

Desafio 12

Rafael Gomes Carneiro - RA185462

SQLite e Polars

```
library(reticulate)
use_python("C:/Users/rafa3/AppData/Local/Programs/Python/Python312/python.exe",
           required = TRUE)

setwd("C:/Users/Public/Documents/UNICAMP/ME315 - Banco de Dados/Desafios")
```

```
import polars as pl
import sqlite3

print(pl.__version__)
```

1.33.1

```
print(sqlite3.sqlite_version)
```

3.42.0

```
conn = sqlite3.connect('../dados/data.db')
cursor = conn.cursor()
```

```
cursor.execute("DROP TABLE vendas")
```

<sqlite3.Cursor object at 0x000002AB0C6876C0>

```
cursor.execute('''
CREATE TABLE IF NOT EXISTS vendas (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    vendedor TEXT,
    produto TEXT,
    valor REAL,
    data_venda TEXT,
    UNIQUE(vendedor, produto, data_venda)
)
''')
```

<sqlite3.Cursor object at 0x000002AB0C6876C0>

```
conn.commit()
```

```
cursor.execute('''
INSERT INTO vendas (vendedor, produto, valor, data_venda)
VALUES
    ('Ana', 'Produto A', 120.5, '2024-09-01'),
    ('Carlos', 'Produto B', 200.0, '2024-10-02'),
    ('Ana', 'Produto C', 150.0, '2024-09-03'),
    ('Bruno', 'Produto A', 300.0, '2024-11-04'),
    ('Carlos', 'Produto C', 100.0, '2024-10-05');
''')
```

<sqlite3.Cursor object at 0x000002AB0C6876C0>

```
conn.commit()
```

```
cursor.execute("SELECT * FROM vendas")
```

<sqlite3.Cursor object at 0x000002AB0C6876C0>

```
rows = cursor.fetchall()
for row in rows:
    print(row)
```

```
(1, 'Ana', 'Produto A', 120.5, '2024-09-01')
(2, 'Carlos', 'Produto B', 200.0, '2024-10-02')
```

```
(3, 'Ana', 'Produto C', 150.0, '2024-09-03')
(4, 'Bruno', 'Produto A', 300.0, '2024-11-04')
(5, 'Carlos', 'Produto C', 100.0, '2024-10-05')
```

```
dados = pl.read_database("SELECT * FROM vendas", conn)
print(dados)
```

shape: (5, 5)

	id	vendedor	produto	valor	data_venda
	---	---	---	---	---
	i64	str	str	f64	str
1		Ana	Produto A	120.5	2024-09-01
2		Carlos	Produto B	200.0	2024-10-02
3		Ana	Produto C	150.0	2024-09-03
4		Bruno	Produto A	300.0	2024-11-04
5		Carlos	Produto C	100.0	2024-10-05

```
vendas_total = pl.read_database("""
    SELECT vendedor, SUM(valor) as total_vendas
    FROM vendas
    GROUP BY vendedor;
""", conn)
print(vendas_total)
```

shape: (3, 2)

	vendedor	total_vendas
	---	---
	str	f64
	Ana	270.5
	Bruno	300.0
	Carlos	300.0

```
vendas_medias = pl.read_database(''
    SELECT vendedor, AVG(valor) as total_vendas
```

```

        FROM vendas
        GROUP BY vendedor;
''' , conn)
print(vendas_medias)

```

shape: (3, 2)

vendedor	total_vendas
---	---
str	f64
Ana	135.25
Bruno	300.0
Carlos	150.0

```

vendas_comb = pl.read_database("""
SELECT vendedor,
        COUNT(*) as numero_vendas,
        SUM(valor) as total_vendas,
        AVG(valor) as media_vendas
FROM vendas
GROUP BY vendedor;
""", conn)

print(vendas_comb)

```

shape: (3, 4)

vendedor	numero_vendas	total_vendas	media_vendas
---	---	---	---
str	i64	f64	f64
Ana	2	270.5	135.25
Bruno	1	300.0	300.0
Carlos	2	300.0	150.0

```

ticket_alto = pl.read_database("""
SELECT * FROM vendas WHERE valor >= 200

```

```
""" , conn)
print(ticket_alto)
```

shape: (2, 5)

id	vendedor	produto	valor	data_venda
---	---	---	---	---
i64	str	str	f64	str
2	Carlos	Produto B	200.0	2024-10-02
4	Bruno	Produto A	300.0	2024-11-04

```
vendas_mensais = pl.read_database("""
SELECT strftime('%Y-%m', data_venda) AS mes, SUM(valor) AS total_vendas
FROM vendas GROUP BY mes ORDER BY mes
""", conn)
print(vendas_mensais)
```

shape: (3, 2)

mes	total_vendas
---	---
str	f64
2024-09	270.5
2024-10	300.0
2024-11	300.0

```
cursor.execute('''
CREATE TABLE IF NOT EXISTS produtos (
    id INTEGER PRIMARY KEY,
    nome TEXT NOT NULL,
    categoria TEXT NOT NULL,
    preco REAL NOT NULL,
    estoque INTEGER NOT NULL
);
''')
```

<sqlite3.Cursor object at 0x000002AB0C6876C0>

```

cursor.execute('''
INSERT INTO produtos (nome, categoria, preco, estoque) VALUES
    ('Produto A', 'Categoria 1', 100.0, 50),
    ('Produto B', 'Categoria 2', 150.0, 30),
    ('Produto C', 'Categoria 1', 200.0, 20),
    ('Produto D', 'Categoria 2', 250.0, 10),
    ('Produto E', 'Categoria 3', 300.0, 0);
''')

```

<sqlite3.Cursor object at 0x000002AB0C6876C0>

```
conn.commit()
```

```

prods = pl.read_database("SELECT * FROM produtos", conn)
print(prods)

```

shape: (50, 5)

id	nome	categoria	preco	estoque
---	---	---	---	---
i64	str	str	f64	i64
1	Produto A	Categoria 1	100.0	50
2	Produto B	Categoria 2	150.0	30
3	Produto C	Categoria 1	200.0	20
4	Produto D	Categoria 2	250.0	10
5	Produto E	Categoria 3	300.0	0
...
46	Produto A	Categoria 1	100.0	50
47	Produto B	Categoria 2	150.0	30
48	Produto C	Categoria 1	200.0	20
49	Produto D	Categoria 2	250.0	10
50	Produto E	Categoria 3	300.0	0

```

lucros = pl.read_database("""
SELECT produto, valor AS compra, preco AS venda, preco-valor AS lucro
FROM vendas
INNER JOIN produtos ON vendas.produto = produtos.nome
""", conn)
print(lucros)

```

shape: (50, 4)

produto	compra	venda	lucro
---	---	---	---
str	f64	f64	f64
Produto A	120.5	100.0	-20.5
Produto A	120.5	100.0	-20.5
Produto A	120.5	100.0	-20.5
Produto A	120.5	100.0	-20.5
Produto A	120.5	100.0	-20.5
...
Produto C	100.0	200.0	100.0
Produto C	100.0	200.0	100.0
Produto C	100.0	200.0	100.0
Produto C	100.0	200.0	100.0
Produto C	100.0	200.0	100.0

```
lucro_medio = pl.read_database("""
SELECT vendedor, produto, AVG(preco-valor) AS lucro_medio
FROM vendas
INNER JOIN produtos ON vendas.produto = produtos.nome
GROUP BY vendedor
""", conn)
print(lucro_medio)
```

shape: (3, 3)

vendedor	produto	lucro_medio
---	---	---
str	str	f64
Ana	Produto A	14.75
Bruno	Produto A	-200.0
Carlos	Produto B	25.0

```
horario_normal <- function(timestamp = Sys.time()) {
  format(timestamp, "%d/%m/%Y %H:%M:%S")
}
```

```
cat("Arquivo compilado em:", horario_normal(), "\n")
```

Arquivo compilado em: 09/10/2025 11:27:19