UFCD 10788- Fundamentos da linguagem SQL

Introdução às bases de dados

- O que é uma base de dados?
  - conjunto de dados organizados e estruturados
  - os dados podem relacionar-se de forma a ser acessados e manipulados
  - O termo base de dados refere-se aos ficheiros onde se encontram os dados
  - Os dados são geridos por um Sistema de Gestão de Base de Dados (SGBD)

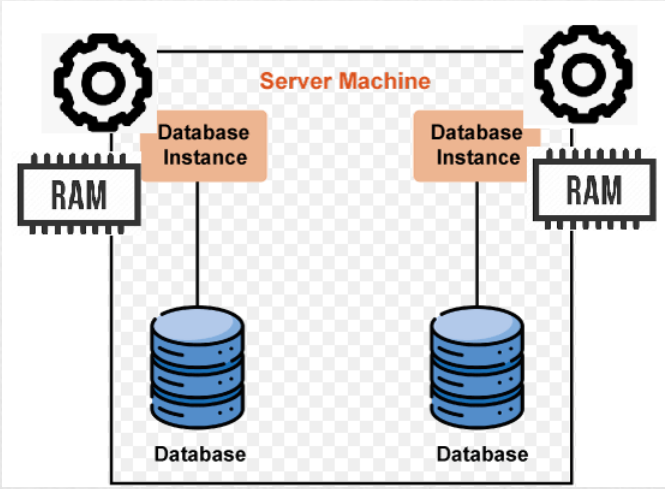
Tipos de bases de dados

- Bases de dados relacionais (RDBMS)
  - Utilizam tabelas organizadas com linhas e colunas para armazenar dados
  - As tabelas podem estar relacionadas
  - Exemplos: MySQL, PostgreSQL, SQL Server, Oracle, SQLite
- Bases de dados não relacionais (NoSQL)
  - São usadas para dados não estruturados ou semi-estruturados, como documentos JSON, gráficos ou grandes volumes de dados não tabulares
  - Exemplos: MongoDB, Cassandra, Redis
- Bases de dados em nuvem
  - Armazenam dados na nuvem, permitindo acesso remoto e escalabilidade fácil
  - Exemplos: Google Cloud, AWS RDS, Azure SQL.

Tipos de bases de dados

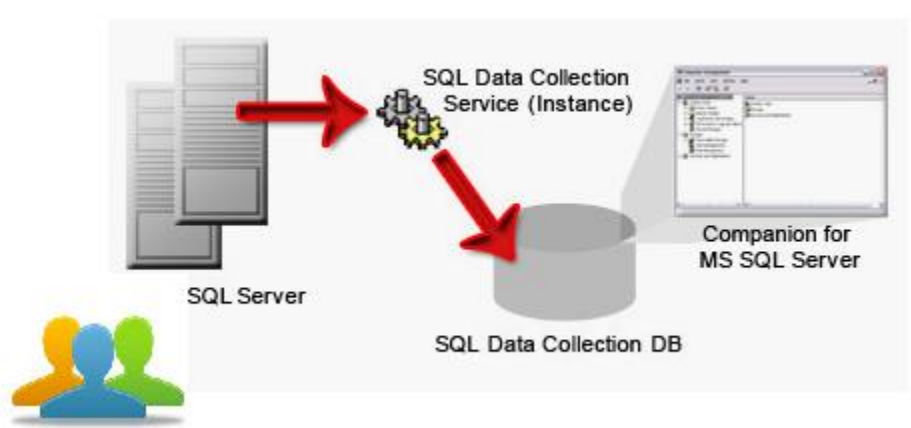
Como funcionam as bases de dados

- Instância de um Sistema de Gerenciamento de Bases de Dados (SGBD)
  - A instância de um SGBD refere-se ao conjunto de processos e de memória que são responsáveis pela gestão de uma ou mais bases de dados



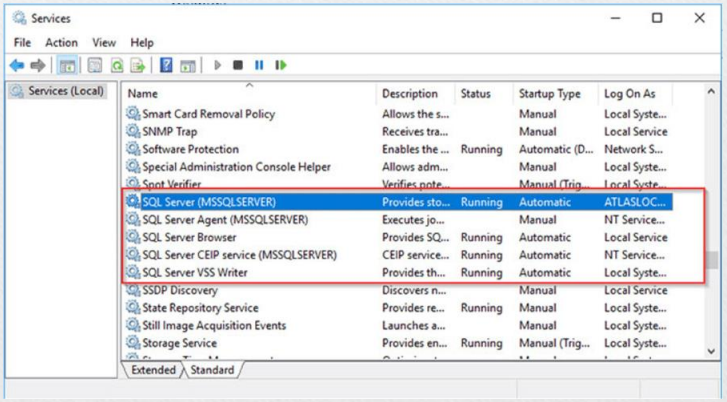
Como funcionam as bases de dados

- Instância de um Sistema de Gerenciamento de Bases de Dados (SGBD)
  - A instância do SGBD executa processos que são responsáveis pela execução das operações no banco de dados (inserção, consulta, atualização e exclusão de dados)



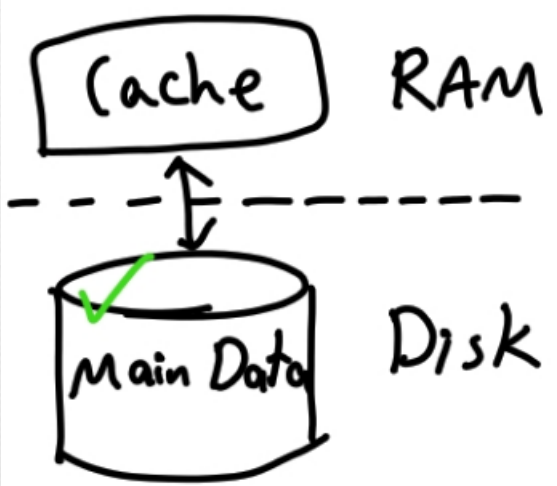
Como funcionam as bases de dados

- Sistema de Gerenciamento de Bases de Dados (SGBD)
  - A instância do SGBD executa processos que são responsáveis pela execução das operações no banco de dados (inserção, consulta, atualização e eliminação de dados)

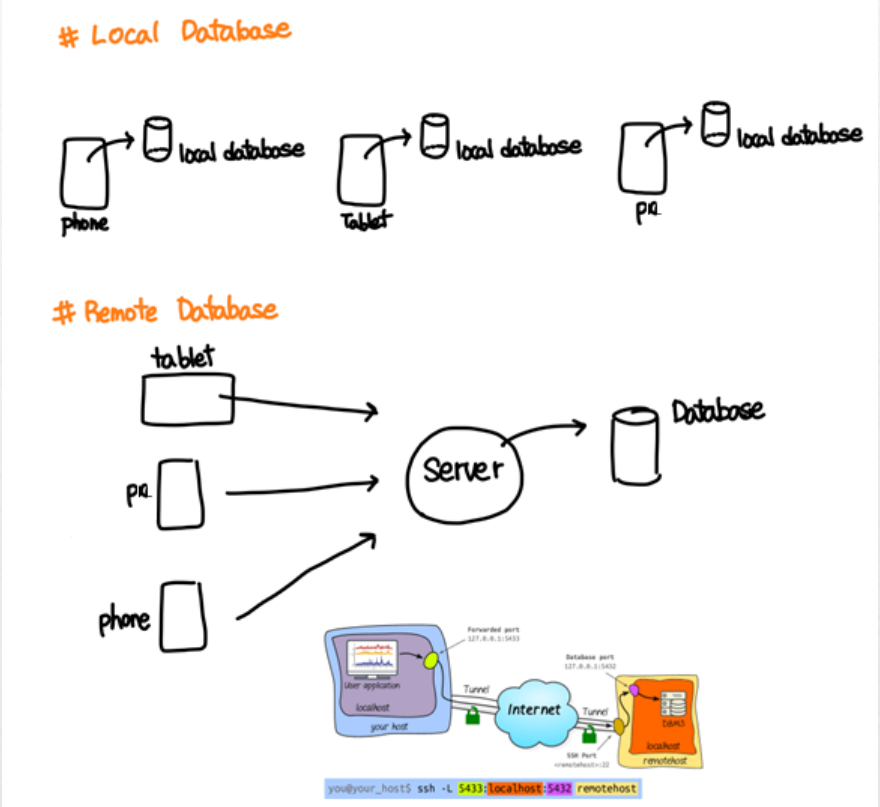


Como funcionam as bases de dados

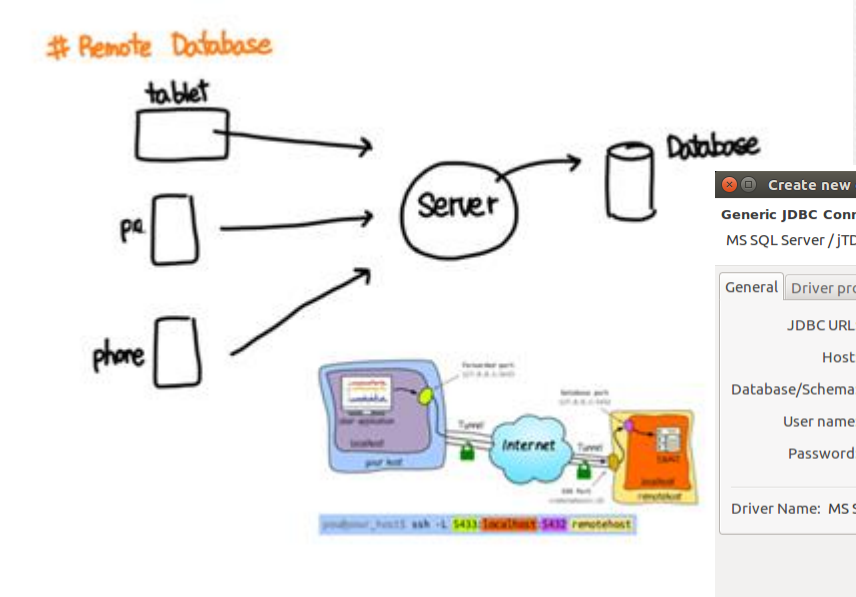
- Sistema de Gerenciamento de Bases de Dados (SGBD)
  - A instância do SGBD aloca e efetua a gestão de memória para otimizar o acesso aos dados, como a *cache* de dados e *buffers* para melhorar a performance das consultas.

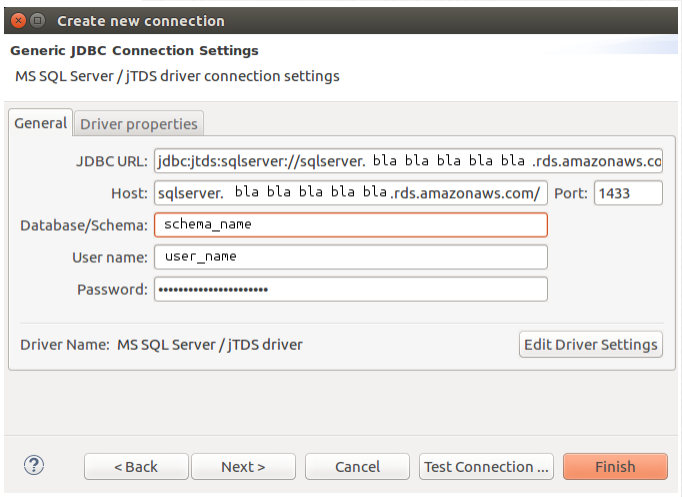


Como funcionam as bases de dados

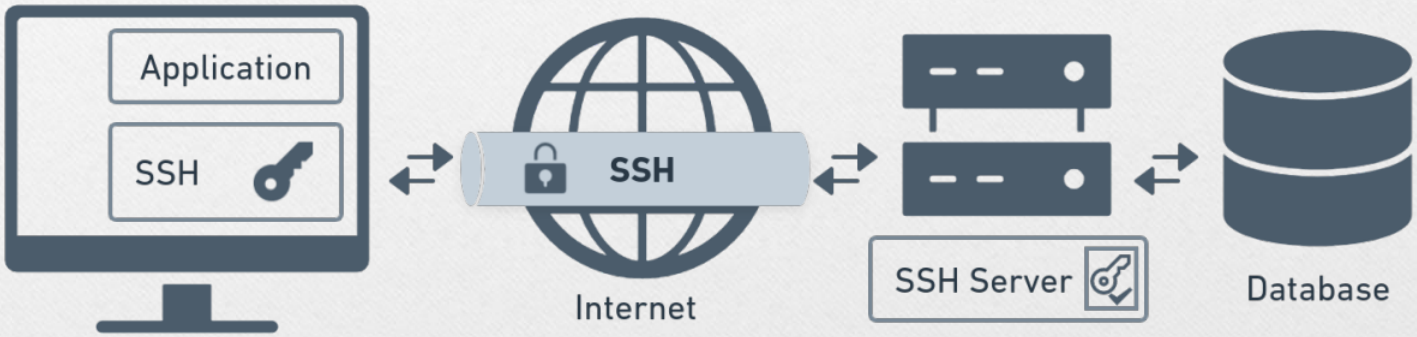


Como funcionam as bases de dados



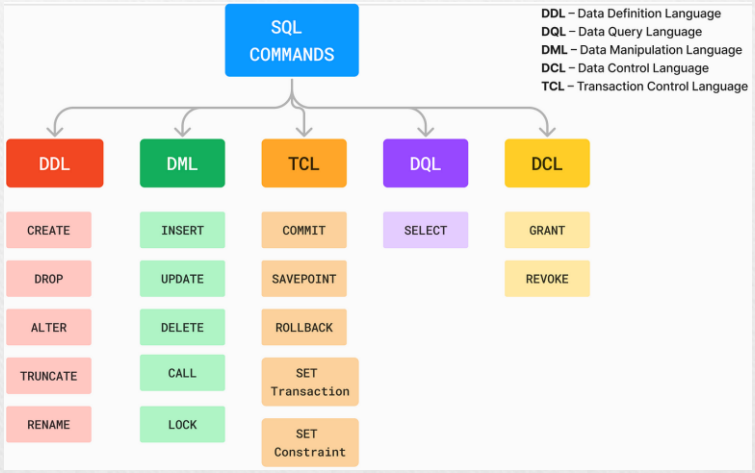


Como funcionam as bases de dados

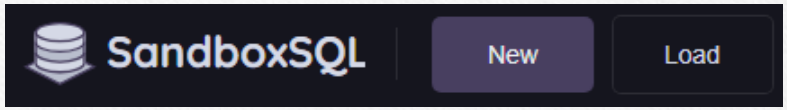


Como funcionam as bases de dados

- Consultas em bases de dados
  - As consultas e manipulação de dados são feitas em SQL (*Structured Query Language*)



IDE de desenvolvimento



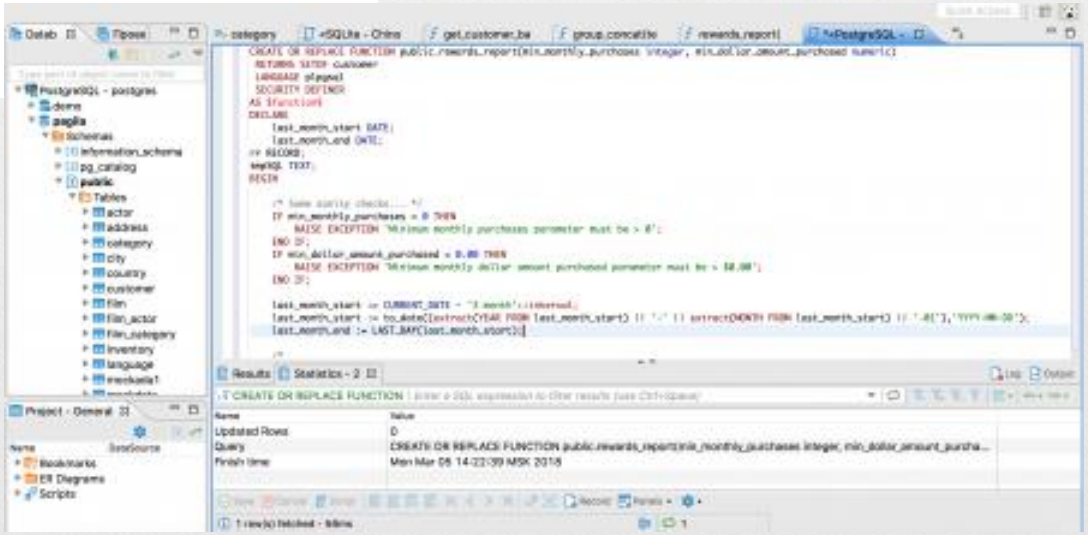
www.sandboxsql.com

Google Colab

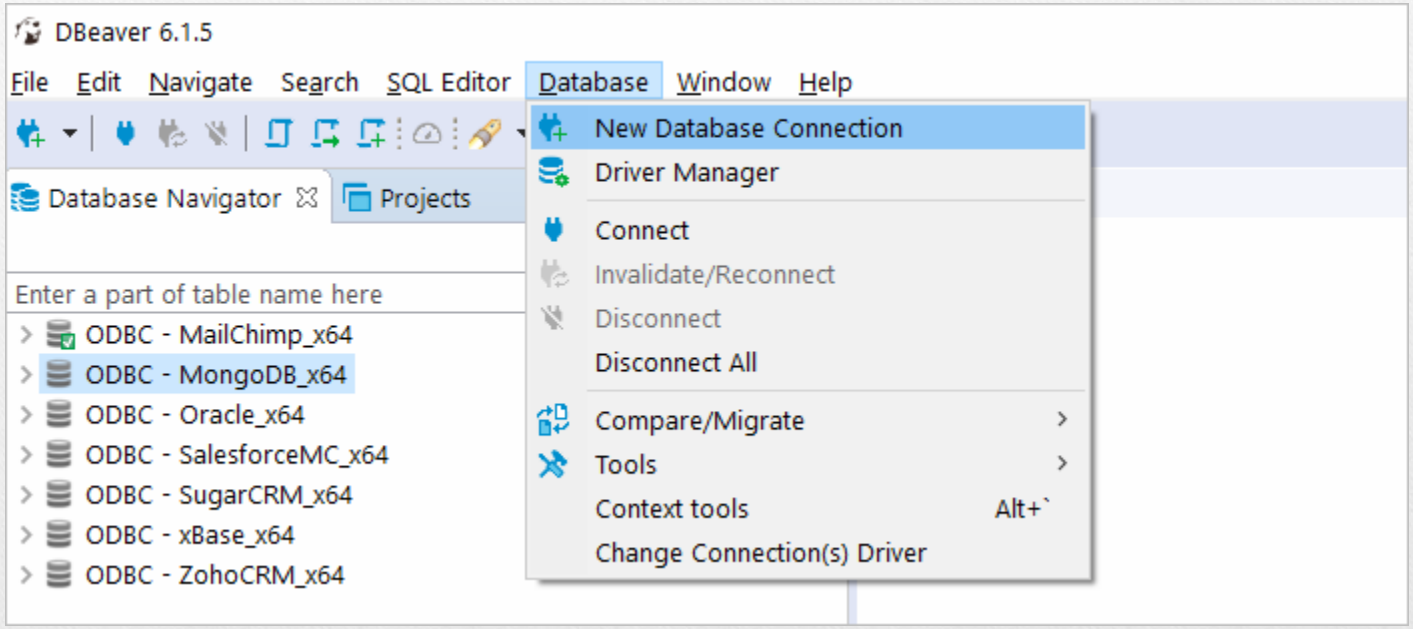
Jupyter Notebook

IDE de desenvolvimento

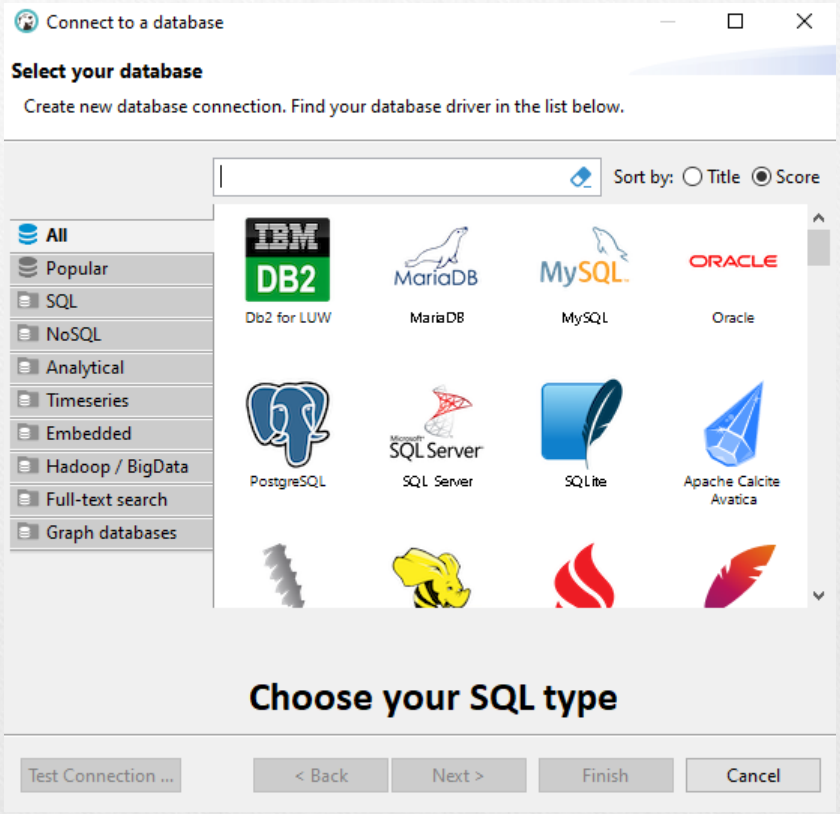
DBeaver



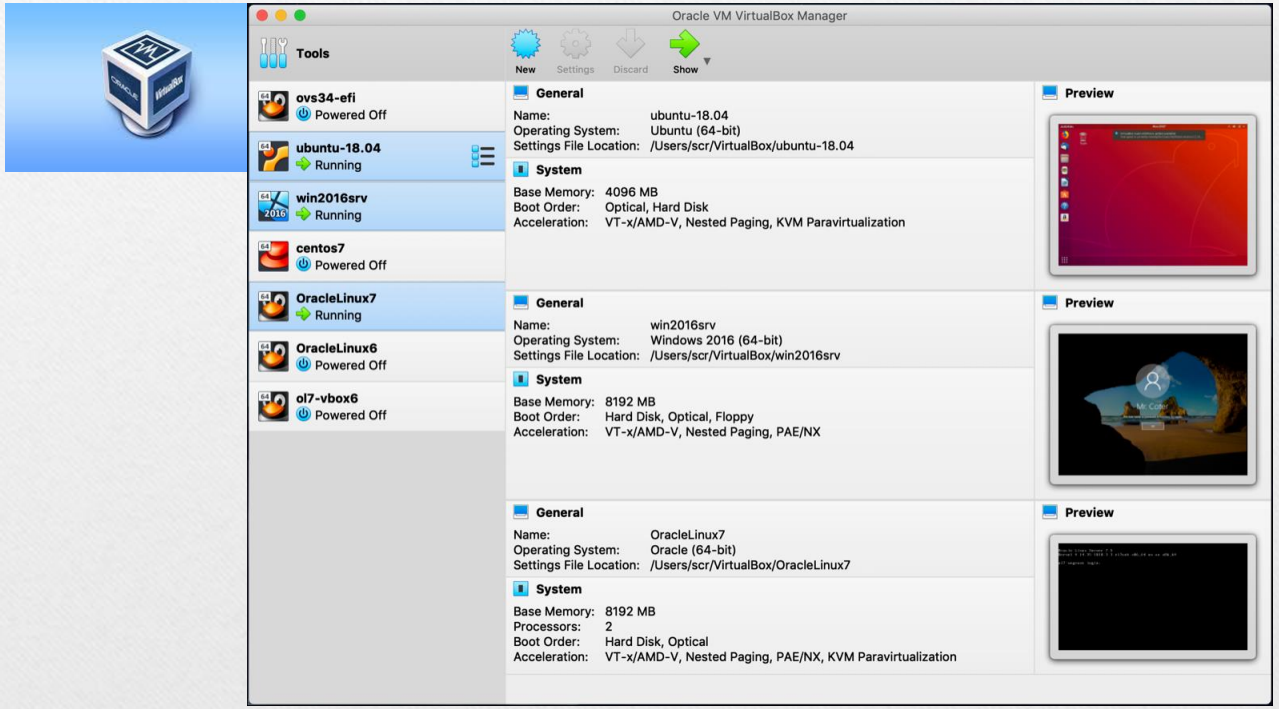
Conexão a uma base de dados SQLite



Conexão a uma base de dados SQLite



IDE de desenvolvimento



SQL

Google Colab

```
[ ] !pip install jupysql
```

Mostrar saída oculta

```
[ ] %load_ext sql
```

```
[ ] %sql sqlite:///Hospital.sqlite
```

Mostrar saída oculta

```
%%sql
SELECT * FROM Patient
```

Running query in 'sqlite:///Hospital.sqlite'

SSN	Name	Address	Phone	InsuranceID	PCP
100000001	John Smith	42 Foobar Lane	555-0256	68476213	1
100000002	Grace Ritchie	37 Snafu Drive	555-0512	36546321	2
100000003	Random J. Patient	101 Omgbbq Street	555-1204	65465421	2
100000004	Dennis Doe	1100 Foobaz Avenue	555-2048	68421879	3

SQL

- Data Definition Language (DDL)
  - *Create/alter/delete* tabelas e respetivos atributos
- Data Manipulation Language (DML)
  - *Insert/delete/Update* linhas nas tabelas

Estrutura de uma Base de Dados Relacional

- Os dados são armazenados em tabelas
- As chaves estrangeiras são usadas para relacionar tabelas
- Uma das propriedades das bases de dados relacionais é a integridade referencial, que garante que os dados sejam consistentes.
- O uso do SQL oferece uma grande flexibilidade para consultar, filtrar e agregar dados de variadas formas

Estrutura de uma Base de Dados Relacional

- Tabelas: Armazenam dados de forma organizada, com cada tabela representando uma entidade ou objeto.
- Colunas: Representam os atributos ou características dos dados
- Linhas: Cada linha contém um registo de dados (exemplo: um cliente ou produto).
- Chave primária (*Primary Key*): Uma coluna ou conjunto de colunas que identifica de forma única cada registo de uma tabela.
- Chave estrangeira (*Foreign Key*): Estabelece uma relação entre duas tabelas, ligando uma chave primária de uma tabela a outra tabela.

Estrutura de uma Base de Dados Relacional

Tabelas no SQL

Tabela

Atributos

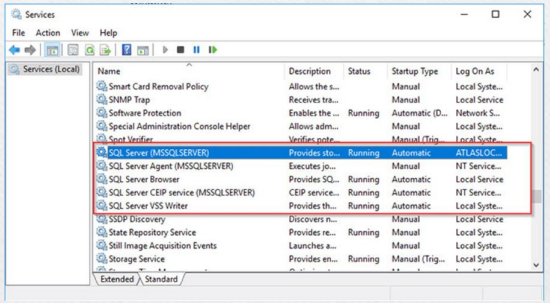
Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Tuplos/Registos/Linhas

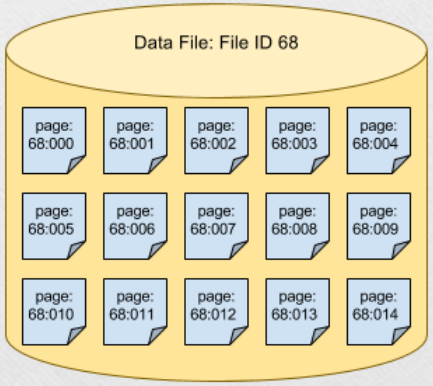
Estrutura de uma Base de Dados Relacional

	A	B	C	D	E	F
1	Employee Table					
2	Id	Name	Salary	Age	Gender	Dept
3	1	Anne	95000	43	Female	Sales
4	2	Claire	80000	35	Female	Analytics
5	3	David	70000	45	Male	Operations
6	4	Phil	85000	37	Male	Sales
7	5	Ray	90000	45	Female	Analytics
8	6	Rachel	60000	27	Female	Sales
9	7	Bob	80000	34	Male	Operations



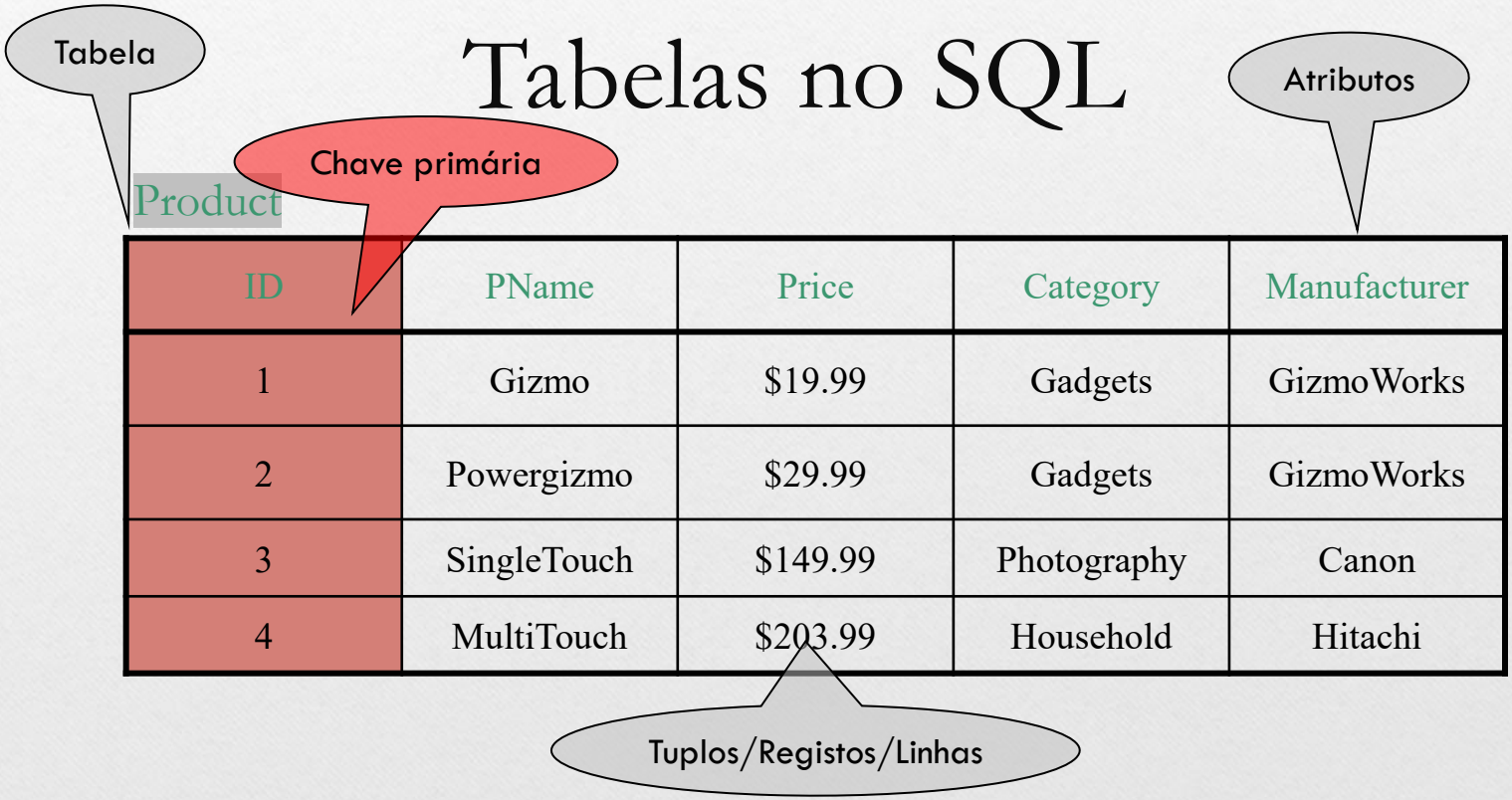

```
SELECT  *
FROM Employee
WHERE Gender='Female'
```

Id	Name	Salary	Age	Gender	Dept
1	Anne	95000	43	Female	Sales
2	Claire	80000	35	Female	Analytics
5	Ray	90000	45	Female	Analytics
6	Rachel	60000	27	Female	Sales



Estrutura de uma Base de Dados Relacional

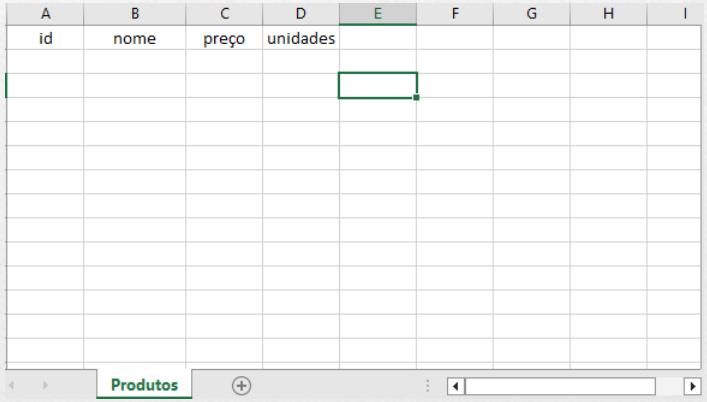
Tabelas no SQL



ID	PName	Price	Category	Manufacturer
1	Gizmo	$19.99	Gadgets	GizmoWorks
2	Powergizmo	$29.99	Gadgets	GizmoWorks
3	SingleTouch	$149.99	Photography	Canon
4	MultiTouch	$203.99	Household	Hitachi

Definição da tabela

```
CREATE TABLE products (
    ID          INT,
    Pname       VARCHAR(128),
    price       DECIMAL(10,2),
    category    VARCHAR(32),
    Pname       VARCHAR(128)
);
```



Eliminação da tabela

```
DROP TABLE products
```

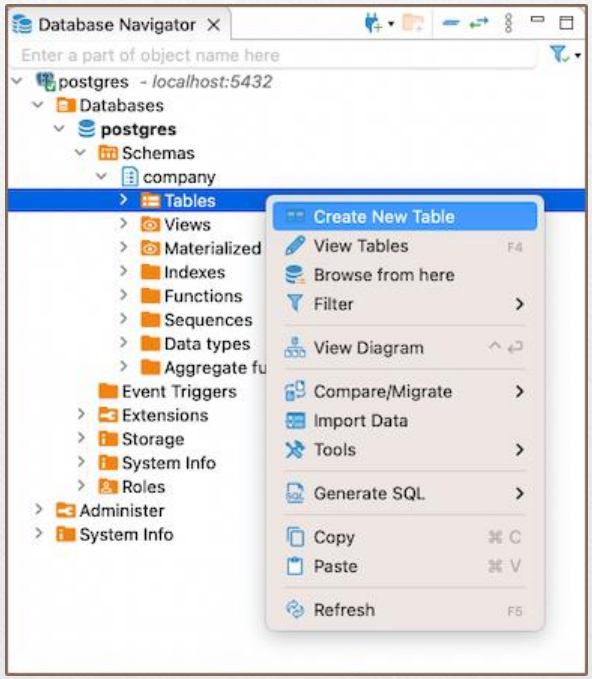
Definição de chave primária

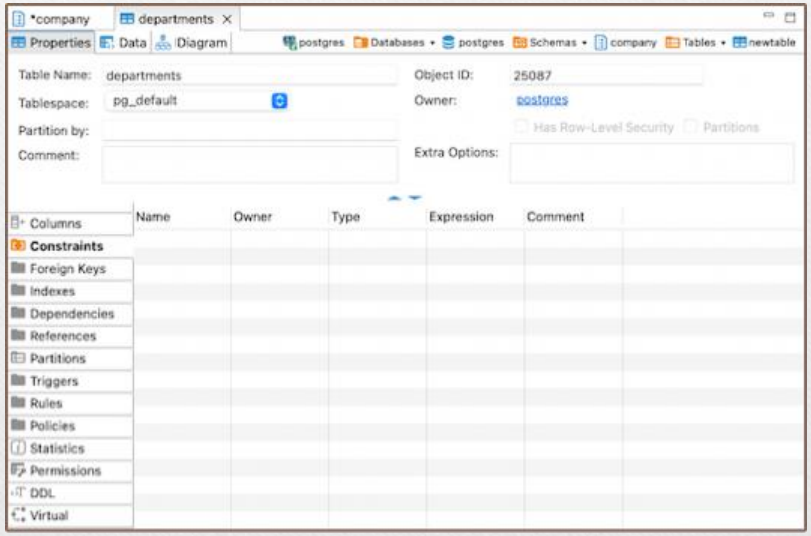
```
CREATE TABLE products (
    ID INT AUTO_INCREMENT PRIMARY KEY,
    Pname    VARCHAR(128),
    price       DECIMAL(10,2),
    category VARCHAR(32),
    Pname     VARCHAR(128)
);
```

- O *AUTO_INCREMENT* cria automaticamente valores únicos e sequenciais para a chave primária, evitando conflitos de ID
- A chave primária também poderia ser definida através do campo *Pname*

Definição de chave primária

- DBeaver Documentation





SQL

- Data Definition Language (DDL)
  - *create*/*alter*/*delete* tabelas e respetivos atributos
- Data Manipulation Language (DML)
  - *insert*/*delete*/*update* linhas nas tabelas

Consulta/*Query* SQL

```
SELECT  <attributes>
FROM     <one or more relations>
WHERE  <conditions>
```

```
SELECT  Name, Units, Price, discount
FROM    Produts
WHERE   units = 100
    AND price = 50
    AND rate > 5
```

Eliminações(*delete*)

```
DELETE    FROM    PURCHASE
WHERE     seller = 'Joe'    AND
                  product = 'Brooklyn Bridge'
```

- Usar a cláusula WHERE corretamente para evitar apagar registos acidentalmente
- Também é possível uso de *sub-queries* para identificar os registos a eliminar

Atualizações(update)

```
UPDATE   PRODUCT
SET      price = price/2, units=100
WHERE    Product.name  ='Xbox'
```

- Usar a cláusula WHERE corretamente para evitar alterar registos acidentalmente
- É possível atualizar mais do que uma coluna na mesma instrução

Inserções(*insert*)

```
INSERT INTO R(A1,…., An) VALUES (v1,…., vn)
```

Exemplo: Inserção de um nova compra:

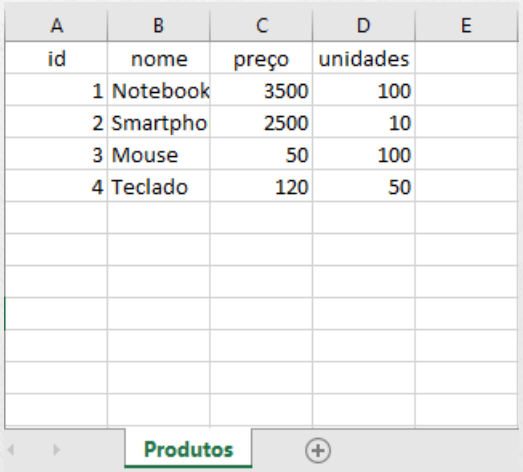
```
INSERT INTO Purchase(buyer, seller, product, store)
        VALUES ('Joe', 'Fred', 'wakeup-clock-espresso-machine',
                'The Sharper Image')
```

- Aos atributos não especificados é assignado o valor NULL
- Não existe qualquer ordem para a especificação dos atributos

Inserções(*insert*)

```
[8] %%sql
    INSERT INTO produtos (numero,nome, preco, unidades) VALUES (1,'Notebook', 3500.00, 10);
    INSERT INTO produtos (numero,nome, preco, unidades) VALUES (2,'Smartphone', 2500.00, 20);
    INSERT INTO produtos (numero,nome, preco, unidades) VALUES (3,'Mouse', 50.00, 100);
    INSERT INTO produtos (numero,nome, preco, unidades) VALUES (4,'Teclado', 120.00, 50);
```

numero	nome	preco	unidades
1	Notebook	3500.0	10
2	Smartphone	2500.0	20
3	Mouse	50.0	100
4	Teclado	120.0	50
1	Notebook	3500.0	10
2	Smartphone	2500.0	20
3	Mouse	50.0	100
4	Teclado	120.0	50



	A	B	C	D	E
1	id	nome	preço	unidades	
2	1	Notebook	3500	100	
3	2	Smartpho	2500	10	
4	3	Mouse	50	100	
5	4	Teclado	120	50	

Produtos

Consultas com uma tabela

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

```
SELECT  *
FROM    Product
WHERE   category='Gadgets'
```

"seleção"

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks

Consultas com uma tabela

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

```
SELECT   PName, Price, Manufacturer
FROM     Product
WHERE    Price > 100
```

"seleção" e "projeção"

PName	Price	Manufacturer
SingleTouch	$149.99	Canon
MultiTouch	$203.99	Hitachi

Chaves primárias e chaves estrangeiras

Company

CName	StockPrice	Country
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

Chave

Chave estrangeira

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Valores distintos

```
SELECT   DISTINCT category
FROM     Product
```

Category
Gadgets
Photography
Household

```
SELECT   category
FROM     Product
```

Category
Gadgets
Gadgets
Photography
Household

Valores Null(“nulos”)

- Sempre que não existe valor pode ser definida o NULL
- Pode ter vários significados:
  - O valor não existe
  - O valor existe mas é desconhecido
  - O valor não é aplicável
- Em cada atributo é especificado se pode ser nulo (atributo anulável) ou não
- Como o SQL lida com tabelas que possuem valores NULLs?

Valores Null("nulos")

- x IS NULL
- x IS NOT NULL

```
SELECT *
FROM    Person
WHERE  age < 25  OR  age >= 25 OR age IS
NULL
```

Valores Null("nulos")

- x IS NULL
- x IS NOT NULL

```
SELECT *
FROM    Person
WHERE  age < 25  OR  age >= 25 OR age IS
NULL
```

Funções SQL

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

```
SELECT   COUNT(*) as Contagem
FROM     Product
WHERE    Price > 100
```

Contagem
2

COUNT(*) : efetua a contagem de registos/linhas na tabela

Funções SQL

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

```
SELECT  MAX(Price) as Preco_maximo,
        MIN(Price) as Preco_minimo
FROM    Produot
```

Preco_maximo	Preco_minimo
$203.99	$19.99

- **MAX(Price)** : calcula o valor máximo da coluna "Price"
- **MIN(Price)** : calcula o valor minmo da coluna "Price"

O operador *LIKE*

```
SELECT   *
FROM     Products
WHERE    PName LIKE '%gizmo%'
```

- s **LIKE** p: correspondência de padrões em *strings*
- p pode conter dois símbolos especiais:
  - % = qualquer sequência de caracteres
  - _ = um único caracter

O operador *LIKE*

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

```
SELECT  Pname,Price
FROM    Product
WHERE Pname LIKE '%h'
```

SingleTouch	$149.99
MultiTouch	$203.99

```
SELECT  Pname,Price
FROM    Product
WHERE Pname LIKE '%iz%'
```

Gizmo	$19.99
Powergizmo	$29.99

```
SELECT  Pname,Price
FROM    Product
WHERE Pname LIKE '%m_'
```

Gizmo	$19.99
Powergizmo	$29.99

Junções

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT  *
FROM    Product, Company
WHERE   Manufacturer = CName
```

Gizmo	$19.99	Gadgets	GizmoWorks
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

GizmoWorks	100	Canada
GizmoWorks	25	USA
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

Junções

Product

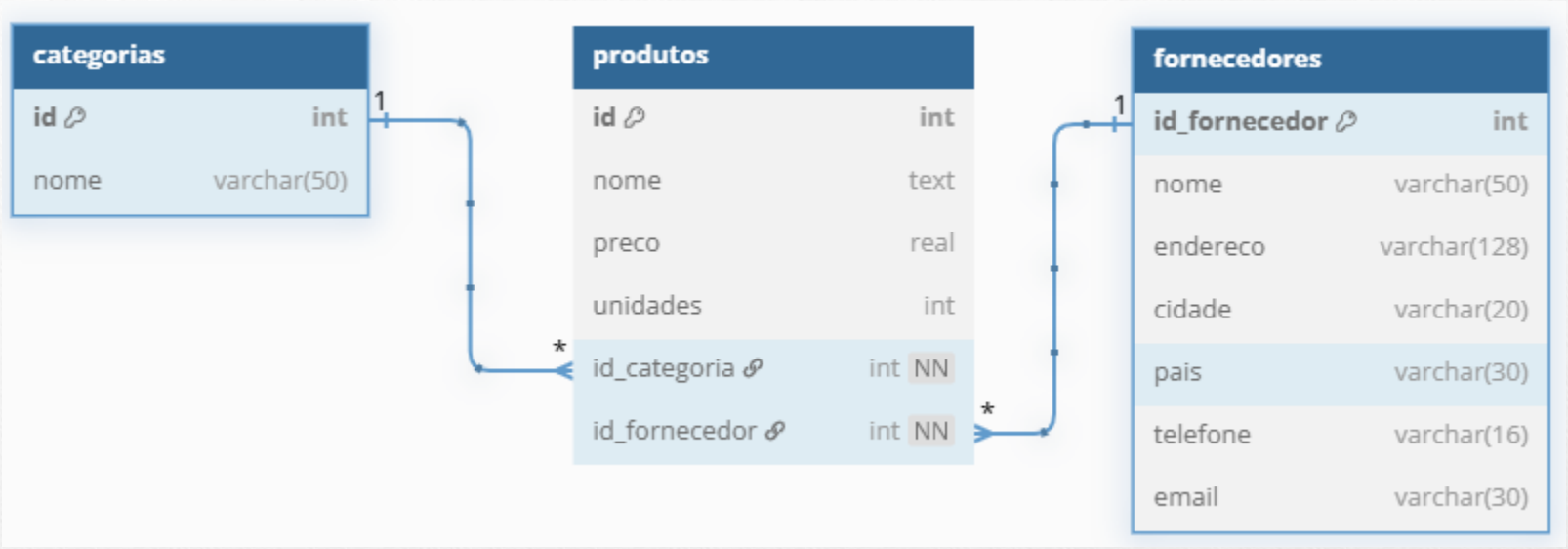
PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Company

Cname	StockPrice	Country
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT  *
FROM    Product, Company
WHERE   Manufacturer = CName
```

Modelo de dados “Produto”



dbdiagram.io

draw.io

Junções

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Company

Cname	StockPrice	Country
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT   PName, Price
FROM     Product, Company
WHERE    Manufacturer=CName AND Country='Japan'
         AND Price <= 200
```

PName	Price
SingleTouch	$149.99

Ordenação dos resultados

```
SELECT   pname, price, manufacturer
FROM     Product
WHERE    category='gizmo' AND price > 50
ORDER BY  price, pname DESC
```

- No caso de valores idênticos, é necessário a utilização de outro atributo no *ORDER BY coluna1, coluna2 etc.*
- Por defeito a ordenação é ascendente
- Caso seja pretendida uma ordenação descendente é necessário utilizar *DESC*

Junções

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
Canon	65	Japan
Hitachi	15	Japan

```
SELECT  *
FROM    Product, Company
WHERE   Manufacturer = CName
```

Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

GizmoWorks	100	Canada
GizmoWorks	100	Canada
Canon	65	Japan
Hitachi	15	Japan

Agrupamentos("group by")

- O "group by" serve para agrupar dados e aplicar funções de agregação sobre esse agrupamentos
- Mecanismo de agregação relacional que reduz múltiplas linhas numa única linha por grupo, baseado em critérios

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
Multi 2.0	$499.99	Household	Hitachi
MultiTouch	$203.99	Household	Hitachi

Agrupamentos(“group by”)

1. Agrupar dados por uma(s) coluna(s)

   Exemplo: agrupar vendas por vendedor, ou alunos por curso.

2. Transforma linhas em grupos

   Cada valor único da coluna agrupada é um grupo.

3. Utilização das funções de agregação

   Como SUM(), AVG(), COUNT(), MAX(), MIN() — que trabalham sobre grupos.

1. Agrupa antes de filtrar com HAVING."

   WHERE filtra linhas; HAVING filtra grupos

Agrupamentos

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
Multi 2.0	$499.99	Household	Hitachi
MultiTouch	$203.99	Household	Hitachi


```
SELECT category
FROM   Product
GROUP BY category
```

Category
Gadgets
Photography
Household

Agrupamentos

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
Multi 2.0	$499.99	Household	Hitachi
MultiTouch	$203.99	Household	Hitachi


```
SELECT category,COUNT(*)
FROM  Product
GROUP BY category
```

Category	Count(*)
Gadgets	2
Photography	1
Household	2

Agrupamentos

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
Multi 2.0	$499.99	Household	Hitachi
MultiTouch	$203.99	Household	Hitachi


```
SELECT category,COUNT(*),MIN(Price)
FROM  Product
GROUP BY category
```

Category	Count(*)	Min(Price)
Gadgets	2	$19.99
Photography	1	$149.99
Household	2	$203.99

Cláusula "*having*"

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
Multi 2.0	$499.99	Household	Hitachi
MultiTouch	$203.99	Household	Hitachi


```
SELECT category, COUNT(*), MIN(Price)
FROM   Product
GROUP BY category

HAVING COUNT(*) > 1
```

Category	Count(*)	Min(Price)
Gadgets	2	$19.99
~~Photography~~	~~1~~	~~$149.99~~
Household	2	$203.99

Cláusula "*having*"

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
Multi 2.0	$499.99	Household	Hitachi
MultiTouch	$203.99	Household	Hitachi


```
SELECT category, COUNT(*), MIN(Price)
FROM   Product
GROUP BY category

HAVING MIN(Price) > 20
```

Category	Count(*)	Min(Price)
~~Gadgets~~	~~2~~	~~$19.99~~
Photography	1	$149.99
Household	2	$203.99

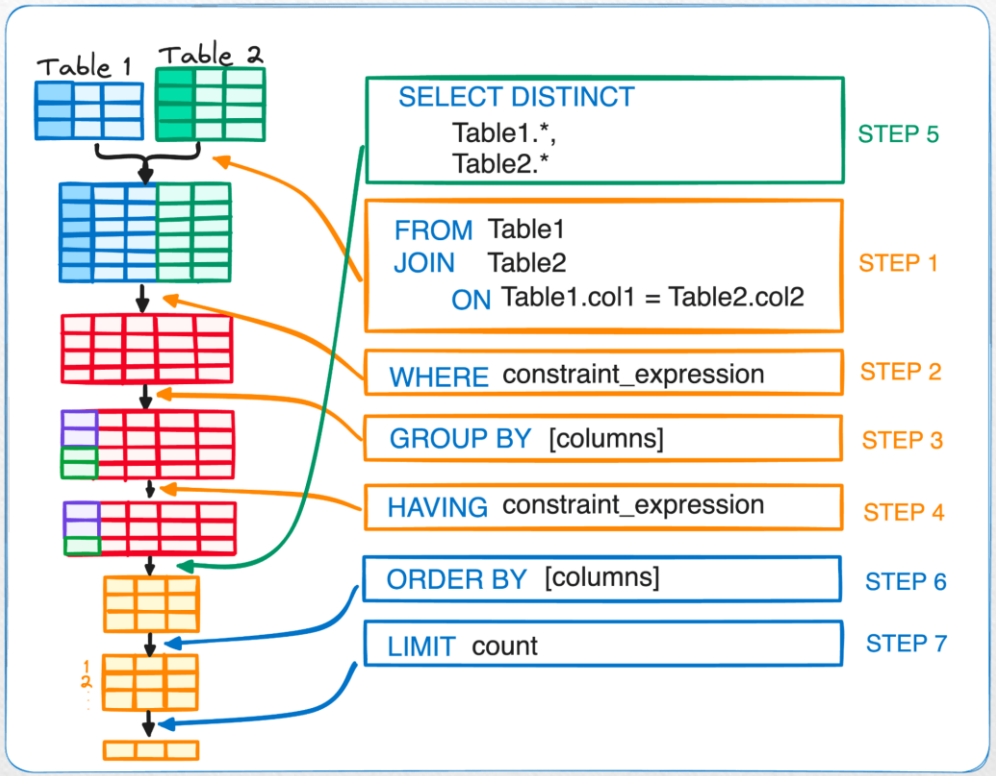
Cláusula "*having*"

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
Multi 2.0	$499.99	Household	Hitachi
MultiTouch	$203.99	Household	Hitachi

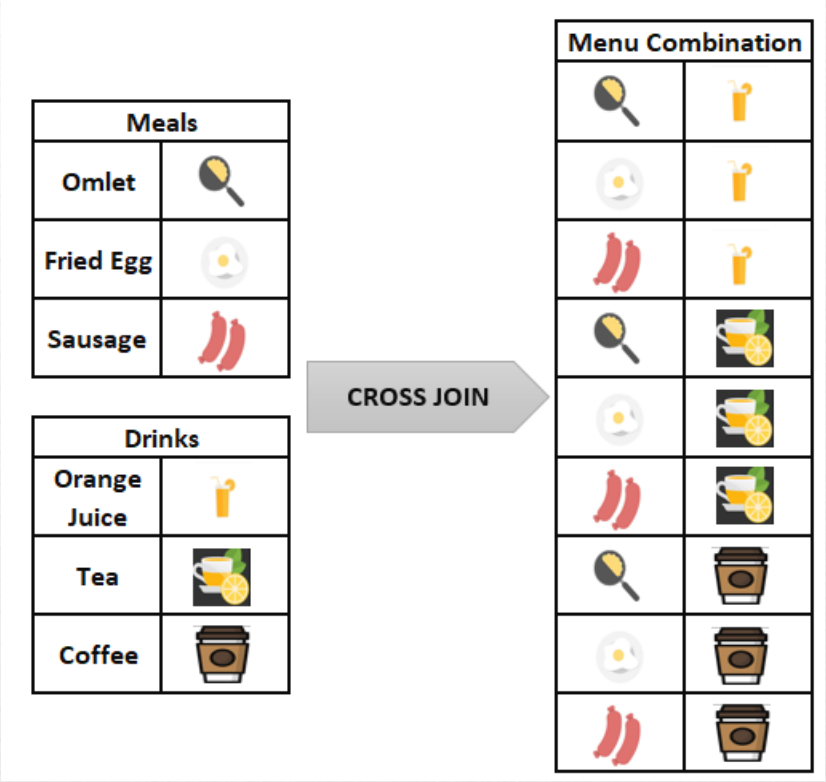

```
SELECT category, COUNT(*), MIN(Price)
FROM   Product

WHERE Price > 19.99
GROUP BY category

HAVING MIN(Price) > 20
```

Category	Count(*)	Min(Price)
Gadgets	1	$29.99
Photography	1	$149.99
Household	2	$203.99

Ordem da execução da consulta



Produto cartesiano



Sub-queries "IN"

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT *
FROM Product
WHERE Manufacturer In (
```

Canon
Hitachi

```
)
```

SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Sub-queries "IN"

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

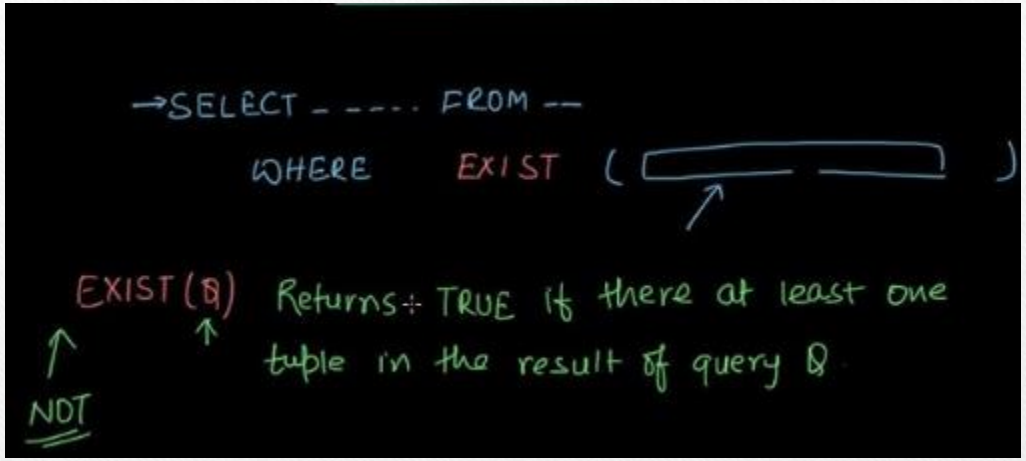
Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT *
FROM Product
WHERE Manufacturer In (SELECT Cname FROM Company WHERE Cname='Japan')
```

SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

Sub-queries "EXISTS"




```
SELECT  *
FROM  Company c
WHERE  c.Cname In (SELECT manufacturer FROM Produts )
```

```
SELECT  *
FROM Company c
WHERE  EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = c.Cname)
```

Sub-queries "EXISTS"

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon

Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = c.Cname)
```

GizmoWorks	100	Canada

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = 'GizmoWorks')
```

True

Sub-queries "EXISTS"

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon

Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = c.Cname)
```

GizmoWorks	25	USA

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = 'GizmoWorks')
```

True

Sub-queries "EXISTS"

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon

Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = c.Cname)
```

Canon	65	Japan

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = 'Canon')
```

True

Sub-queries "EXISTS"

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon

Company

Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = c.Cname)
```

Hitachi	15	Japan

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = 'Hitachi')
```

False

Sub-queries "EXISTS"

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon

Company

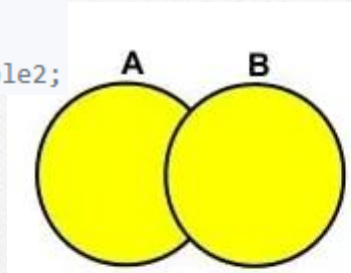
Cname	StockPrice	Country
GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

```
SELECT *
FROM Company c
WHERE EXISTS (SELECT * FROM Product p WHERE p. Manufacturer = c.Cname)
```

GizmoWorks	100	Canada
GizmoWorks	25	USA
Canon	65	Japan

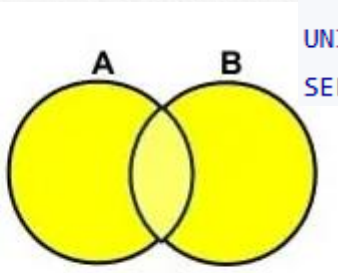
Operadores de grupo

```
SELECT * FROM Table1
UNION
SELECT * FROM Table2;
```

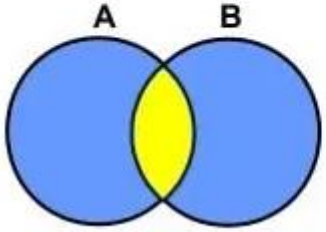


UNION

```
SELECT * FROM Table1
UNION ALL
SELECT * FROM Table2;
```

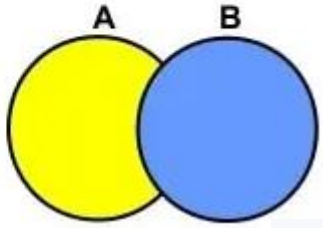


UNION ALL



INTERSECT

```
SELECT * FROM Table1
INTERSECT
SELECT * FROM Table2;
```



EXCEPT/MINUS

```
SELECT * FROM Table1
MINUS
SELECT * FROM Table2;
```

Operadores de grupo

Intersect

customers

CUSTOMER_ID	NAME
1	Amelia
2	Isla
3	Jessica
4	Lily

contacts

CONTACT_ID	NAME
1	Amelia
2	Olivia
3	Isla
4	Emily

```
SELECT name
FROM  Customers
intersect

SELECT name
FROM  Contacts
```

NAME
Amelia
Isla

Operadores de grupo

Except

customers

CUSTOMER_ID	NAME
1	Amelia
2	Isla
3	Jessica
4	Lily

contacts

CONTACT_ID	NAME
1	Amelia
2	Olivia
3	Isla
4	Emily

```
SELECT name
FROM  Customers
except

SELECT name
FROM  Contacts
```

NAME
Jessica
Lily

Operadores de grupo

Intersect

customers

CUSTOMER_ID	NAME
1	Amelia
2	Isla
3	Jessica
4	Lily

contacts

CONTACT_ID	NAME
1	Amelia
2	Olivia
3	Isla
4	Emily

```
SELECT name
FROM  Customers
union

SELECT name
FROM  Contacts
```

NAME
Amelia
Emily
Isla
Jessica
Lily
Olivia

Operadores de grupo

Intersect

customers

CUSTOMER_ID	NAME
1	Amelia
2	Isla
3	Jessica
4	Lily

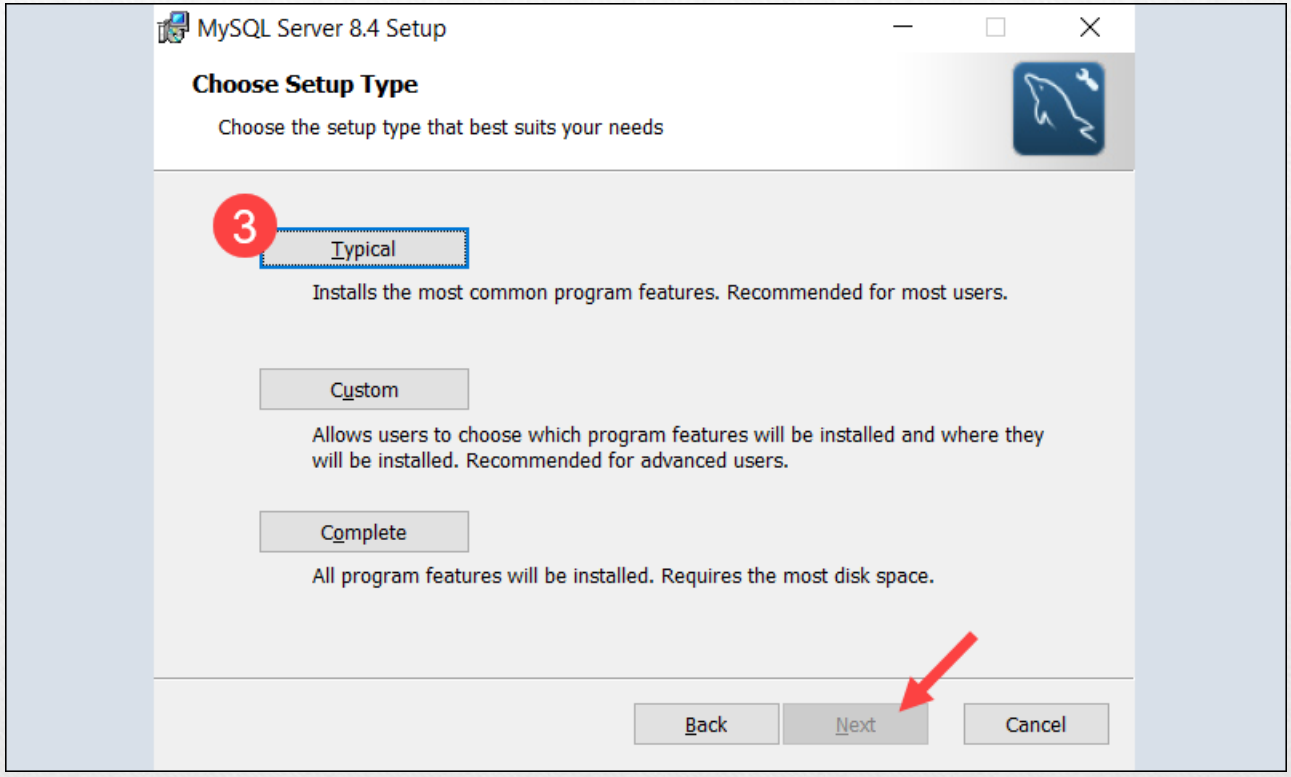
contacts

CONTACT_ID	NAME
1	Amelia
2	Olivia
3	Isla
4	Emily

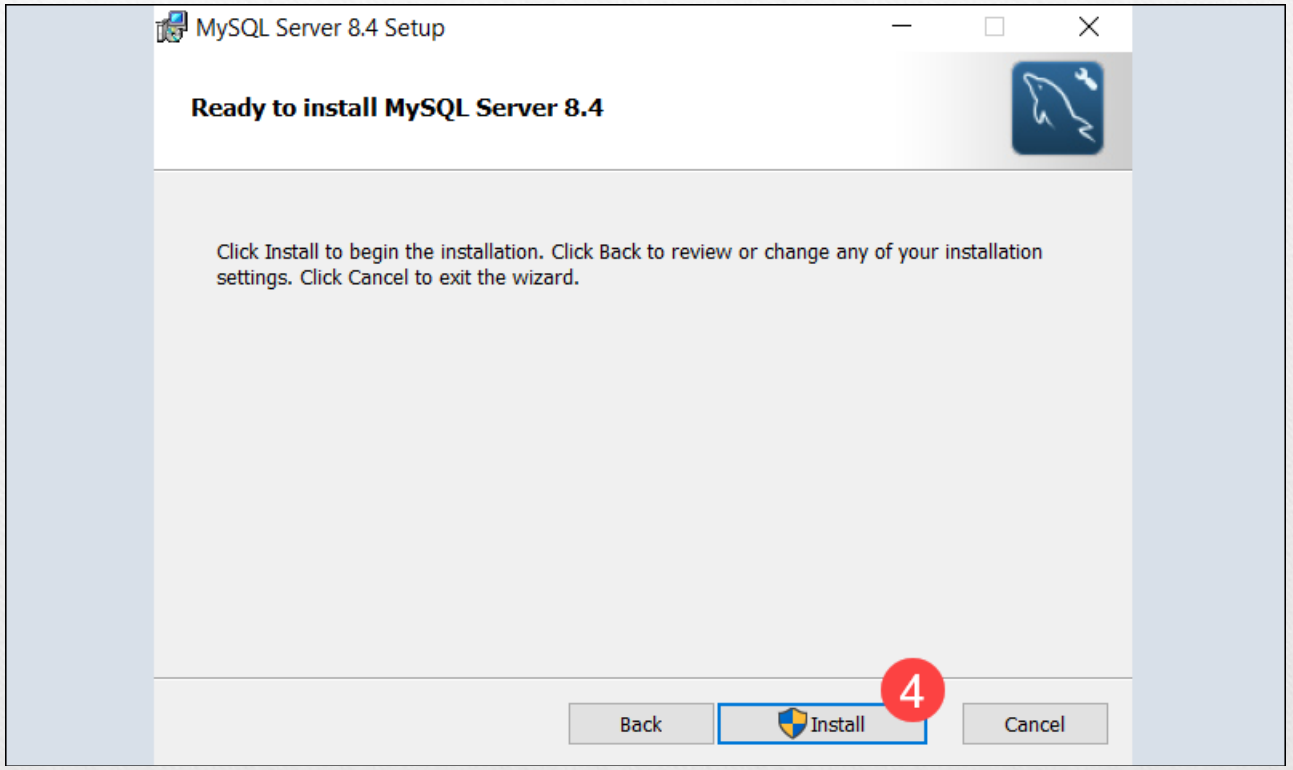
```
SELECT name
FROM  Customers
union all

SELECT name
FROM  Contacts
```

NAME
Amelia
Amelia
Emily
Isla
Isla
Jessica
Lily
Olivia

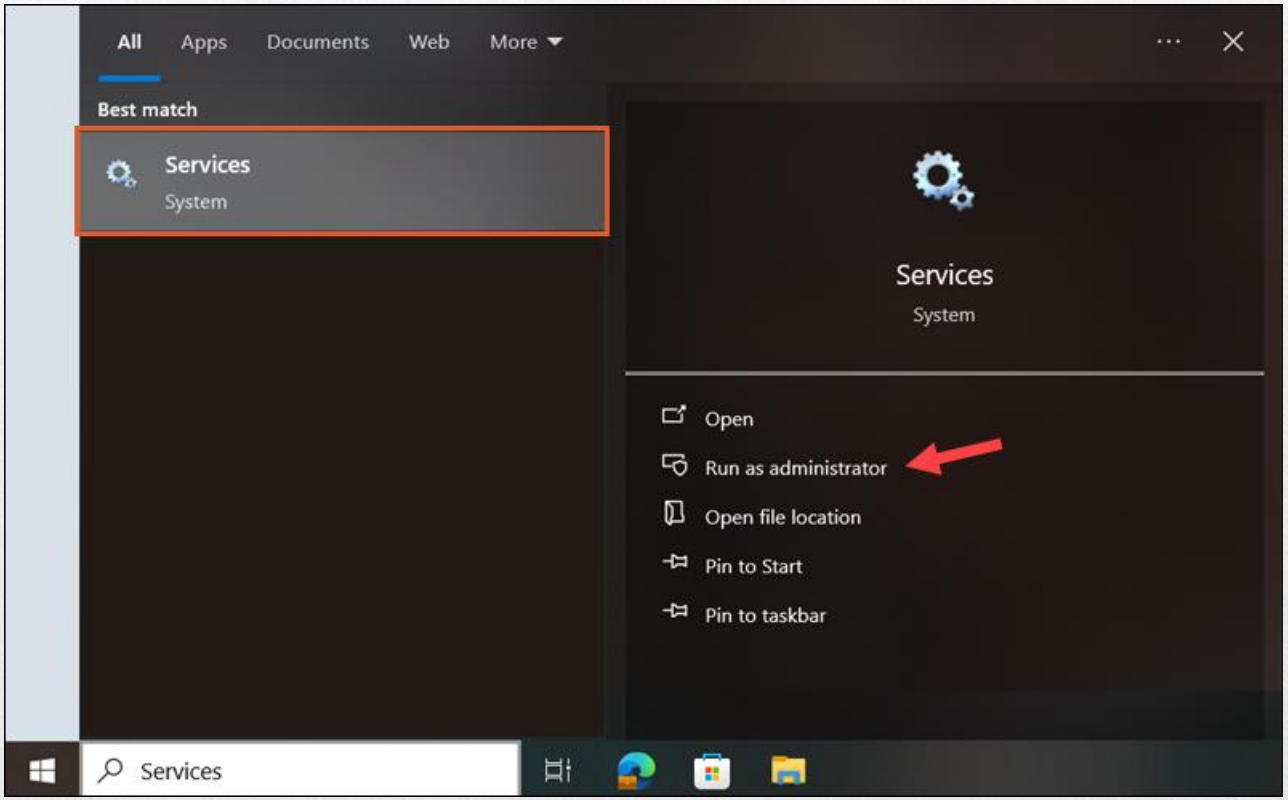
Instalação MySQL-Windows



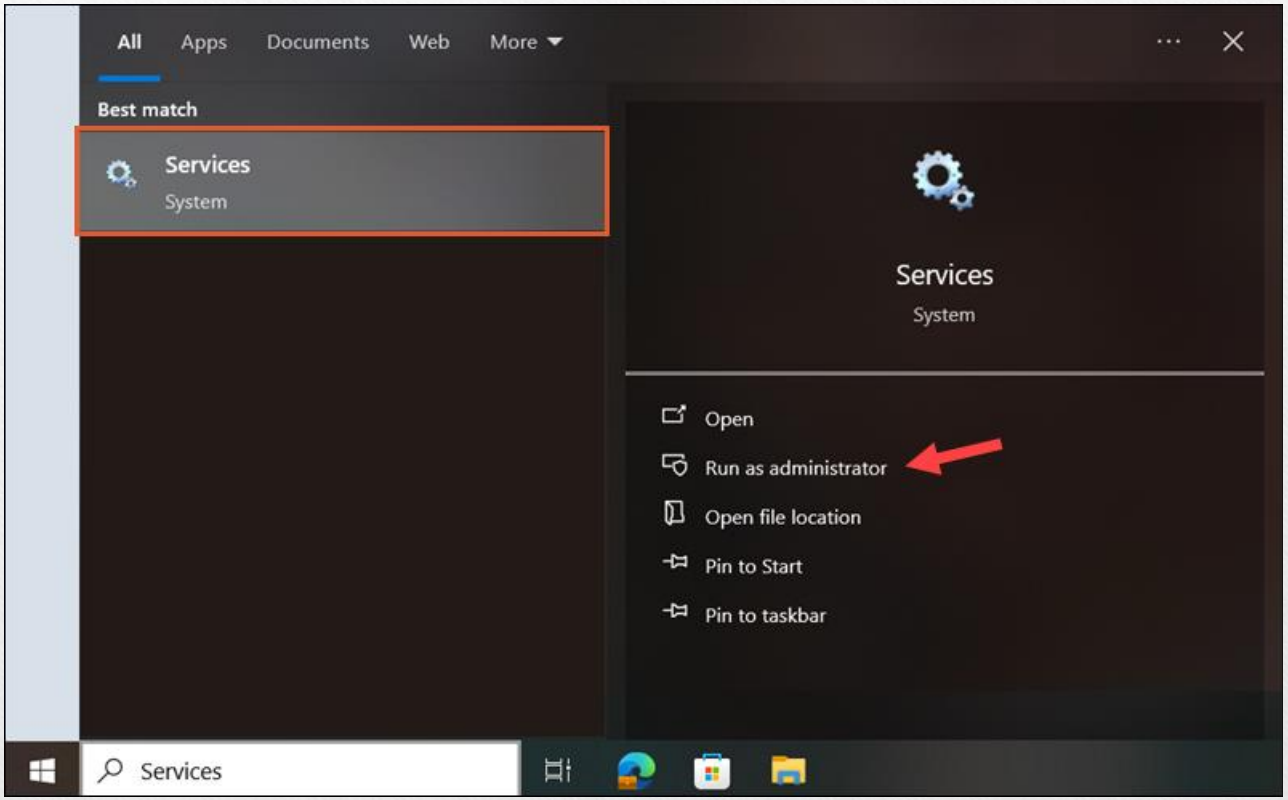
Instalação MySQL-Windows



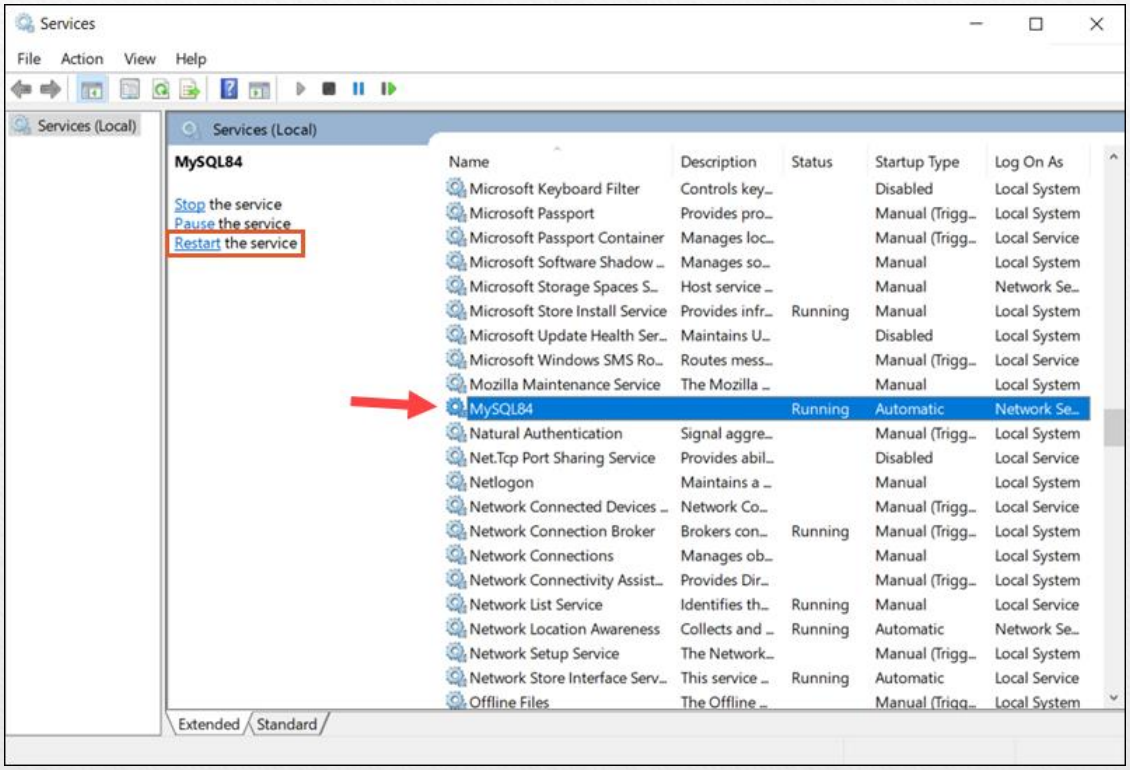
Instalação MySQL-Windows



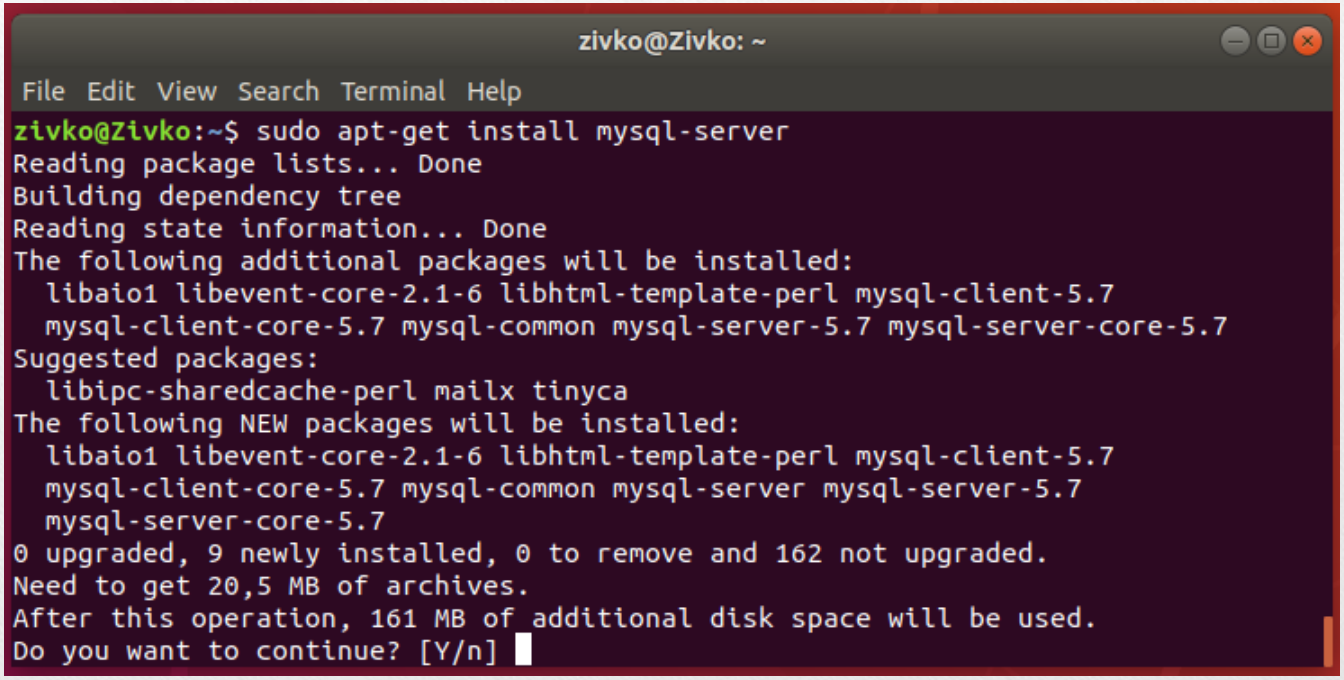
Instalação MySQL-Windows



Instalação MySQL-Windows



Instalação MySQL-Linux



Instalação MySQL-Linux