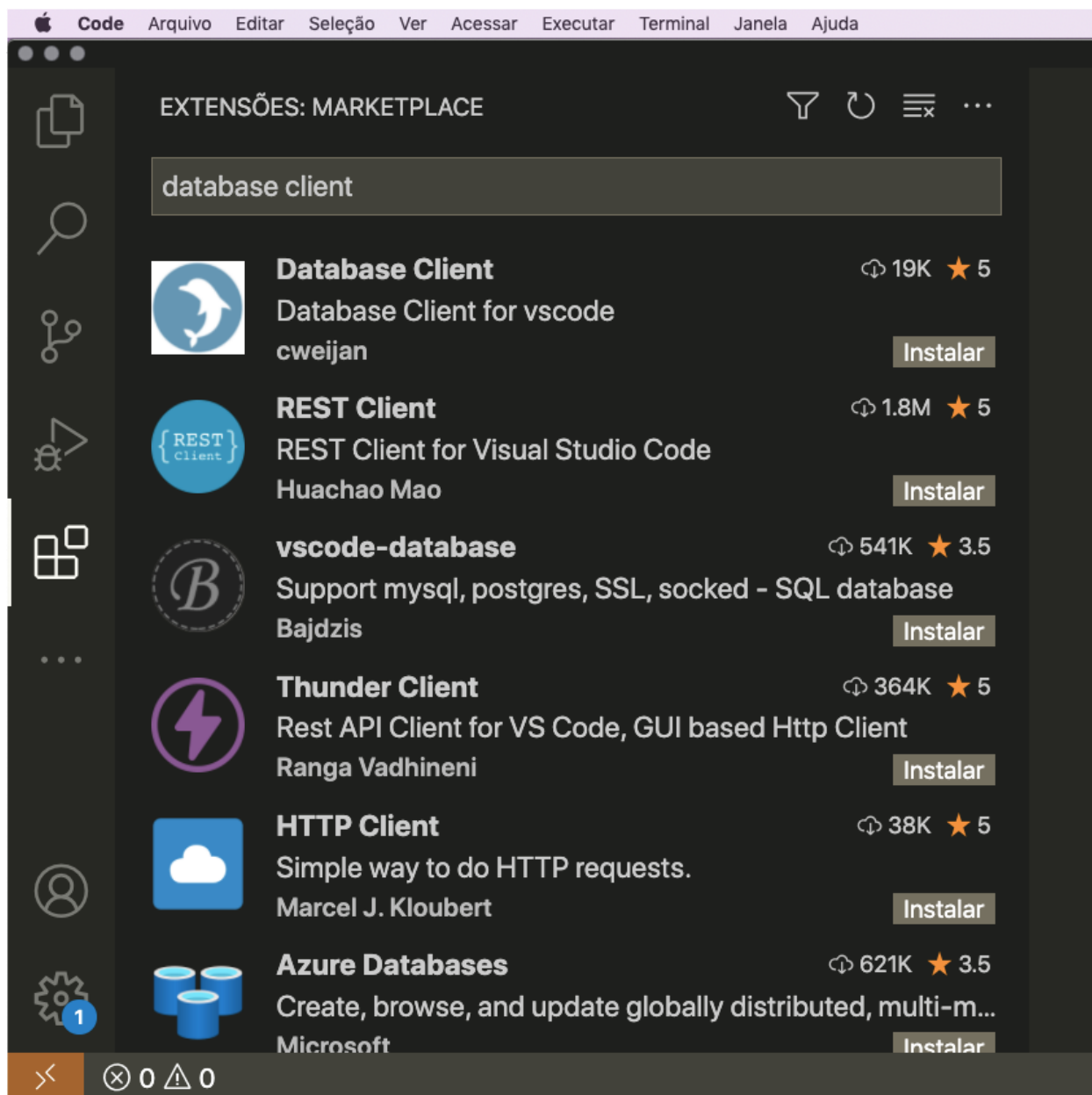
Se estiver usando Windows abra o **Powershell** e digite:

Crie o container do mysql habilitando a porta 3307:

```
docker run --name mysqlbd1 -e MYSQL_ROOT_PASSWORD=stack -p "3307:3306" -d mysql
```

Teste o acesso ao banco de dados usando o Visual Studio Code:

Abra o Visual Studio Code e instale a extensão: **Database Client**



Teste o acesso ao banco de dados Mysql:

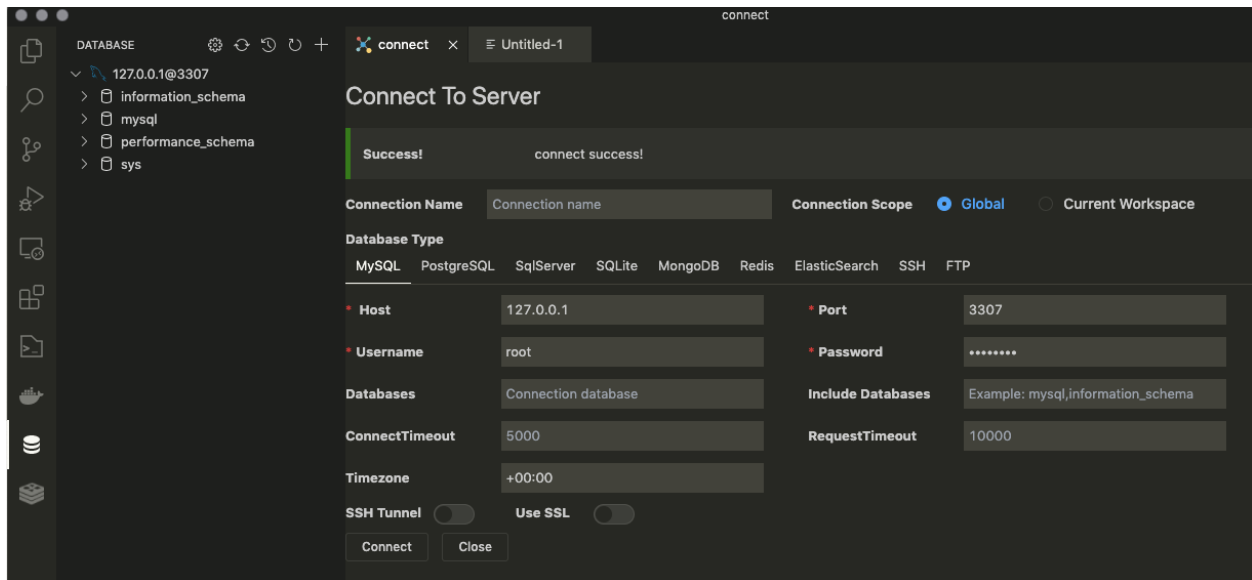
Coloque as configurações:

Host: 127.0.0.1

Username: root

Port: 3307

Password: stack



Instalação e Configuração do Data Lake com Minio Server

Dentro do diretório projeto crie o diretório **datalake**.

Se estiver usando Windows abra o terminal do Powershell e execute o comando:

```
docker run --name minio -d -p 9000:9000 -p 9001:9001 -v "$PWD/datalake:/data" minio/minio server /data --console-address ":9001"
```

Teste o acesso ao Minio:

Abra o browser e digite: <http://localhost:9000/login>

username: minioadmin

password: minioadmin

Instalação e Configuração do Airflow.

1. Dentro do **diretório projeto** crie o diretório **airflow**.
2. Navegue até o diretório airflow e crie o diretório **dags**.
3. Faça download da imagem e execute o container do Apache Airflow.

3a. Se estiver usando Windows abra o terminal do Powershell e execute o comando:

```
docker run -d -p 8080:8080 -v "$PWD/airflow/dags:/opt/airflow/dags/" --  
entrypoint=/bin/bash --name airflow apache/airflow:2.1.1-python3.8 -c '(airflow db init  
&& airflow users create --username admin --password stack --firstname Felipe --lastname  
Lastname --role Admin --email admin@example.org); airflow webserver & airflow scheduler'
```

3b. Instale as bibliotecas necessárias para o ambiente:

Execute o comando abaixo para se conectar ao container do airflow:

```
docker container exec -it airflow bash
```

Em seguida instale as bibliotecas:

```
pip install pymysql xlrd openpyxl minio
```

3c. Se não der nenhum erro, acesse a interface web do Apache Airflow com o endereço (*Aguarde uns 5 minutos antes de abrir o terminal*):

```
https://localhost:8080
```

Faça o login de acesso ao Apache Airflow

Login: admin

Senha: stack

Clique em Admin >> Variables

Crie as seguintes variáveis:

```
data_lake_server = 172.17.0.4:9000
```

```
data_lake_login = minioadmin
```


```
data_lake_password = minioadmin
```

```
database_server = 172.17.0.3 ( Use o comando inspect para descobrir o ip do  
container: docker container inspect mysqlbd1 - localizar o atributo IPAddress)
```

```
database_login = root
```

```
database_password = stack
```

```
database_name = employees
```


Airflow
DAGs
Security ▾
Browse ▾
Admin ▾
Docs ▾

Add Variable

Key *

data_lake_login

Val

minioadmin

Description

Login de acesso ao Data Lake

Save

←

As variáveis criadas devem ficar como:















List Variable

Search ▾

+

Actions ▾

←

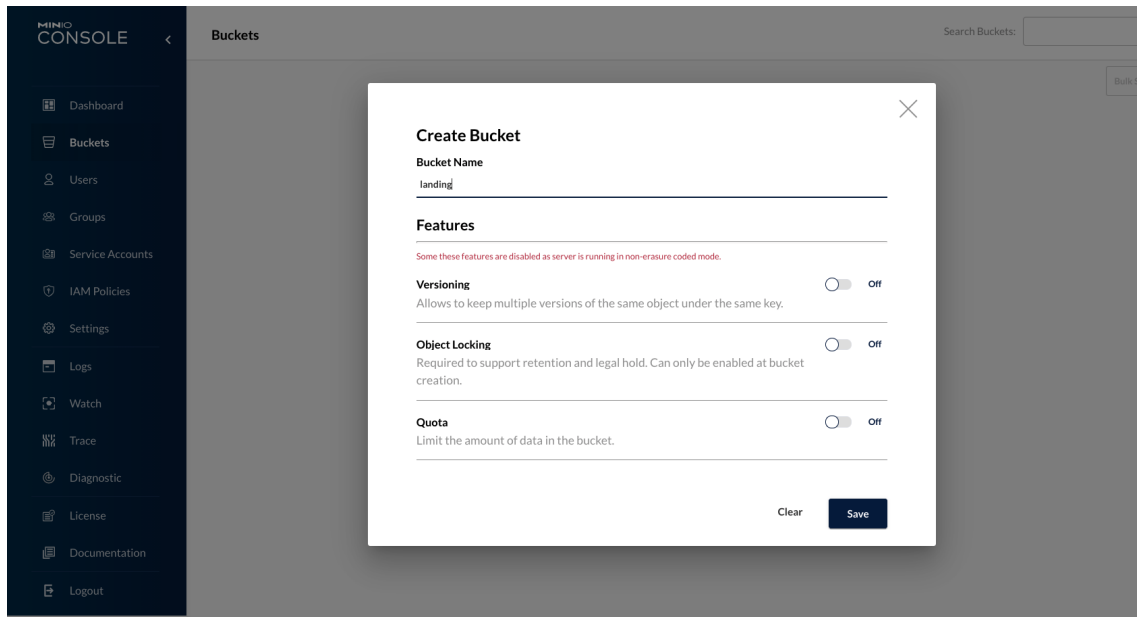
<input type="checkbox"/>	Key ▾	Val ▾	Description ▾
<input type="checkbox"/>  	data_lake_login	minioadmin	Login de conexão com o Data Lake
<input type="checkbox"/>  	data_lake_password	*****	Senha do usuário para acesso ao Data Lake
<input type="checkbox"/>  	data_lake_server	172.17.0.4:9000	Endereço do servidor Data Lake
<input type="checkbox"/>  	database_login	root	login do usuário do banco de dados
<input type="checkbox"/>  	database_name	employees	senha do usuário do banco de dados
<input type="checkbox"/>  	database_password	*****	Senha do usuário de banco de dados
<input type="checkbox"/>  	database_server	172.17.0.2	Ip e porta do servidor de banco de dados

Modelagem de Dados

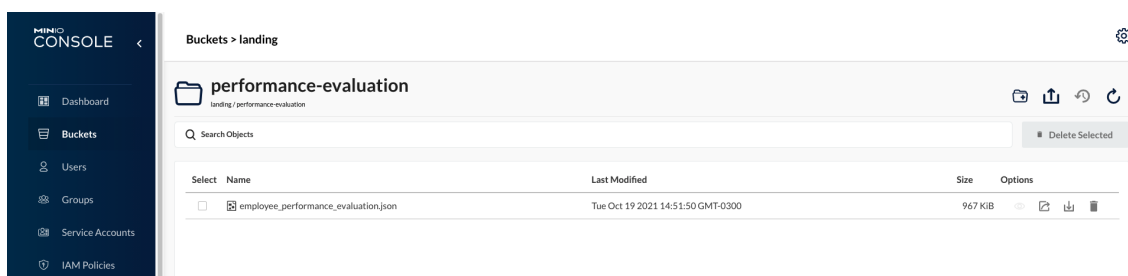
Configurando o Data Lake

- Inicie o container do Minio com o comando:
 - Abra o **Docker Desktop** para iniciar o docker engine

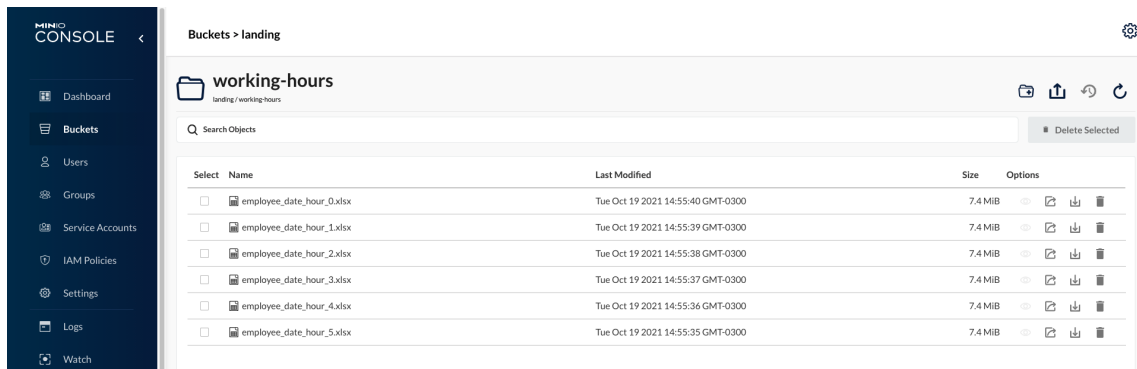
- b. Abra o Powershell (Windows) e execute o comando abaixo para iniciar o container do Minio: `docker container start <nome-do-container>`
- c. Em seguida acesse o console do minio no endereço: <http://localhost:9001/login>
- d. Crie os buckets *landing*, *processing* e *curated* como na imagem a seguir:
- e. Clique em **Buckets** em seguida clique em **create bucket**



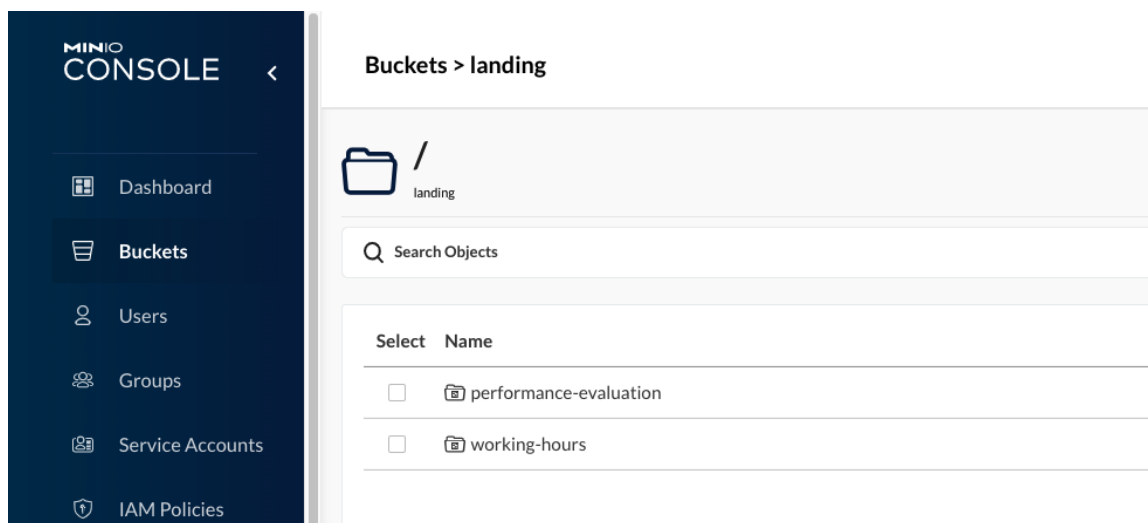
- f. Após criar os buckets clique no bucket landing e crie a pasta: **performance-evaluation** em seguida clique em Upload e carregue o arquivo: **employee_performance_evaluation.json** Veja a imagem abaixo:



- g. Clique no bucket landing
- h. Crie outra pasta chamada **working-hours** e faça upload dos arquivos .xlsx veja como fica na imagem abaixo:



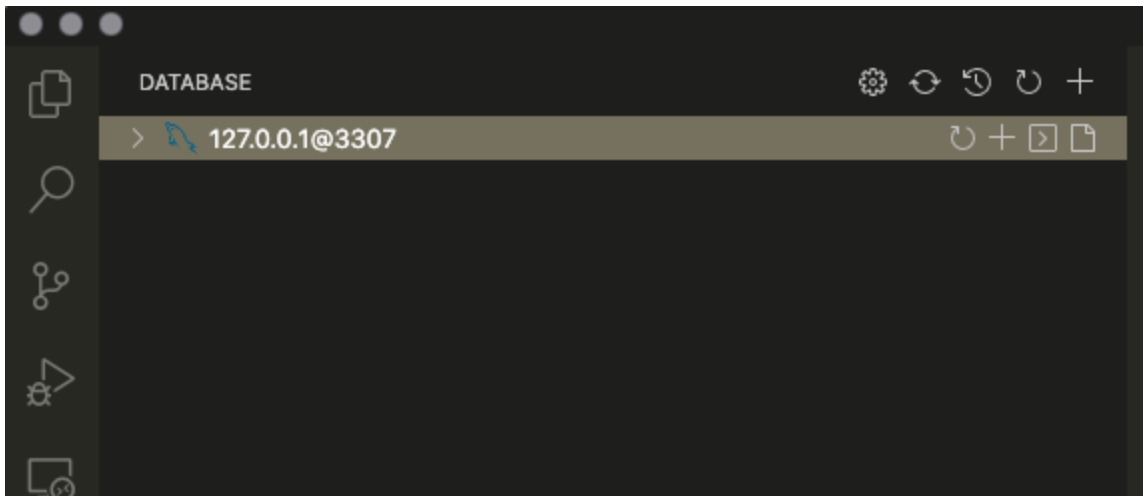
i. Ao clicar no bucket landing ficamos como:



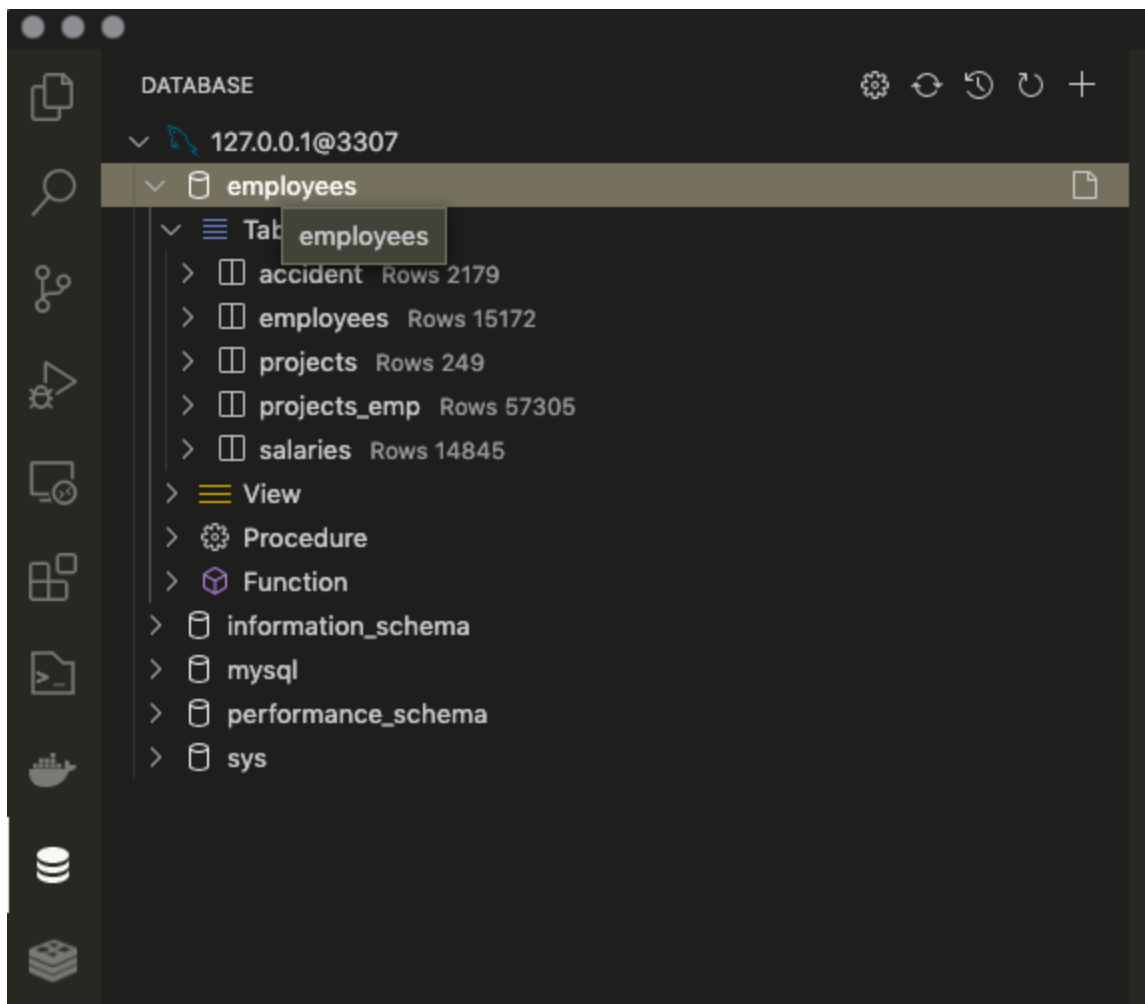
Subindo o banco de dados e carregando o banco de dados

1. Abra o Docker Desktop para iniciar o docker engine
2. Abra o console do Powershell (Windows) ou o terminal linux e execute o comando abaixo para iniciar o container do mysql: `docker container start mysqlbdt1`
3. Em seguida abra o Visual Studio Code para carregar o arquivo .sql para dar carga no banco de dados:

4. Clique com o botão direito do mouse na conexão como mostrado na imagem abaixo

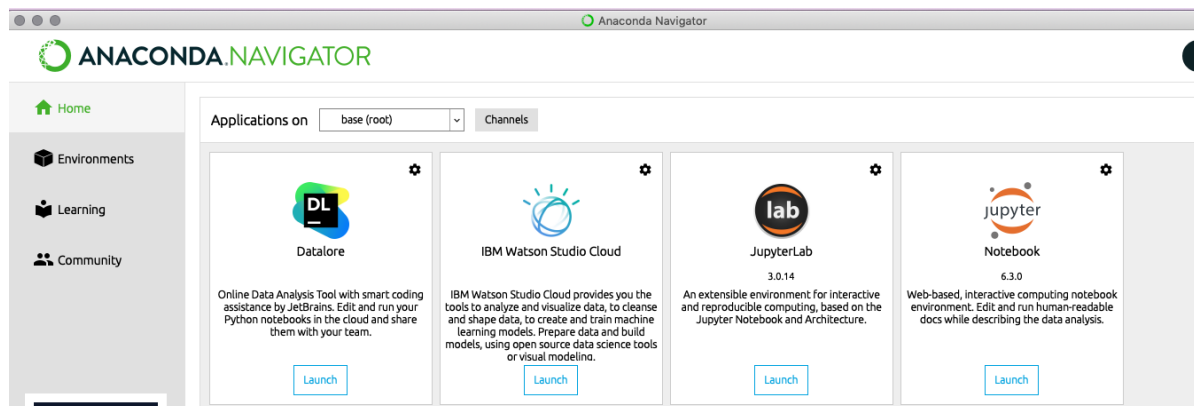


5. Escolha a opção Import Sql
6. Aguarde o processo de importação. Após importar clique em atualizar a conexão para visualizar o banco de dados **employees** recém criado. Veja imagem abaixo:

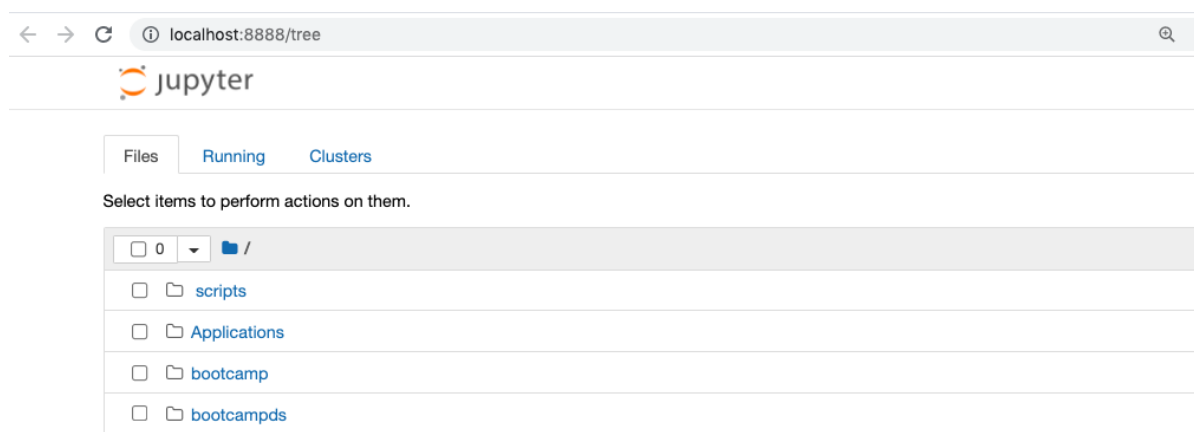


Abrindo o Jupyter Notebook

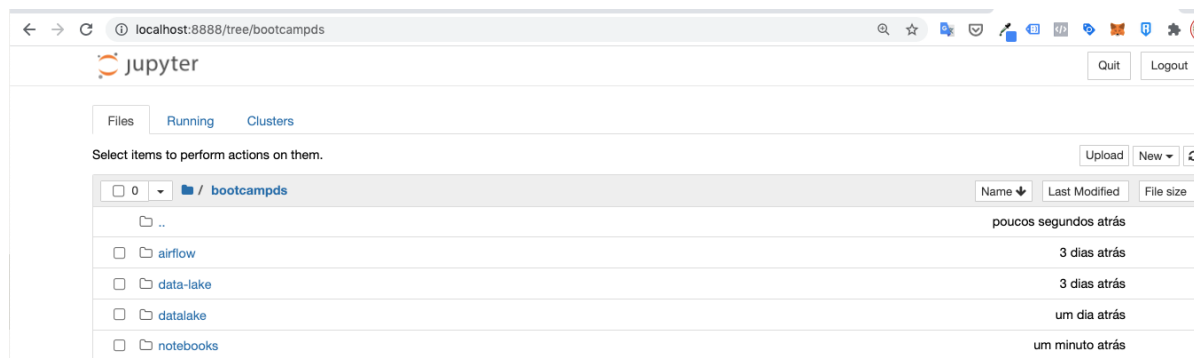
1. Abra o Anaconda Navigator. Se estiver no Windows pesquise no menu iniciar "Anaconda Navigator"
2. Clique em **launch** para abrir o jupyter notebook



3. Ao abrir navegue até o diretório criado para o bootcamp, por exemplo: **bootcamps**



4. Deverá conter a estrutura de diretórios como a seguir:

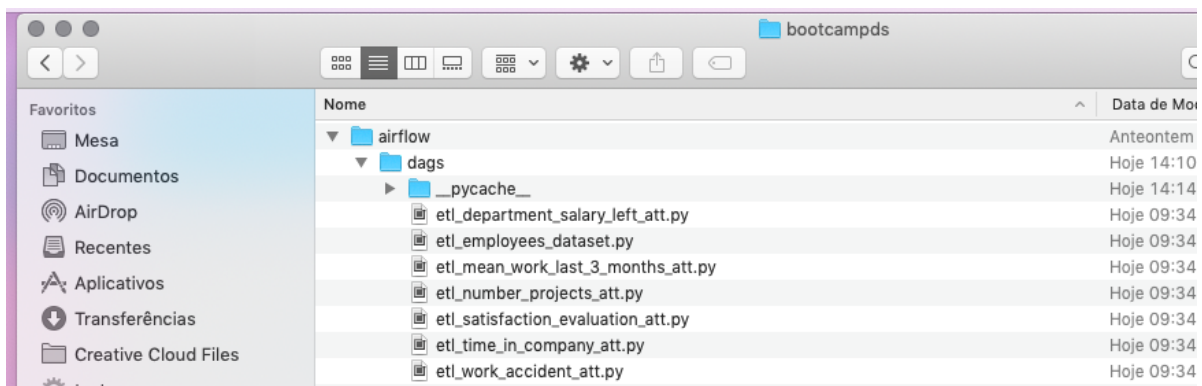


5. Clique no diretório notebooks, deverá conter o notebook **modelagem_dados.ipynb**



Subindo o Airflow e criando as Dags:

1. Descompacte os arquivos .py dentro do seu diretório usado no bootcamp, exemplo *C:\Felipe\bootcampds* ou */home/felipe/bootcampds*
2. Mova os arquivos .py para o diretório **airflow/dags** como na imagem abaixo:



3. **Atenção:** Certifique que todos os arquivos estão dentro do diretório **airflow/dags**
4. Abra o console do Powershell (Windows) ou o terminal linux e execute o comando abaixo para iniciar o container do mysql: `docker container start airflow`
5. Aguarde uns 5 minutos e acesse o console do airflow no endereço: <http://localhost:8080/>
6. Faça o login com usuário admin e a senha stack.
7. Ao clicar em Dags deve aparecer as dags criadas como na imagem abaixo:

DAG	Owner	Runs	Schedule	Last Run	Recent Tasks	Actions	Links
etl_department_salary_left_att	Airflow	1	@once			▶ ↺ ⏏	...
etl_employees_dataset	Airflow	1	@once			▶ ↺ ⏏	...
etl_mean_work_last_3_months_att	Airflow	1	@once			▶ ↺ ⏏	...
etl_number_projects_att	Airflow	1	@once			▶ ↺ ⏏	...
etl_satisfaction_evaluation_att	Airflow	1	@once			▶ ↺ ⏏	...
etl_time_in_company_att	Airflow	1	@once			▶ ↺ ⏏	...
etl_work_accident_att	Airflow	1	@once			▶ ↺ ⏏	...

Rodando as Dags:

1. **Atenção:** Antes de executar as dags verifique se o ip do mysql ou do minio alterou.
2. Para verificar use o comando: `docker container inspect mysqlbd1` e visualize o campo IP Address