[👾 Criar Bd 2](#_94y6yq82q2xl)

[👾 Excluir Bd 2](#_29kx82mn16u6)

[👾 Transferindo dados 2](#_88im6proin7x)

[✍️ Querys importantes: 2](#_6hknt7coqvvo)

[**💫 Tabelas 2**](#_8qn9xv5p7ksj)

[👾 Criando tabelas e colunas 2](#_8t1a40yda9bj)

[👾 Adicionar dados na tabela 3](#_vdx377p77jiw)

[✍️ Importantes: 3](#_i2o230vzjt2d)

[👾 Excluir Tabela 3](#_3w9jc11y5ps7)

[👾 Renomear Tabela 4](#_e4aka6yn0im)

[**💫 Colunas 4**](#_ke54fc1az3jn)

[👾 Adicionar uma nova coluna 4](#_bu097bd0mbsf)

[👾 Excluir coluna 4](#_obdc72s4ox79)

[👾 Modificar o tipo e característica da coluna criada 4](#_tjpjvsgk935v)

[👾 Modificar nome da coluna 4](#_rimla4xa7cl2)

[**💫 Linhas 5**](#_fuve2qxxjqkk)

[👾 Adicionar dados na linha ou modificar dados incorretos 5](#_1zcb81sem7tp)

[👾 Modificar mais linhas com os mesmos dados 5](#_uzf32c4avc48)

[👾 Apagar linhas 5](#_4vqocacws9z0)

[👾 Apagar TODAS as linhas de uma tabela 6](#_yqjatuo0prr2)

[✍️ Gerenciando Cópias de Segurança: 6](#_ptyq44d0qyxg)

[**💫 SELECT 6**](#_h8uvneb1emnh)

[👾 Ordenar de acordo com uma coluna 6](#_732wk9yedwo0)

[👾 Ordenar descendente (de baixo para cima) 6](#_o2vupcw6amcv)

[👾 Ordenar ascendente (de cima para baixo) 6](#_zgmne48btovs)

[👾 Filtrar as colunas ordenando colunas 6](#_qcmbdzg2fe8x)

[👾 Filtrar Linhas 7](#_mnxyr8mtlyli)

[👾 Usando operadores 7](#_wp7v79g8q33p)

[👾 Entre um e outro (BETWEEN, AND) 7](#_8zrqtjz1cwcm)

[👾 Where com mais dados (IN = em) 7](#_xtjvn4nc4ig)

[👾 Combinando testes (AND, OR) 7](#_sywii6hisgw5)

[👾 Puxar um dado 8](#_9vptww5q4izg)

[👾 Puxar dois ou mais dados 8](#_rmivhsqzun1j)

[👾 Filtro de texto 8](#_lzqw1u1vaicc)

[👾 Puxar dados sem repetir as informações 8](#_8du3926cp15h)

[👾 Função de agregação - Selecionar ou totalizar informações 8](#_14935slpvgz6)

[👾 Agrupar as informações 9](#_pdoz232qvtxc)

[👾 Agrupar informação especifica 9](#_dkt9ctn1mb4i)

[👾 Maior entre os dados 9](#_7gvqnf29uf0o)

[👾 Menor entre os dados 9](#_2wu8h48i2e63)

[👾 Somar 9](#_adzrcqlkxrzj)

[👾 Media 9](#_rsf48wpccgrp)

Banco de dados é composto por tabelas > colunas > campos

Ao escolher os campos devemos definir seu tipo, chamamos de Tipos Primitivos. São basicamente os tipo de dados que iremos usar, por exemplo numéricos, data/hora, strings, espacial e etc.

[Tipos de dados MySql (Link complementar)](https://www.rlsystem.com.br/tipos-dados-mysql/)

[Tipos de dados MySql 2 (Link complementar)](https://medium.com/mandabugs/mysql-tipos-de-dados-introdu%C3%A7%C3%A3o-e-dados-num%C3%A9ricos-1-de-3-a6e48fb5e1d3)



Tipos básicos mais utilizados

### **👾 Criar Bd**

*create database nome\_do\_bd;*

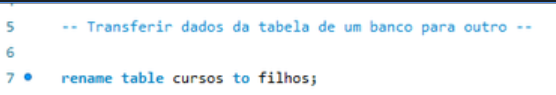
### **👾 Excluir Bd**

*drop database nome\_do\_bd;*

### **👾 Transferindo dados**

Não dá para renomear um banco de dados, dessa forma criamos um bd novo e transferimos as tabelas e dados do antigo bd para a nova.

*rename table nome\_da\_tabela to nome\_do\_bd\_novo;*



#### **✍️ Querys importantes:**

*use nome\_do\_banco;*

Abre o Banco

# **💫 Tabelas**

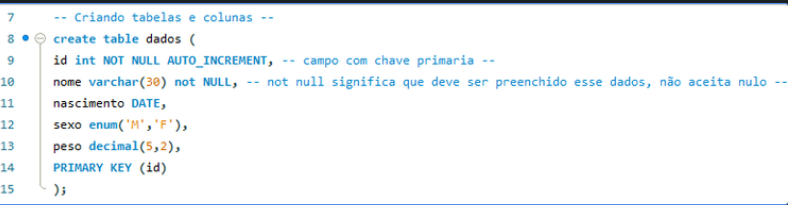
### **👾 Criando tabelas e colunas**

Ao criar a tabela já adicionamos as colunas com seus respectivos tipos

NOT NULL = significa que deverá ser preenchido, não aceita que fique nulo.

Auto\_increment = é um tipo onde se é auto incrementado de forma automática, nesse caso foi o Id

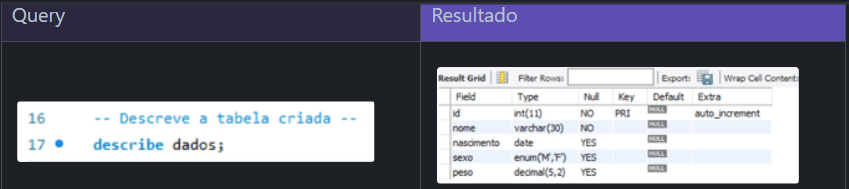
Campos com Chave Primária = A Primary Key é a chave que **é utilizada como identificador único da tabela, sendo representada por aquele campo (ou campos),que não receberá valores repetidos**. É como se fosse um CPF ou matricula, um dado que não se repete.



*describe nome\_da\_tabela;*

Apresenta a tabela criada com suas características e tipos

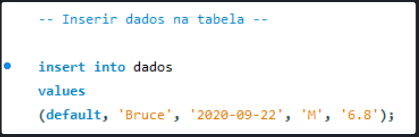
(podendo ser apenas **desc)**

****

### **👾 Adicionar dados na tabela**

*insert into nome\_da\_tabela*

*values (dados\_novos);*

**

Nesse caso poderia também adicionar os campos que serão inseridos, exemplo:

*insert into dados*

*(id, nome, nascimento, sexo, peso)*

*values*

*(default, ‘Bruce', ‘2020-09-22’, ‘M’, '6.8’);*

#### **✍️ Importantes:**

**Default =** irá preencher sozinho

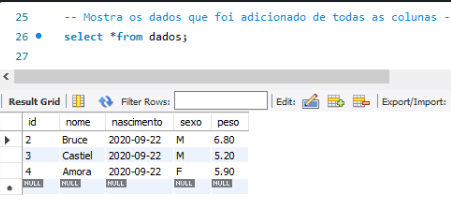
Posso colocar sem o id pois criei o comando de ir automático e sequencial, ou posso  
colocar default.

*select \*from nome\_da\_tabela;*

Mostra os dados que foram adicionados de todas as colunas

Caso eu queira dados somente de uma coluna a query seria:

*select nome, sexo from dados;*

**

### **👾 Excluir Tabela**

Só vai criar uma tabela nova no bd se ele não existir IF NOT EXISTS

Só vai apagar uma tabela ou um banco de dados se ele existir IF EXISTS

*drop table if exists nome\_da\_tabela;*

*create table if not exists cursos (*

*nome varchar(30)*

*);*

### **👾 Renomear Tabela**

*alter table nome\_da\_tabela*

*rename to nome\_da\_nova\_tabela;*

*alter table pessoa*

*rename to gafanhoto;*

# **💫 Colunas**

### **👾 Adicionar uma nova coluna**

Adicionar a descrição e características dessa coluna

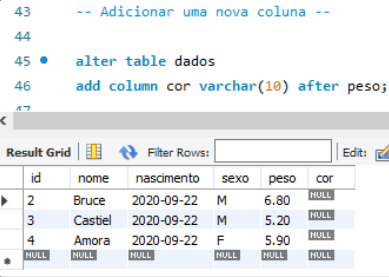
Depois AFTER

Primeiro FIRST

Caso seja a ultima coluna não é necessário adicionar nada

*alter table nome\_da\_tabela*

*add column nome\_da\_coluna;*

**

Nesse exemplo já foi add o tipo e onde ela irá ser adicionada, no caso foi após (after) da coluna peso, mas como seria a ultima coluna não seria necessário colocar o after poderia deixar sem nada.

### **👾 Excluir coluna**

*alter table nome\_da\_tabela*

*drop colunm nome\_da\_coluna;*

### **👾 Modificar o tipo e característica da coluna criada**

*alter table nome\_da\_tabela*

*modify column nome\_da\_coluna;*

*alter table pessoa*

*modify column profissao varchar(20) not null default '';*

### **👾 Modificar nome da coluna**

*ALTER TABLE nome\_da\_tabela*

*CHANGE column nome\_atual\_da\_coluna novo\_nome\_da\_coluna tipo\_de\_dados;*

*ALTER TABLE clientes*

*CHANGE idade idade\_cliente INT;*

# **💫 Linhas**

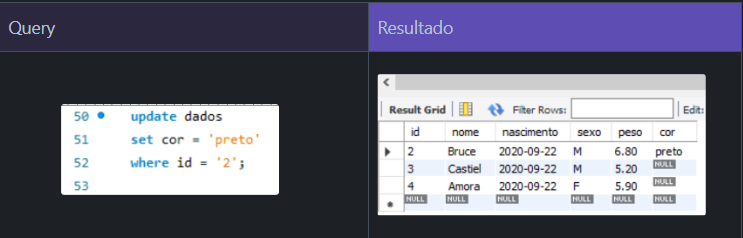
### **👾 Adicionar dados na linha ou modificar dados incorretos**

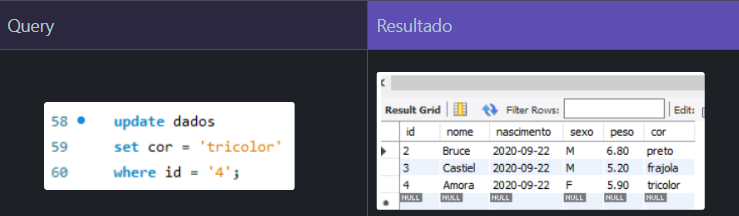
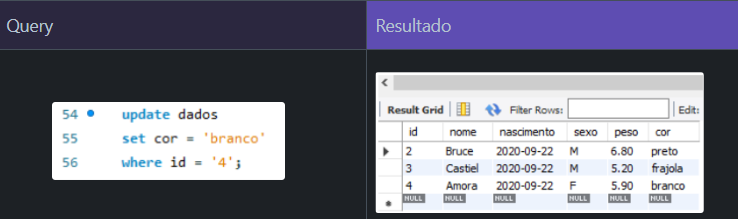
atualize = update   
configure = set  
onde = where  
limit 1; = define quantas linhas poderão ser afetadas, limita a ação do meu comando

*update nome\_da\_tabela*

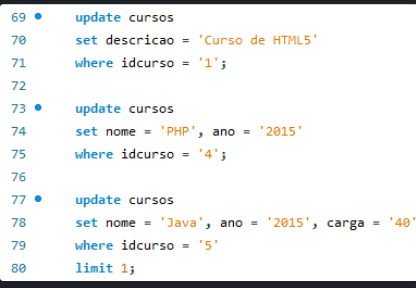
*set nome\_da\_coluna*

*where nome\_da\_linha*





Outros exemplos:



### **👾 Modificar mais linhas com os mesmos dados**

update cursos

set ano = '2050', carga = '800'

where ano = '2018';

### **👾 Apagar linhas**

delete from cursos

where idcurso= '8';

delete from cursos

where ano='2050'

limit 2;

delete from cursos

where idcurso= '10';

### **👾 Apagar TODAS as linhas de uma tabela**

truncate nome\_da\_tabela;

#### **✍️ Gerenciando Cópias de Segurança:**

1. server - data export - seleciona o banco e as tabelas que quer exportar
2. dump structure and data (vai exportar o dado e estrutura)
3. export to self-contained file
4. include create schema

Irá guardar na pasta tudo o que usei a estrutura e os dados podendo levar de um servidor a outro.

Gera um dump completo e importa para outro

# **💫 SELECT**

O \* todas as colunas (campos)  
Seleciona todas as colunas da tabela cursos

select \*from nome\_da\_tabela;

select nome\_da\_coluna, nome\_da\_coluna from nome\_da\_tabela;

### **👾 Ordenar de acordo com uma coluna**

select \*from nome\_da\_tabela

order by nome\_da\_coluna;

### **👾 Ordenar descendente (de baixo para cima)**

select \*from nome\_da\_tabela

order by nome\_da\_coluna desc;

### **👾 Ordenar ascendente (de cima para baixo)**

Pode adicionar asc ou não colocar nada

select \*from nome\_da\_tabela

order by nome\_da\_coluna asc;

### **👾 Filtrar as colunas ordenando colunas**

select nome\_da\_coluna from nome\_da\_tabela

order by nome\_da\_coluna;

select nome, cargo, ano from cursos

order by ano, nome;

select nome, carga, ano from cursos;

### **👾 Filtrar Linhas**

Posso substituir \* pelo nome da coluna

select \*from nome\_da\_tabela

where nome\_da\_coluna = dado\_da\_linha

order by nome\_da\_coluna;

select \*from cursos

where ano = 2016

order by nome;

### **👾 Usando operadores**

=

<=

>=

<> ou !=

selet \*from nome\_da\_tabela

where nome\_da\_coluna <= 2015

order by nome\_da\_coluna;

select nome, descricao from cursos

where ano <= 2015

order by nome;

### **👾 Entre um e outro (BETWEEN, AND)**

select nome\_da\_coluna from nome\_da\_tabela

where nome\_da\_coluna between 2014 and 2016;

select nome, ano from cursos

where ano between 2014 and 2015

order by ano desc, nome;

### **👾 Where com mais dados (IN = em)**

select nome, ano, descricao from cursos

where ano in (2014,2016)

order by ano;

### **👾 Combinando testes (AND, OR)**

select nome, descricao, carga, totaulas from cursos

where carga < 35 and totaulas < 30

order by carga;

select nome, descricao, carga, totaulas from cursos

where carga < 35 or totaulas < 30

order by carga;

### **👾 Puxar um dado**

select nome\_da\_coluna from nome\_da\_tabela

where nome\_da\_coluna ='nome\_do\_dado';

select nome, profissao, sexo from gafanhotos

where profissao = 'programador';

select nome, profissao, sexo from gafanhotos

where profissao in ('programador');

### **👾 Puxar dois ou mais dados**

select nome, profissao, sexo from gafanhotos

where profissao in ('Programador') and sexo = 'M';

select nome, profissao, sexo from gafanhotos

where profissao = 'Programador' and sexo = 'M';

### **👾 Filtro de texto**

like 'a%' - com essa letra no começo da palavra  
like '%a%' - com essa letra em qqlr lugar  
like '%a' - com essa letra no final da palavra  
not like 'a%'  
not like '%a%'  
not like '%a'

select nome, nascimento, nacionalidade, sexo from gafanhotos

where nome like 'J%' and sexo = 'F' and nacionalidade = 'Brasil';

### **👾 Puxar dados sem repetir as informações**

select distinct nome\_da\_coluna from nome\_da\_tabela;

select distinct nacionalidade from gafanhotos;

select distinct carga from cursos

order by carga;

### **👾 Função de agregação - Selecionar ou totalizar informações**

count ( \* )

Irá contar a quantidade da informação

No ligar de \* pode adicionar o nome da coluna desejada

select count(\*)from cursos;

select count(\*)from cursos

where carga >40;

### **👾 Agrupar as informações**

group by

select nome\_da\_coluna count(\*) from nome\_da\_tabela

group by nome\_da\_coluna;

select carga, count(nome) from cursos

group by carga;

select totaulas, count(\*) from cursos

group by totaulas;

select ano, count(\*) from cursos

group by ano;

### **👾 Agrupar informação especifica**

having

select ano, count(\*) from cursos

group by ano

having count(ano) >= 5

order by count(\*) desc;

### **👾 Maior entre os dados**

max()

select max(carga)from cursos;

### **👾 Menor entre os dados**

min()

select nome, min(totaulas) from cursos

where ano = '2016';

### **👾 Somar**

sum()

select sum(totaulas) from cursos where ano = '2016';

### **👾 Media** avg ()