

Lista dos comandos basicos para MySQL - Soulcode

```
1 !curl ipecho.net/plain
```

```
34.74.90.201
```

```
1 # Instalação do pacote mysql-connector-python usando o gerenciador de pacotes pip
```

```
2 !pip install mysql-connector-python
```

```
Collecting mysql-connector-python
```

```
  Downloading mysql_connector_python-8.3.0-cp310-cp310-manylinux_2_17_x86_64.whl (21.5 MB)
```

```
21.5/21.5 MB 41.1 MB/s eta 0:00:00
```

```
Installing collected packages: mysql-connector-python
```

```
Successfully installed mysql-connector-python-8.3.0
```

```
1 # Importação das bibliotecas necessárias
```

```
2 import mysql.connector # necessária para se conectar com o banco de dados MySQL
```

```
3 import pandas as pd # Será usada para visualização das tabelas extraídas do My SQL
```

[+ Código](#)[+ Texto](#)

```
1 # Definição das informações de conexão ao servidor (instância)
```

```
2 host = '34.172.110.76'
```

```
3 user = 'root'
```

```
4 password = ''
```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
```

```
2 connection = mysql.connector.connect(
```

```
3     host=host,
```

```
4     user=user,
```

```
5     password=password,
```

```
6 )
```

```
7 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
```

```
8 query = 'create database locadora;' # Definição da consulta SQL a ser executada
```

```
9 cursor.execute(query) # Execução da consulta no banco de dados
```

```
10 cursor.close() # Fechamento do cursor após a conclusão da consulta
```

```
11 connection.close() # Fechamento da conexão após a consulta
```

```
1 # Definição das informações de conexão ao banco de dados
```

```
2 host = '34.172.110.76'
```

```
3 user = 'root'
```

```
4 password = ''
```

```
5 database = 'locadora'
```

```

1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 CREATE TABLE cliente(
11     codcliente INTEGER PRIMARY KEY,
12     nome VARCHAR(50) NOT NULL,
13     cidade VARCHAR(50),
14     sexo VARCHAR(10),
15     estado VARCHAR(20),
16     estadocivil VARCHAR(20)
17 );''' # Definição da consulta SQL a ser executada
18 cursor.execute(query) # Execução da consulta no banco de dados
19 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
20 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resulta
21 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
22 cursor.close() # Fechamento do cursor após a conclusão da consulta
23 connection.close() # Fechamento da conexão após a consulta

```

—

```

1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 CREATE TABLE marca(
11     codmarca INTEGER PRIMARY KEY,
12     marca VARCHAR(30) NOT NULL
13 );
14 ''' # Definição da consulta SQL a ser executada
15 cursor.execute(query) # Execução da consulta no banco de dados
16 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
17 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resulta
18 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
19 cursor.close() # Fechamento do cursor após a conclusão da consulta
20 connection.close() # Fechamento da conexão após a consulta

```

—

```

1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 CREATE TABLE carro(
11     codcarro INTEGER PRIMARY KEY,
12     codmarca INTEGER NOT NULL,
13     modelo VARCHAR(30) NOT NULL,
14     valor real,
15     FOREIGN KEY (codmarca) REFERENCES marca (codmarca)
16 );
17 ''' # Definição da consulta SQL a ser executada
18 cursor.execute(query) # Execução da consulta no banco de dados
19 result = cursor.fetchall() # Recuperação de todos os resultados da consulta

```

```

20 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resultados da consulta
21 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
22 cursor.close() # Fechamento do cursor após a conclusão da consulta
23 connection.close() # Fechamento da conexão após a consulta

```

```

1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 CREATE TABLE aluguel(
11     codaluguel INTEGER PRIMARY KEY,
12     codcliente INTEGER NOT NULL,
13     codcarro INTEGER NOT NULL,
14     dataaluguel DATE NOT NULL,
15     FOREIGN KEY (codcliente) REFERENCES cliente (codcliente),
16     FOREIGN KEY (codcarro) REFERENCES carro (codcarro)
17 );
18 ''' # Definição da consulta SQL a ser executada
19 cursor.execute(query) # Execução da consulta no banco de dados
20 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
21 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resultados da consulta
22 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
23 cursor.close() # Fechamento do cursor após a conclusão da consulta
24 connection.close() # Fechamento da conexão após a consulta

```

```

1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 INSERT INTO cliente (codcliente, nome, cidade, sexo, estado, estadocivil)
11 VALUES
12 (1, 'Ana Silva', 'Duque de Caxias', 'F', 'RJ', 'C'),
13 (2, 'Bruna Pereira', 'Niterói', 'F', 'RJ', 'C'),
14 (3, 'Túlio Nascimento', 'Duque de Caxias', 'M', 'RJ', 'S'),
15 (4, 'Fernando Souza', 'Campinas', 'M', 'SP', 'S'),
16 (5, 'Lúcia Andrade', 'São Paulo', 'F', 'SP', 'C');
17 ''' # Definição da consulta SQL a ser executada
18 cursor.execute(query) # Execução da consulta no banco de dados
19 connection.commit() # Confirmar as alterações
20 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
21 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resultados da consulta
22 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
23 cursor.close() # Fechamento do cursor após a conclusão da consulta
24 connection.close() # Fechamento da conexão após a consulta

```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 INSERT INTO marca (codmarca, marca) VALUES
11 (1, 'Ford'),
12 (2, 'Fiat'),
13 (3, 'Chevrolet'),
14 (4, 'Volkswagen'),
15 (5, 'Renault');
16 ''' # Definição da consulta SQL a ser executada
17 cursor.execute(query) # Execução da consulta no banco de dados
18 connection.commit() # Confirmar as alterações
19 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
20 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resulta
21 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
22 cursor.close() # Fechamento do cursor após a conclusão da consulta
23 connection.close() # Fechamento da conexão após a consulta
```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 INSERT INTO carro (codcarro, modelo, codmarca, valor) VALUES
11 (1, 'Ford', 1,100.00),
12 (2, 'Fiat', 2,150.00),
13 (3, 'Onix', 3,170.00),
14 (4, 'Polo', 4, 150.00),
15 (5, 'Kwid', 5, 120.00);
16 ''' # Definição da consulta SQL a ser executada
17 cursor.execute(query) # Execução da consulta no banco de dados
18 connection.commit() # Confirmar as alterações
19 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
20 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resulta
21 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
22 cursor.close() # Fechamento do cursor após a conclusão da consulta
23 connection.close() # Fechamento da conexão após a consulta
```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 INSERT INTO aluguel (codaluguel, codcliente, codcarro, dataaluguel)VALUES
11 (1, 3, 2, '2023/04/01'),
12 (2, 2, 1, '2023/04/02'),
13 (3, 2, 1, '2023/04/03'),
14 (4, 2, 3, '2023/04/04'),
15 (5, 1, 4, '2023/04/05'),
16 (6, 1, 4, '2023/04/06'),
17 (7, 1, 1, '2023/04/07'),
18 (8, 5, 2, '2023/04/08'),
19 (9, 5, 2, '2023/04/09'),
20 (10, 3, 1, '2023/04/10');
21 ''' # Definição da consulta SQL a ser executada
22 cursor.execute(query) # Execução da consulta no banco de dados
23 connection.commit() # Confirmar as alterações
24 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
25 tables = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resulta
26 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
27 cursor.close() # Fechamento do cursor após a conclusão da consulta
28 connection.close() # Fechamento da conexão após a consulta
```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 SELECT * FROM cliente;
11 ''' # Definição da consulta SQL a ser executada
12 cursor.execute(query) # Execução da consulta no banco de dados
13 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
14 cliente = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os result
15 display(tables) # Exibição do DataFrame Pandas contendo os resultados da consulta
16 cursor.close() # Fechamento do cursor após a conclusão da consulta
17 connection.close() # Fechamento da conexão após a consulta
```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 SELECT * FROM marca;
11 ''' # Definição da consulta SQL a ser executada
12 cursor.execute(query) # Execução da consulta no banco de dados
13 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
14 marca = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resultados
15 display(marca) # Exibição do DataFrame Pandas contendo os resultados da consulta
16 cursor.close() # Fechamento do cursor após a conclusão da consulta
17 connection.close() # Fechamento da conexão após a consulta
```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 SELECT * FROM carro;
11 ''' # Definição da consulta SQL a ser executada
12 cursor.execute(query) # Execução da consulta no banco de dados
13 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
14 carro = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resultados
15 display(carro) # Exibição do DataFrame Pandas contendo os resultados da consulta
16 cursor.close() # Fechamento do cursor após a conclusão da consulta
17 connection.close() # Fechamento da conexão após a consulta
```

```
1 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
2 connection = mysql.connector.connect(
3     host=host,
4     user=user,
5     password=password,
6     database = database,
7 )
8 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
9 query = '''
10 SELECT * FROM aluguel;
11 ''' # Definição da consulta SQL a ser executada
12 cursor.execute(query) # Execução da consulta no banco de dados
13 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
14 aluguel = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resultados
15 display(aluguel) # Exibição do DataFrame Pandas contendo os resultados da consulta
16 cursor.close() # Fechamento do cursor após a conclusão da consulta
17 connection.close() # Fechamento da conexão após a consulta
```

```
1 display(cliente)
2 display(marca)
3 display(carro)
4 display(aluguel)
```

| | codcliente | nome | cidade | sexo | estado | estadocivil |
|---|------------|------------------|-----------------|------|--------|-------------|
| 0 | 1 | Ana Silva | Duque de Caxias | F | RJ | C |
| 1 | 2 | Bruna Pereira | Niterói | F | RJ | C |
| 2 | 3 | Túlio Nascimento | Duque de Caxias | M | RJ | S |
| 3 | 4 | Fernando Souza | Campinas | M | SP | S |
| 4 | 5 | Lúcia Andrade | São Paulo | F | SP | C |

| | codmarca | marca |
|---|----------|------------|
| 0 | 1 | Ford |
| 1 | 2 | Fiat |
| 2 | 3 | Chevrolet |
| 3 | 4 | Volkswagen |
| 4 | 5 | Renault |

| | codcarro | codmarca | modelo | valor |
|---|----------|----------|--------|-------|
| 0 | 1 | 1 | Ford | 100.0 |
| 1 | 2 | 2 | Fiat | 150.0 |
| 2 | 3 | 3 | Onix | 170.0 |
| 3 | 4 | 4 | Polo | 150.0 |
| 4 | 5 | 5 | Kwid | 120.0 |

| | codaluguel | codcliente | codcarro | dataaluguel |
|---|------------|------------|----------|-------------|
| 0 | 1 | 3 | 2 | 2023-04-01 |
| 1 | 2 | 2 | 1 | 2023-04-02 |
| 2 | 3 | 2 | 1 | 2023-04-03 |
| 3 | 4 | 2 | 3 | 2023-04-04 |
| 4 | 5 | 1 | 4 | 2023-04-05 |
| 5 | 6 | 1 | 4 | 2023-04-06 |
| 6 | 7 | 1 | 1 | 2023-04-07 |
| 7 | 8 | 5 | 2 | 2023-04-08 |
| 8 | 9 | 5 | 2 | 2023-04-09 |

| codcliente | | nome | cidade | sexo | estado | estadocivil |
|------------|---|------------------|-----------------|------|--------|-------------|
| 0 | 1 | Ana Silva | Duque de Caxias | F | RJ | C |
| 1 | 2 | Bruna Pereira | Niterói | F | RJ | C |
| 2 | 3 | Túlio Nascimento | Duque de Caxias | M | RJ | S |
| 3 | 4 | Fernando Souza | Campinas | M | SP | S |
| 4 | 5 | Lúcia Andrade | São Paulo | F | SP | C |

| codmarca | | marca |
|----------|---|------------|
| 0 | 1 | Ford |
| 1 | 2 | Fiat |
| 2 | 3 | Chevrolet |
| 3 | 4 | Volkswagen |
| 4 | 5 | Renault |

| codcarro | codmarca | modelo | valor |
|----------|----------|--------|------------|
| 0 | 1 | 1 | Ford 100.0 |
| 1 | 2 | 2 | Fiat 150.0 |
| 2 | 3 | 3 | Onix 170.0 |
| 3 | 4 | 4 | Polo 150.0 |
| 4 | 5 | 5 | Kwid 120.0 |

| codaluguel | codcliente | codcarro | dataaluguel | |
|------------|------------|----------|-------------|------------|
| 0 | 1 | 3 | 2 | 2023-04-01 |
| 1 | 2 | 2 | 1 | 2023-04-02 |
| 2 | 3 | 2 | 1 | 2023-04-03 |
| 3 | 4 | 2 | 3 | 2023-04-04 |
| 4 | 5 | 1 | 4 | 2023-04-05 |
| 5 | 6 | 1 | 4 | 2023-04-06 |
| 6 | 7 | 1 | 1 | 2023-04-07 |
| 7 | 8 | 5 | 2 | 2023-04-08 |
| 8 | 9 | 5 | 2 | 2023-04-09 |
| 9 | 10 | 3 | 1 | 2023-04-10 |

```
1 '''
2 Imagine que você queira exibir em uma única tabela o modelo e a marca.
3 Como poderíamos realizar a junção entre as tabelas (marca e carro)?
4 '''
5 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
6 connection = mysql.connector.connect(
7     host=host,
8     user=user,
9     password=password,
10    database=database,
11 )
12 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
13 query = '''
14 SELECT carro.modelo, marca.marca
15 FROM carro, marca
16 WHERE carro.codmarca = marca.codmarca
17 ''' # Definição da consulta SQL a ser executada
18 cursor.execute(query) # Execução da consulta no banco de dados
19 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
20 relation = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os resul
21 display(relation) # Exibição do DataFrame Pandas contendo os resultados da consulta
22 cursor.close() # Fechamento do cursor após a conclusão da consulta
23 connection.close() # Fechamento da conexão após a consulta
```

| | modelo | marca |
|---|--------|------------|
| 0 | Ford | Ford |
| 1 | Fiat | Fiat |
| 2 | Onix | Chevrolet |
| 3 | Polo | Volkswagen |
| 4 | Kwid | Renault |


```

1 '''
2 Agora com Alias (apelido)
3 '''
4 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
5 connection = mysql.connector.connect(
6     host=host,
7     user=user,
8     password=password,
9     database=database,
10 )
11 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
12 query = '''
13 SELECT c.modelo, m.marca
14 FROM carro AS c, marca AS m
15 WHERE c.codmarca = m.codmarca
16 ''' # Definição da consulta SQL a ser executada
17 cursor.execute(query) # Execução da consulta no banco de dados
18 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
19 pedidos = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os result
20 display(pedidos) # Exibição do DataFrame Pandas contendo os resultados da consulta
21 cursor.close() # Fechamento do cursor após a conclusão da consulta
22 connection.close() # Fechamento da conexão após a consulta

```

| | modelo | marca |
|---|--------|------------|
| 0 | Ford | Ford |
| 1 | Fiat | Fiat |
| 2 | Onix | Chevrolet |
| 3 | Polo | Volkswagen |
| 4 | Kwid | Renault |

```

1 '''
2 Usando Inner Join
3 '''
4 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
5 connection = mysql.connector.connect(
6     host=host,
7     user=user,
8     password=password,
9     database=database,
10 )
11 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
12 query = '''
13 SELECT c.modelo, m.marca
14 FROM carro AS c
15 INNER JOIN marca AS m
16 ON c.codmarca = m.codmarca;
17 ''' # Definição da consulta SQL a ser executada
18 cursor.execute(query) # Execução da consulta no banco de dados
19 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
20 pedidos = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os result
21 display(pedidos) # Exibição do DataFrame Pandas contendo os resultados da consulta
22 cursor.close() # Fechamento do cursor após a conclusão da consulta
23 connection.close() # Fechamento da conexão após a consulta
24

```

| | modelo | marca |
|---|--------|------------|
| 0 | Ford | Ford |
| 1 | Fiat | Fiat |
| 2 | Onix | Chevrolet |
| 3 | Polo | Volkswagen |
| 4 | Kwid | Renault |

```
1 '''
2 Usando simplesmente Join
3 '''
4 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
5 connection = mysql.connector.connect(
6     host=host,
7     user=user,
8     password=password,
9     database=database,
10 )
11 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
12 query = '''
13 SELECT c.modelo, m.marca
14 FROM carro AS c
15 JOIN marca AS m
16 ON c.codmarca = m.codmarca;
17 ''' # Definição da consulta SQL a ser executada
18 cursor.execute(query) # Execução da consulta no banco de dados
19 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
20 pedidos = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os result
21 display(pedidos) # Exibição do DataFrame Pandas contendo os resultados da consulta
22 cursor.close() # Fechamento do cursor após a conclusão da consulta
23 connection.close() # Fechamento da conexão após a consulta
```

| | modelo | marca |
|---|--------|------------|
| 0 | Ford | Ford |
| 1 | Fiat | Fiat |
| 2 | Onix | Chevrolet |
| 3 | Polo | Volkswagen |
| 4 | Kwid | Renault |

```

1 '''
2 Agora, vamos a um exemplo mais complexo, em que desejamos visualizar o nome do
3 cliente e o modelo do carro alugado. Nesse caso, vamos precisar realizar dois comandos JOINS.
4 '''
5 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
6 connection = mysql.connector.connect(
7     host=host,
8     user=user,
9     password=password,
10    database=database,
11 )
12 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
13 query = '''
14 SELECT cl.nome, ca.modelo
15 FROM cliente AS cl
16 JOIN aluguel AS al
17   ON cl.codcliente = al.codcliente
18 JOIN carro AS ca
19   ON al.codcarro = ca.codcarro;
20 ''' # Definição da consulta SQL a ser executada
21 cursor.execute(query) # Execução da consulta no banco de dados
22 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
23 pedidos = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os result
24 display(pedidos) # Exibição do DataFrame Pandas contendo os resultados da consulta
25 cursor.close() # Fechamento do cursor após a conclusão da consulta
26 connection.close() # Fechamento da conexão após a consulta

```

| | nome | modelo |
|---|------------------|--------|
| 0 | Túlio Nascimento | Fiat |
| 1 | Bruna Pereira | Ford |
| 2 | Bruna Pereira | Ford |
| 3 | Bruna Pereira | Onix |
| 4 | Ana Silva | Polo |
| 5 | Ana Silva | Polo |
| 6 | Ana Silva | Ford |
| 7 | Lúcia Andrade | Fiat |
| 8 | Lúcia Andrade | Fiat |
| 9 | Túlio Nascimento | Ford |

```

1 # Left Join
2 '''
3 Outro tipo de JOIN que pode ser usado é o LEFT JOIN, o qual pode ser muito útil em
4 casos em que é necessário retornar dados de uma tabela que não tenha correspondentes
5 em outra tabela.
6 Sintaxe:
7 SELECT colunas
8 FROM tabela_esquerda
9 LEFT JOIN tabela_direita
10  ON tabela_esquerda.coluna_chave = tabela_direita.coluna_chave;
11 Quando executamos esse comando, são retornadas todas as linhas da tabela da esquerda,
12 mesmo que não haja correspondência na tabela da direita. Se não houver correspondência,
13 o SQL preencherá as colunas da tabela da direita com valores nulos
14
15
16 EXEMPLO
17 Para dar mais um exemplo, vamos recuperar, em nosso banco de dados, a lista de todos
18 os clientes e a data dos aluguéis realizados por eles. Porém, queremos, também, como
19 resultado aqueles clientes que não alugaram nenhum carro.
20 O resultado da nossa consulta fica:
21 '''

```

```

1 '''
2 EXEMPLO
3 Para dar mais um exemplo, vamos recuperar, em nosso banco de dados, a lista de todos
4 os clientes e a data dos aluguéis realizados por eles. Porém, queremos, também, como
5 resultado aqueles clientes que não alugaram nenhum carro.
6 O resultado da nossa consulta fica:
7 '''
8 # Estabelecimento da conexão com o banco de dados usando as informações fornecidas
9 connection = mysql.connector.connect(
10     host=host,
11     user=user,
12     password=password,
13     database=database,
14 )
15 cursor = connection.cursor() # Criação de um cursor para executar consultas SQL no banco de dados
16 query = '''
17 SELECT cl.nome, al.dataaluguel
18 FROM cliente AS cl
19 LEFT JOIN aluguel AS al
20 ON cl.codcliente = al.codcliente;
21 ''' # Definição da consulta SQL a ser executada
22 cursor.execute(query) # Execução da consulta no banco de dados
23 result = cursor.fetchall() # Recuperação de todos os resultados da consulta
24 pedidos = pd.DataFrame(result, columns=cursor.column_names) # Criação de um DataFrame Pandas usando os result
25 display(pedidos) # Exibição do DataFrame Pandas contendo os resultados da consulta
26 cursor.close() # Fechamento do cursor após a conclusão da consulta
27 connection.close() # Fechamento da conexão após a consulta

```

| | nome | dataaluguel |
|----|------------------|-------------|
| 0 | Ana Silva | 2023-04-05 |
| 1 | Ana Silva | 2023-04-06 |
| 2 | Ana Silva | 2023-04-07 |
| 3 | Bruna Pereira | 2023-04-02 |
| 4 | Bruna Pereira | 2023-04-03 |
| 5 | Bruna Pereira | 2023-04-04 |
| 6 | Túlio Nascimento | 2023-04-01 |
| 7 | Túlio Nascimento | 2023-04-10 |
| 8 | Fernando Souza | None |
| 9 | Lúcia Andrade | 2023-04-08 |
| 10 | Lúcia Andrade | 2023-04-09 |

```

1 # Right Join
2 '''
3 EXEMPLO

```