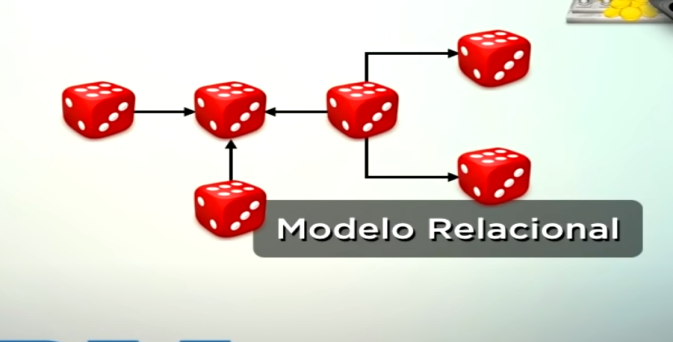


Aula 1 – Banco de dados

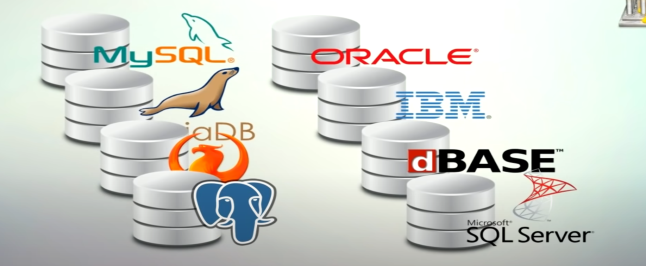
O que é um Banco de dados?











Aula 2

Instalando o MySQL com WAMP



**DICA - ACID**

**D**urabilidade

**I**solamento

**C**onsistência

**A**tomicidade

**Princípios ACID**

1. **Atomicidade (Atomicity)**

**Definição**: Este princípio garante que todas as operações em uma transação são completadas com sucesso ou, em caso de falha, nenhuma das operações é realizada.

**Exemplo**: Em uma transferência bancária, a quantia deve ser debitada de uma conta e creditada em outra. Se uma das operações falhar, nenhuma das duas deve ser realizada.

1. **Consistência (Consistency)**

**Definição**: Assegura que uma transação leva o banco de dados de um estado consistente a outro estado consistente. Qualquer dado escrito no banco de dados deve ser válido segundo todas as regras definidas, incluindo constraints, cascades, triggers, etc.

**Exemplo**: Se uma regra diz que um campo de saldo não pode ser negativo, uma transação que resultaria em um saldo negativo deve ser revertida.

1. **Isolamento (Isolation)**

**Definição**: Garante que transações concorrentes sejam executadas como se fossem serializadas, ou seja, uma transação não deve interferir em outra transação em execução simultânea.

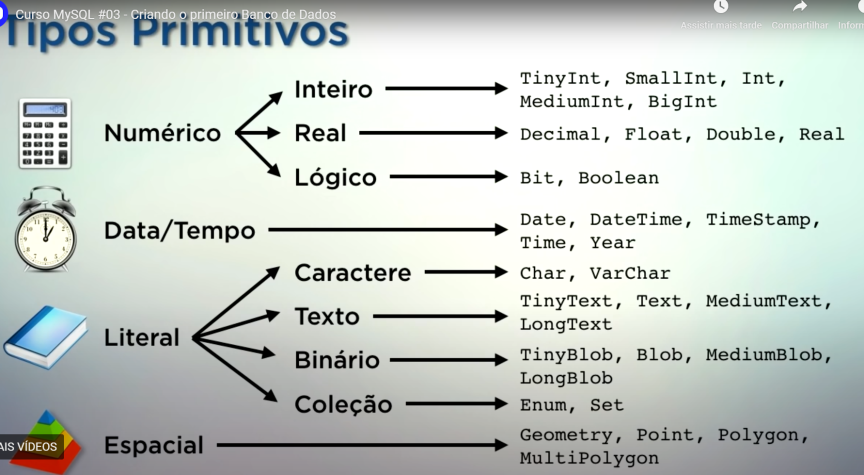
**Exemplo**: Se duas transações tentam atualizar o mesmo registro simultaneamente, o isolamento garante que uma seja executada antes da outra, evitando inconsistências.

1. **Durabilidade (Durability)**

**Definição**: Assegura que uma vez que uma transação tenha sido confirmada, suas alterações são permanentes no banco de dados, mesmo em caso de falhas no sistema.

**Exemplo**: Após um commit de uma transação de venda, os dados de venda são garantidos de serem persistidos, mesmo que haja uma queda de energia imediatamente após a confirmação.

Aula 03 – Criando Banco de Dados



create database cadastro;

Criando Tabelas

create table pessoas(

nome varchar (30),

idade tinyint (3),

sexo char(1),

peso float,

altura float,

nacionalidade varchar(20)

);

describe pessoas;

Criando banco de dados com caracteres utf8

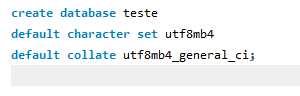
create database cadastro

default character set utf8

default collate utf8\_general\_ci;

Melhorando a Estrutura do Banco de Dados

Aula 4

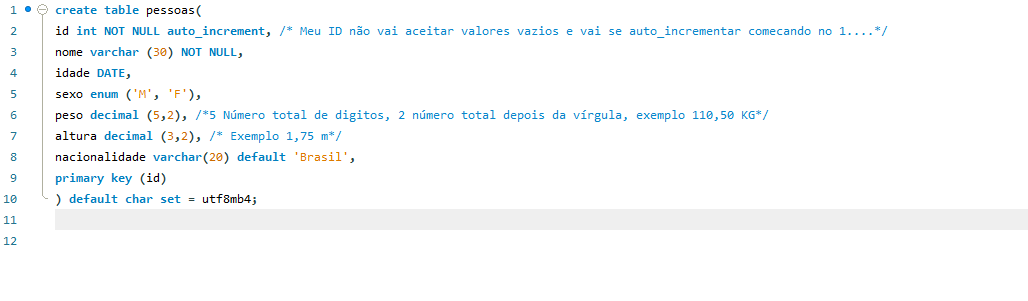


Apagar um banco de dados - drop database cadastro;

****

**Constraints** são restrições criadas no banco de dados para: Impedir que dados inválidos sejam cadastrados no banco de dados. Garantir a qualidade dos dados. Garantir que campos obrigatórios sejam preenchidos.

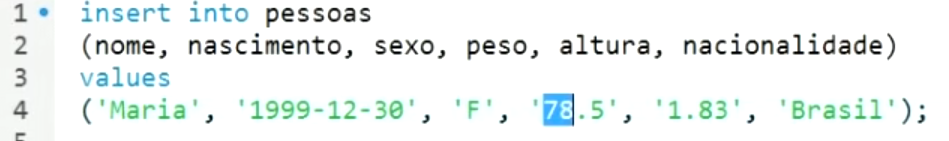
**Criando tabela com constraints**

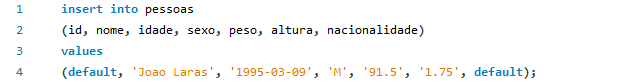


Inserindo Dados na Tabela

Aula 5

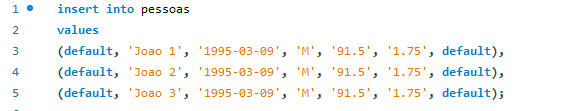
**Insert into**

****

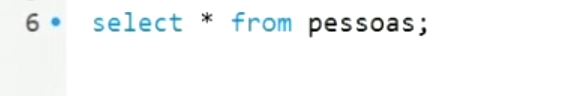
****

****

Adicionando varias pessoas ao mesmo tempo.

****

**Verificar todos os dados da tabela pessoas**

****

DDL, ou Data Definition Language (Linguagem de Definição de Dados), é um conjunto de comandos SQL usados para definir e gerenciar a estrutura de bancos de dados e objetos de banco de dados, como tabelas, índices, e esquemas. Esses comandos permitem a criação, alteração e exclusão de objetos no banco de dados.

Os comandos mais comuns do DDL são:

**CREATE**: Utilizado para criar novos objetos no banco de dados, como tabelas, índices, ou esquemas.

**ALTER**: Usado para modificar a estrutura de objetos existentes no banco de dados.

**DROP**: Utilizado para remover objetos do banco de dados.

**TRUNCATE**: Usado para remover todos os registros de uma tabela sem remover a estrutura da tabela.

DML, ou Data Manipulation Language (Linguagem de Manipulação de Dados), é um conjunto de comandos SQL usados para inserir, atualizar, excluir e recuperar dados armazenados em um banco de dados. Esses comandos são usados para a manipulação direta dos dados nas tabelas.

Os comandos mais comuns do DML são:

**INSERT**: Utilizado para inserir novos registros em uma tabela.

Alterando a Estrutura da Tabela

Aula 5

Adicionando uma nova coluna profissão na tabela pessoas

Sem especificar onde eu quero adicionar, ele sempre vai adicionar a coluna por ultimo.



Usando o comando after (depois), ele vai adicionar a coluna depois do nome (onde eu especifico).



Usando o comando first (primeiro), ele vai adicionar a coluna em primeiro.

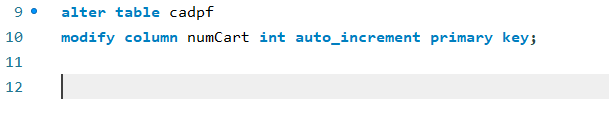


Deletando uma coluna na tabela pessoas -DROP



Modificando os dados de uma coluna - MODIFY





Alterando o nome de uma coluna – CHANGE

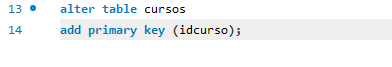


Renomeando Tabela

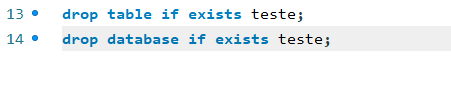


Adicionando chave primaria na tabela





Apagando uma database ou uma tabela



Manipulando linhas (UPDATE, DELETE E TRUNCATE).

Aula 7

Update



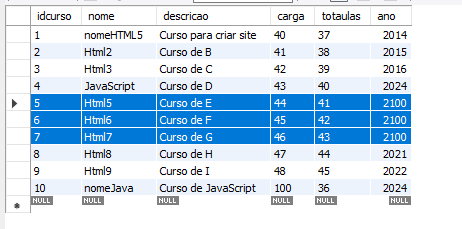




Atualize os cursos, configurando o nome para **nomeHTML5**, onde id curso = ‘1’;

Modificando varias linhas ao mesmo tempo

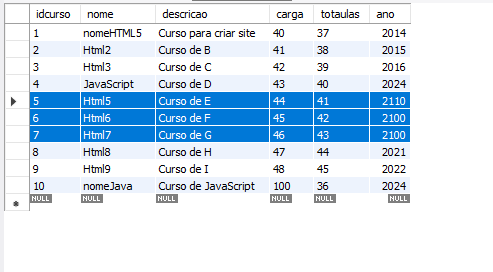
Edit > preferences > SQL editor > (x) safe update.





Usando o limit.





Apagando linhas





Removendo todas as linhas de uma tabela.



Gerenciando Cópias de Segurança MYSQL

Aula 8

Exportando uma base de dados

Server > DataExport> Start export

Importando uma base de dados

Server > DataImport> Start Import

PHPMyAdmin

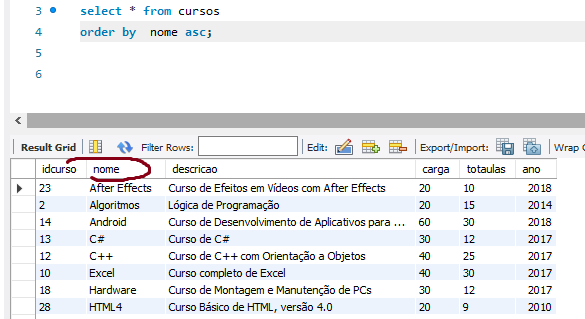
Aula 9

SELECT PARTE 1

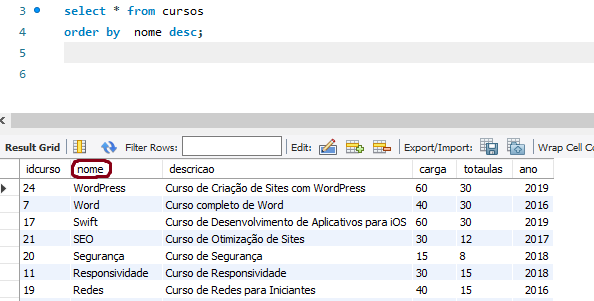
Aula 11

Comando select \* from

Asc = ascendente.



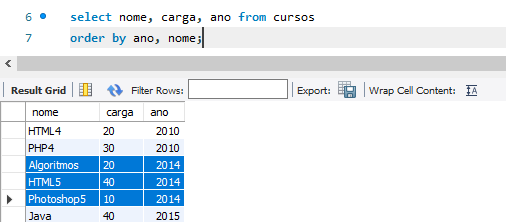
Desc = descendente



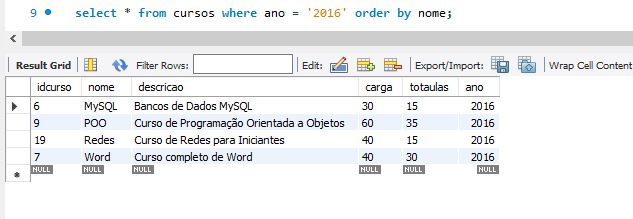
Filtrando a busca na base de dados



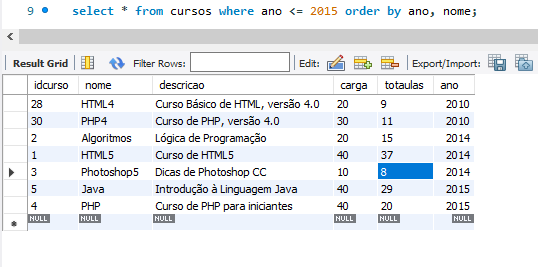
Buscando por nome e carga e ordenando por ano e nome.



Selecionando as linhas dos curso na qual o ano é 2016, e colocando em ordem pelo nome.



Usando operadores relacionais para filtrar as linhas por ano, e colocando em ordem por ano e nome. ( <, <=, >, >=, =, !=).



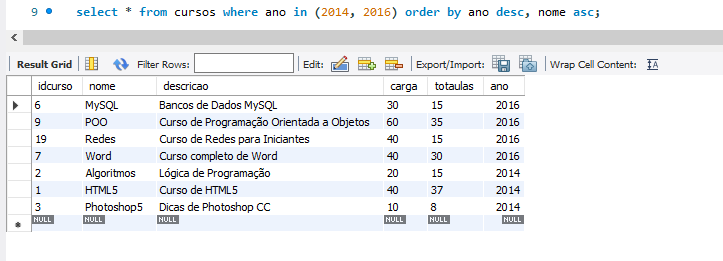
Usando between (ENTRE) entre 2014 e 2016.

Selecionando \* todos de cursos onde ano entre 2014 e 2016.



Usando IN (EM) em 2014 e 2016.

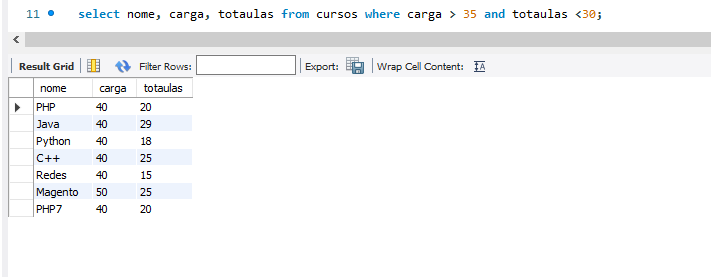
Selecionando \* todos de cursos onde ano em 2014 e 2016.



Filtrando usando dois operadores lógicos.

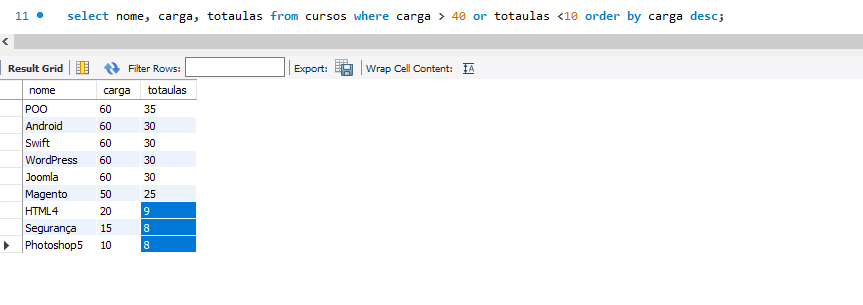
USANDO O AND (e).

Selecionando nome, cargas e totaulas de cursos onde carga maior que 35 e totaulas menor que 30.



USANDO OR (ou).

Selecionando nome, cargas e totaulas de cursos onde carga maior que 40 ou totaulas menor que 10, e ordenar por carga decrescente .



SELECT PARTE 2

Aula 12

Usando operador like.

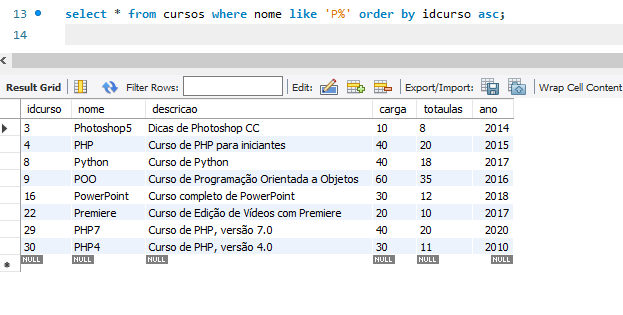
Selecionando \* todos de cursos onde nome é semelhante a p, P, ordenando em ordem por idcurso acrescente .

% = QUALQUER COISA.

P%

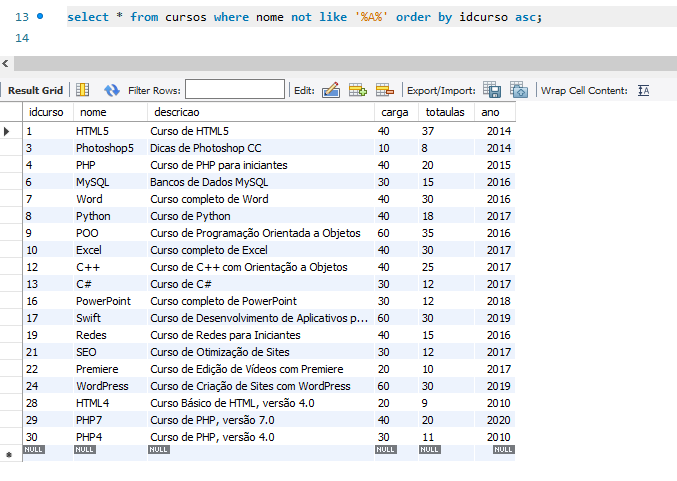
%P

%P%



Usando operador not like.

Selecionando \* todos de cursos onde nome não é semelhante a A, a, ordenando em ordem por idcurso acrescente .

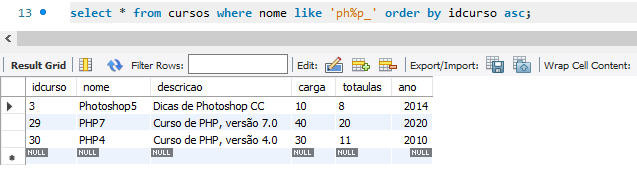


Selecionando \* todos de cursos onde nome é semelhante a ph%p--, ordenando em ordem por idcurso acrescente .

\_ OBRIGADA QUE TENHA QUALQUER LETRA

Photoshop

Começa com ph, % tem qualquer caractere e termina com p, e tem um caractere no final\_ que é o 5.

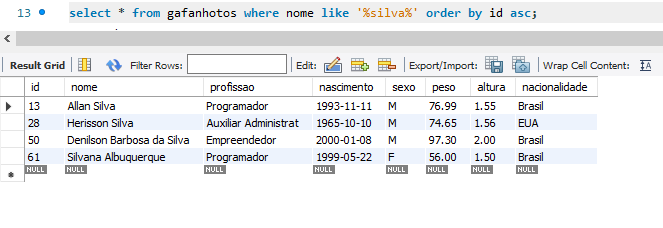


Selecionando \* todos de gafanhotos onde nome é semelhante a silva, ordenando em ordem por id acrescente .

% qualquer coisa, Silva = %silva

Silva, % qualquer coisa silva = silva%

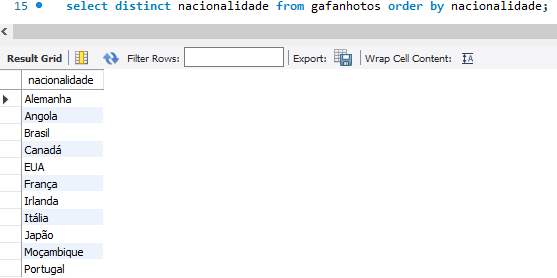
% qualquer coisa, Silva, qualquer coisa %



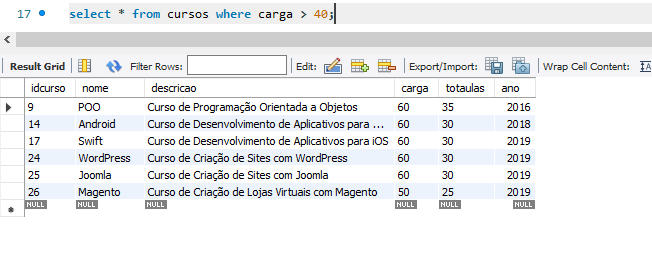
Usando operador distinct.

Distinct = oque não é igual.

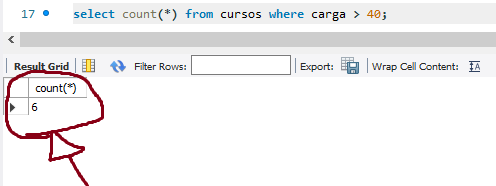
Selecione oque não é igual em nacionalidade de gafanhotos ordenado em nacionalidade;



Função de agregação count (\*).



Usando função count.

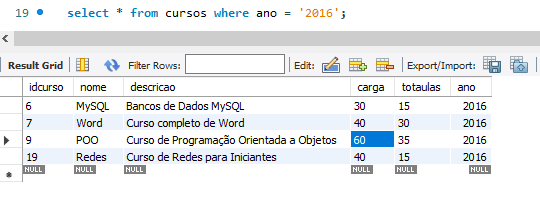


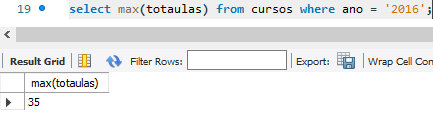


Usando a função max.

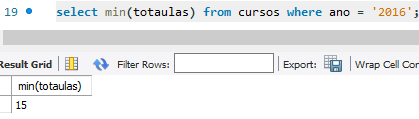
Selecione a maior carga em de cursos.



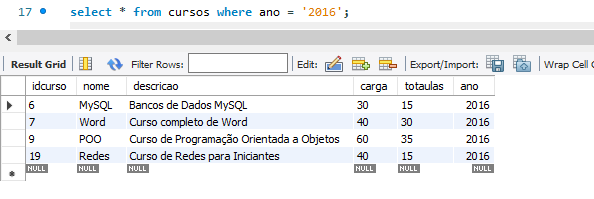


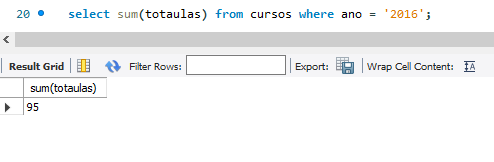


Usando a função min.



Usando função sum. SOMA

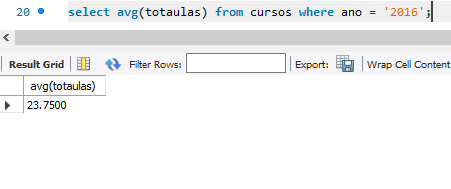




Usando função avg. MÉDIA

Totaulas = 95;

Cursos em 2016 = 4;



Exercicios.

1) Uma lista com o nome de todos os gafanhotos Mulheres.

R: select nome from gafanhotos where sexo = 'F';

2) Uma lista com os dados de todos aqueles que nasceram entre 1/Jan/2000 e 31/Dez/2015.

select \* from gafanhotos where nascimento between '2000-01-01' and '2015-12-31' order by nascimento asc;

3) Uma lista com o nome de todos os homens que trabalham como programadores.

select nome from gafanhotos where sexo = 'M' and profissao = 'Programador' ;

4) Uma lista com os dados de todas as mulheres que nasceram no Brasil e que têm seu nome iniciando com a letra J.

select \* from gafanhotos where sexo = 'F' and nacionalidade = 'Brasil' and nome like 'j%';

5) Uma lista com o nome e nacionalidade de todos os homens que têm Silva no nome, não nasceram no Brasil e pesam menos de 100 Kg.

select nome, nacionalidade from gafanhotos where sexo = 'M' and nome like'%silva%' and nacionalidade != 'Brasil' and peso <100;

6) Qual é a maior altura entre gafanhotos Homens que moram no Brasil?

select max(altura) from gafanhotos where sexo ='M' and nacionalidade = 'Brasil';

7) Qual é a média de peso dos gafanhotos cadastrados?

select avg(peso) from gafanhotos;

73.967705

8) Qual é o menor peso entre os gafanhotos Mulheres que nasceram fora do Brasil e entre 01/Jan/1990 e 31/Dez/2000?

select min(peso) from gafanhotos where sexo = 'F' and nacionalidade != 'Brasil' and nascimento between '1990-01-01' and '2000-12-31';

35,90

9) Quantas gafanhotos Mulheres tem mais de 1.90cm de altura?

select count(\*) from gafanhotos where sexo = 'F' and altura > 1.90;

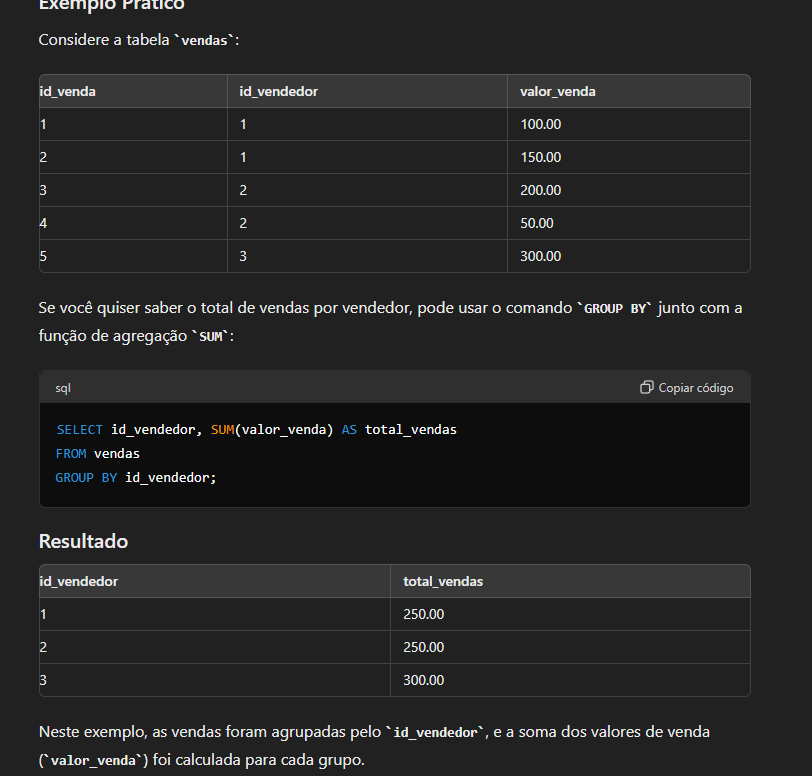
5

SELECT PARTE 2

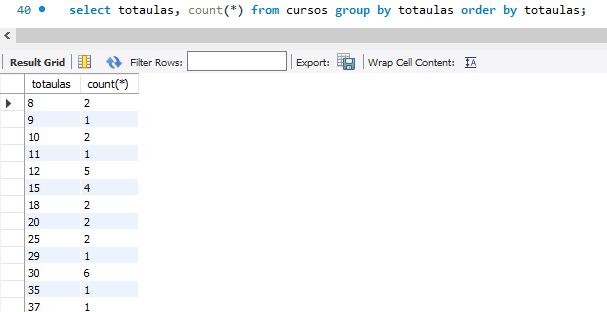
Aula 13

Usando o agrupamento

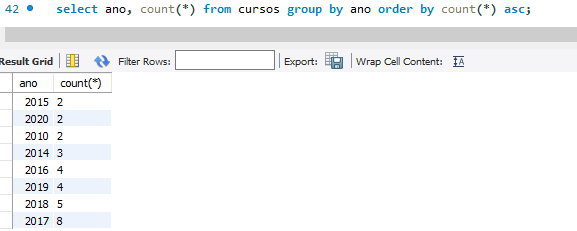
O comando GROUP BY em MySQL é utilizado para agrupar linhas que possuem valores idênticos em colunas especificadas em conjuntos de resumos, como sumarização de dados. Ele é frequentemente usado em conjunto com funções de agregação (SUM, COUNT, AVG, MAX, MIN) para realizar operações em grupos de dados.



Selecione totaulas, contando \* todos de cursos agrupado por totaulas e ordenado por totaulas;

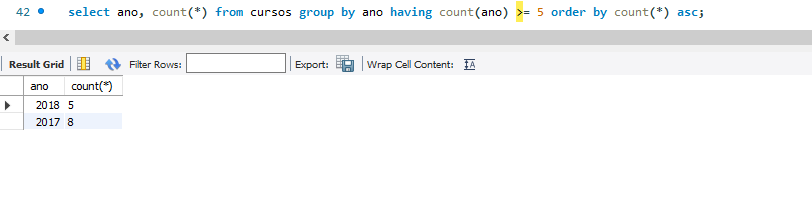


Selecione ano, contando \* todos de cursos agrupado por ano e ordenado por count\*todos ascrecente;

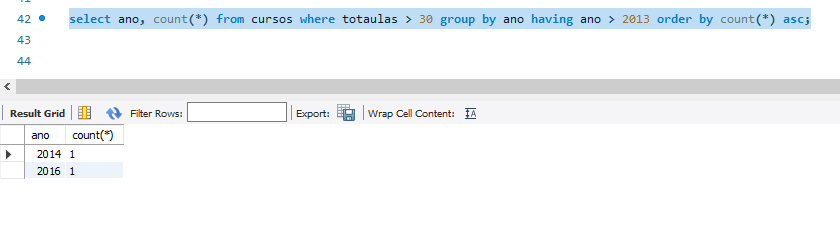


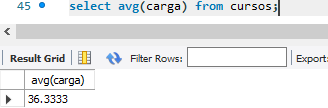
Usando HAVING = where dentro do select

Selecione ano, contando \* todos de cursos agrupado por ano tendo count(ano) > = 5 e ordenado por count\*todos ascrecente;

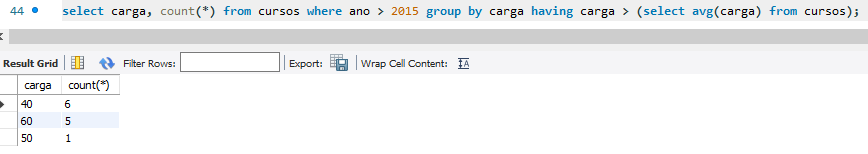


Selecione ano, contando \* todos de cursos onde totaulas > 30 agrupado por ano tendo ano > 2013 e ordenado por count\*todos ascrecente;





Selecione carga, contando \* todos de cursos onde ano > 2015 agrupado por carga tendo carga> (media de carga);



Exercicios.

Exercicios.

1) Uma lista com as profissões dos gafanhotos e seus respectivos quantitativos.

select profissao, count(\*) from gafanhotos group by profissao order by profissao asc;

2) Quantos gafanhotos Homens (M) e quantas gafanhotos Mulheres (F) nasceram após 01/01/2005?

select sexo, count(\*) from gafanhotos where nascimento > 2005-01-01 group by sexo;

3) Uma lista com os gafanhotos que nasceram fora do Brasil, mostrando o páis de origem e total de pessoas nascidas lá. Só nos interessam os países que tiverem mais de 3 gafanhotos com essa nacionalidade.

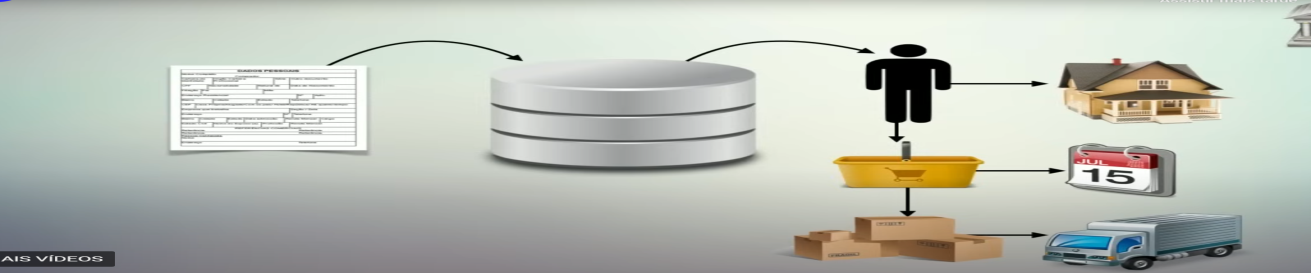
select nacionalidade, count(\*) from gafanhotos where nacionalidade != 'Brasil' group by nacionalidade having count(nacionalidade) > 3;

4) Uma lista agrupada pela altura dos gafanhotos, mostrando quantas pessoas pesam mais de 100kg e que estão acima da média de altura de todos os cadastrados.

select nome, peso, altura, count(\*) from gafanhotos where peso > 100 group by nome, peso, altura having altura > (select avg(altura)from gafanhotos);

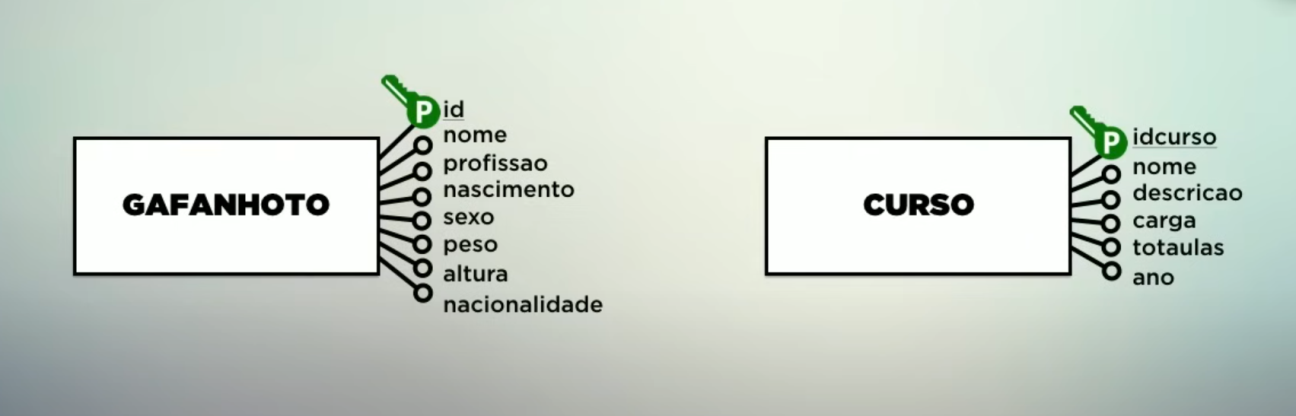
MODELO RELACIONAL

Aula 14

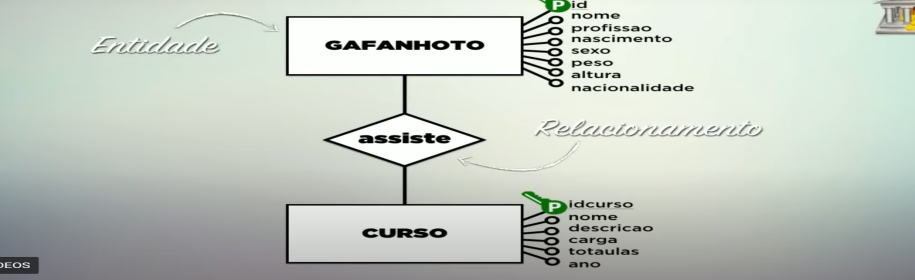




Chave primária



DER – DIAGRAMA ENTIDADE RELACIONAMENTO



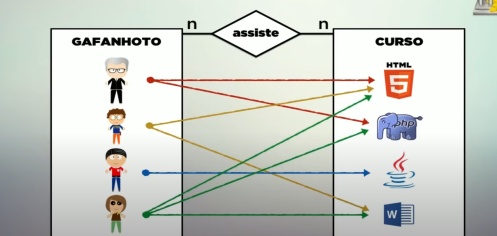
CARDINALIDADE

Muitos para muitos

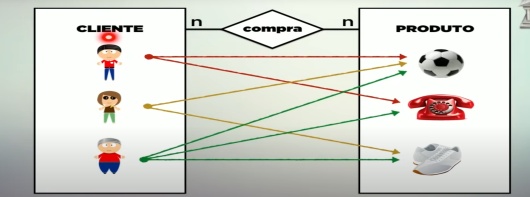
Exemplo:

Um gafanhoto pode assistir vários cursos.

Um curso pode ser assistido por vários gafanhotos.





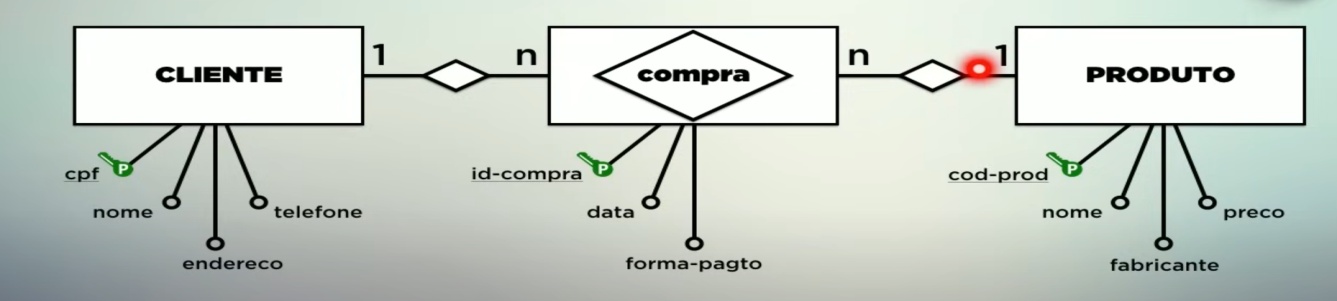


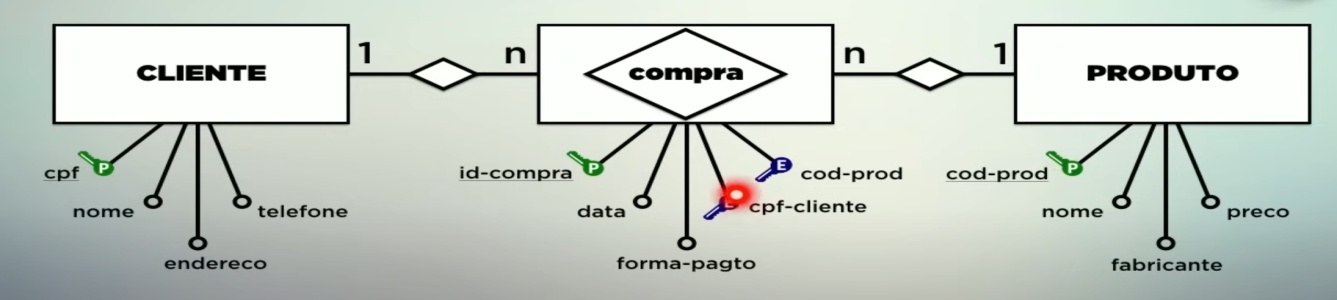
Relacionamento compra vira uma entidade.

Compra também tem seus atributos.

Criei dois mini relacionamentos.

Regra: Pegue a chave primária do lado um (CLIENTE e PRODUTO), e jogue para o lado muitos, como chave estrangeira.



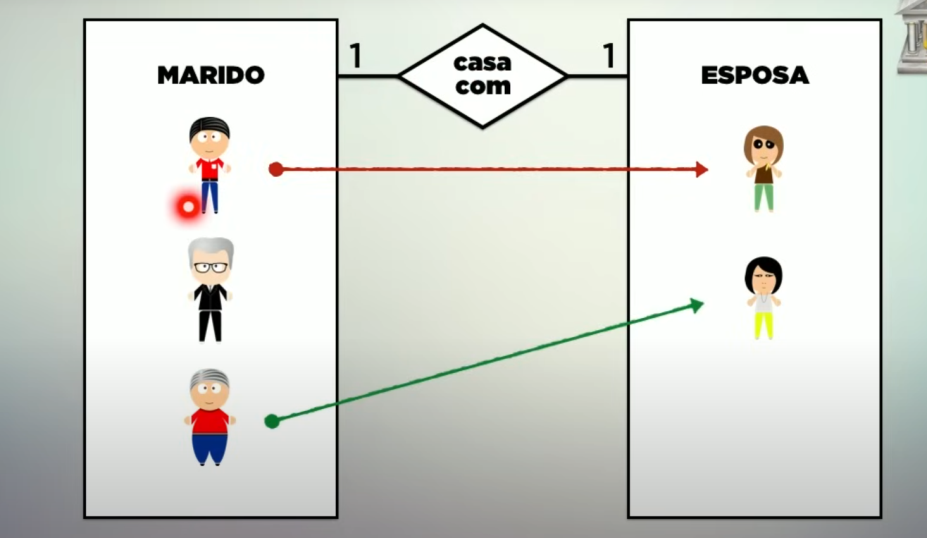


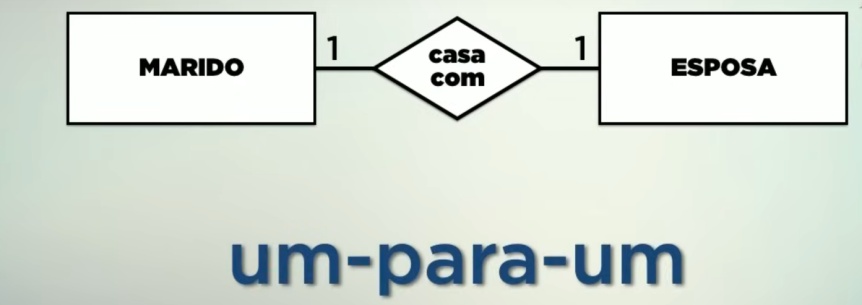
Um para um

Exemplo:

Um marido pode ter apenas uma esposa.

Uma esposa pode ter apenas um marido.

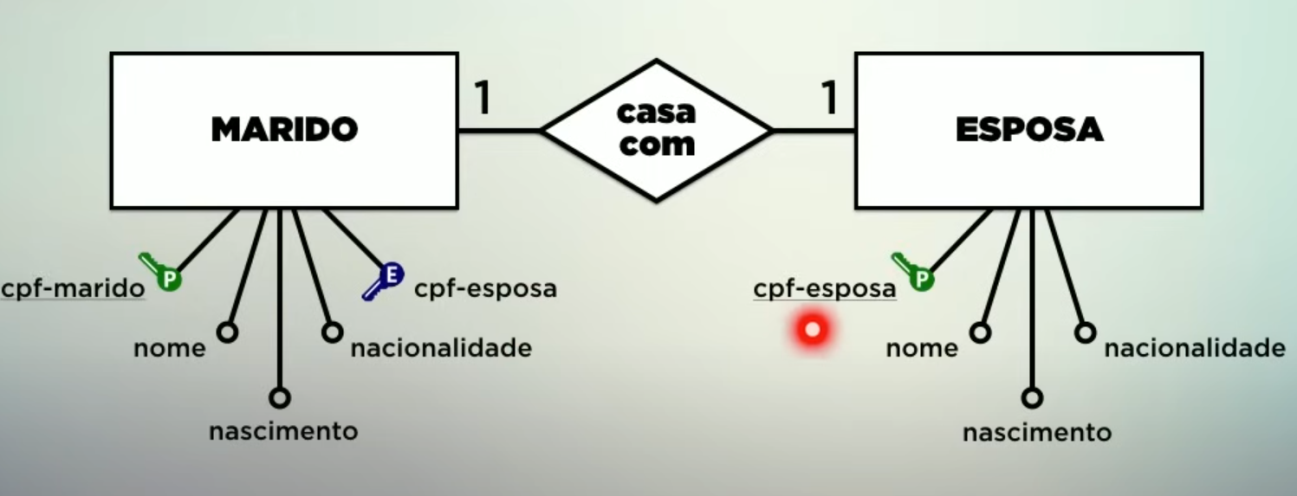




Escolha uma entidade dominante.

EXEMPLO: ENTIDADE DOMINANTE MARIDO.

Regra: Pega a chave da outra entidade não dominante (ESPOSA) e transfira para o lado dominante (MARIDO), colocando-a como chave estrangeira.

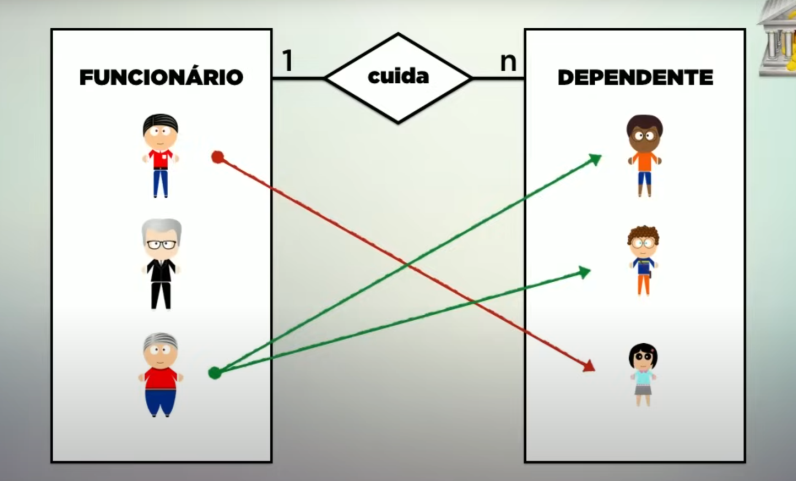


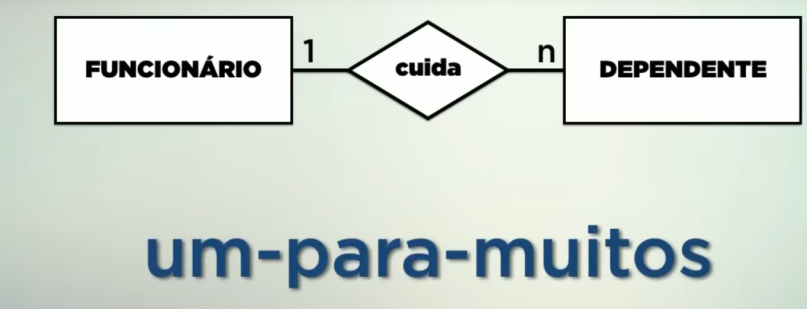
Um para muitos

Exemplo:

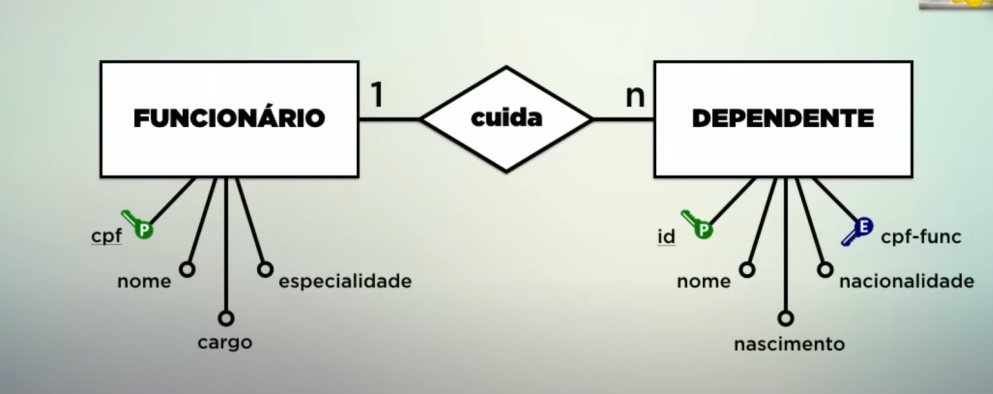
Um funcionário pode cuidar de vários dependentes.

Vários dependentes pode ser cuidado por apenas um funcionário.





Regra: Pegue a chave primária do lado um e jogue para o lado muitos, como chave estrangeira.



CHAVE ESTRAGEIRA

É uma chave primaria de algum lugar que veio para outro lugar.

