Anotações do Banco de Dados feito nas aulas de MySQL (CURSO EM VIDEO)

create database cadastro

default character set utf8mb4

default collate utf8mb4\_general\_ci;

create table pessoas(

id int NOT NULL AUTO\_INCREMENT,

nome varChar(30) not null,

nascimento date,

sexo enum('M', 'F'),

peso decimal(5,2),

altura decimal(3,2),

nacionalidade varChar(20) default 'Brasil',

primary key(id)

)default charset = utf8mb4;

INSERT INTO pessoas

(id,nome,nascimento,sexo,peso,altura,nacionalidade)

VALUES

(DEFAULT, 'Ana','1999-08-04','F','52.5','1.56','Inglaterra'),

(DEFAULT, 'Maria','1987-05-11','F','54.5','1.65','Espanha'),

(DEFAULT, 'Pedro','1998-03-21','M','77.5','1.98','Rússia');

/\* alter table pessoas

drop column profissao; \*/

alter table pessoas

add column profissao varChar(10) after nome;

alter table pessoas

modify column profissao varChar(20) not null default '';

alter table pessoas

change column profissao prof varChar(20) not null default '';

alter table pessoas rename to gafanhotos;

select \* from gafanhotos;

create table if not exists cursos(

nome varChar(30) not null unique,

descricao text,

carga int unsigned,

totaulas int unsigned,

ano year default '2015'

)default charset=utf8mb4;

alter table cursos

add column idcurso int first;

alter table cursos

add primary key(idcurso);

insert into cursos values

('1', 'HTML4', 'Curso de HTML5', '40', '37', '2014'),

('2', 'Algoritmos', 'Lógica de Progamação', '20', '15', '2014'),

('3', 'Photoshop', 'Dicas de Photoshop CC', '10', '8', '2014'),

('4', 'PGP', 'Curso de PHP para iniciantes', '40', '20', '2010'),

('5', 'Jarva', 'Introdução à Linguagem Java', '40', '29', '2010'),

('6', 'MySQL', 'Banco de Dados MySQL', '30', '15', '2016'),

('7', 'Word', 'Curso completo de Word', '40', '30', '2016'),

('8', 'Sapateado', 'Danças Rítmicas', '40', '37', '2018'),

('9', 'Cozinha Árabe', 'Aprender a fazer Kibe', '40', '30', '2018'),

('10', 'YouTuber', 'Gerar polêmica e ganhar inscritos', '5', '2', '2018');

update cursos

set nome = 'HTML5'

where idcurso = '1'

limit 1;

update cursos

set nome = 'PHP', ano = '2015'

where idcurso = '4'

limit 1;

delete from cursos

where idcurso = '8'

limit 1;

delete from cursos

where ano = '2018'

limit 2;

/\* truncate table cursos; APAGA TODOS OS REGISTROS DA TABELA! \*/

select \* from cursos;

select \* from gafanhotos;

/\* TODAS AS FUNÇÕES DO SELECT ABAIXO \*/

select \* from cursos order by nome; /\* Alinhar na ordem alfabetica de acordo com a coluna nome \*/

select \* from cursos order by nome desc; /\* Alinhar na ordem alfabetica de acordo com a coluna nome de baixo pra cima(ordem inversa) \*/

select nome,carga,ano from cursos order by nome; /\* Mostra todos os registros de tabelas especificas \*/

select \* from cursos where ano = '2016' order by nome; /\* Mostrar todos os cursos de 2016 \*/

select \* from cursos where ano between '2014' and '2016' order by ano, nome desc; /\* Mostrar todos os cursos entre 2014(between '2014') e 2016. Ordenando eles por ano e nome decrescente \*/

select \* from cursos where ano in ('2016','2017','2018') order by nome; /\* Mostrar todos os cursos de 2016, 2017 e 2018 \*/

select nome, carga, totaulas from cursos where carga > 35 and totaulas < 30; /\* Mostrar todos os registros nas tabelas nome, carga e totaulas que tenham carga maior que 35 e o total de aulas menor que 30 \*/

select \* from cursos where nome like 'P%'; /\* Mostrar todos os cursos que começam com a letra P. O "like" é usado como "parecido", exemplo: nome é semelhante(like) a 'P%'. O % na frente do P, significa que ele vai selecionar qualquer ou nenhuma letra depois desse P, é como se tivesse completando o que vai ser escrito com o registro da tabela \*/

select \* from cursos where nome like '%P%'; /\* Mostrar todos os registros dos cursos que tenham o P em qualquer parte do seu nome \*/

select \* from cursos where nome not like '%P%'; /\* Mostrar todos os registros dos cursos que não tenham o P em qualquer parte do seu nome \*/

select distinct nacionalidade from cursos order by nacionalidade; /\* O distinct mostra quais nacionalidades existem na tabela apenas uma vez, se tiver alguma nacionalidade que apareça mais de uma vez na tabela, o distinct mostra ela só uma vez \*/

select count(\*) from cursos; /\* Mostrar a quantidade de cursos que existem na tabela \*/

select count(\*) from cursos where carga > 40; /\* Mostrar a quantidade de cursos que tenham a carga maior que 40 que existem na tabela \*/

select max(carga) from cursos; /\* Mostrar a carga maxima que existe entre os cursos na tabela \*/

select nome, max(totaulas) from cursos where ano = '2016'; /\* Mostrar o curso que teve mais aula no ano de 2016 \*/

select nome, min(totaulas) from cursos; /\* Mostrar o curso que teve menos aulas \*/

select nome, min(totaulas) from cursos where ano = '2016'; /\* Mostrar o curso que teve menos aulas em 2016 \*/

select nome, sum(totaulas) from cursos where ano = '2016'; /\* Mostrar o somatório das aulas dos anos de 2016 \*/

select nome, avg(totaulas) from cursos where ano = '2016'; /\* Mostrar a média das aulas dos cursos em 2016 \*/

select carga from cursos group by carga order by carga; /\* Diferente do distinct, o group by faz com que agrupem todas as cargas parecidas em um bloco só, mas você consegue ver a quantidade em cada bloco usando o count(\*), no distinct não dá pra fazer isso \*/

select carga, count(\*) from cursos group by carga order by count(\*) desc; /\* Nesse exemplo, como citado no trecho acima, eu conseguirei ver quantos grupos tem a mesma carga. Usando o count(\*) e o group by \*/

select carga, count(\*) from cursos where totaulas = 30 group by carga; /\* Usando o group by com o where \*/

select carga,count(\*) from cursos where ano > 2015 group by carga having carga > (select avg(carga) from cursos); /\* Mostrando a quantidade de cursos que tenham o ano maior que 2015 agrupados pela carga e tendo essa carga maior que a média de todas as cargas.\*/

/\* EXERCICIOS AULA 03 - SELECT

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

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

03) 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.

04) 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.

\*/

/\* RESPOSTAS NA ORDEM \*/

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

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

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

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

/\* EXERCICIOS AULA 02 - SELECT

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

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

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

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

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

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

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

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

\*/

/\* RESPOSTAS NA ORDEM \*/

select nome from gafanhotos where sexo = 'F';

select nome, nascimento from gafanhotos where (nascimento between '2000-01-01' and '2015-12-31') order by nascimento;

select nome, profissao, sexo from gafanhotos where (profissao = ('Programador' or 'programador')) and (sexo = 'M');

select nome, nacionalidade from gafanhotos where (nacionalidade = 'Brasil') and (nome like 'J%') and (sexo = 'F');

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

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

select avg(peso) from gafanhotos;

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

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

/\* CHAVES ESTRANGEIRAS AULA 15 - CURSO EM VIDEO \*/

use cadastro;

describe gafanhotos;

alter table gafanhotos add cursopreferido int;

alter table gafanhotos add foreign key(cursopreferido) references cursos(idcurso);

select \* from gafanhotos;

select \* from cursos;

update gafanhotos set cursopreferido = '6' where id = '1'; /\* Selecionando o cursopreferido de maneira manual \*/

/\* A maneira "automática" seria modificar o cursopreferido no workbench após usar a opção select \* from gafanhotos e depois clicar no aply, o workbench vai criar o código pra você e facilitar sua vida \*/

select gafanhotos.nome, gafanhotos.cursopreferido, cursos.nome, cursos.ano from gafanhotos join cursos

on gafanhotos.cursopreferido = cursos.idcurso; /\* Juntando as duas tabelas com o join e igualando o número do cursopreferido(tabela gafanhotos) ao idcurso(tabela cursos) \*/

select gafanhotos.nome, gafanhotos.cursopreferido, cursos.nome, cursos.ano from gafanhotos /\* <- LEFT PREFERENCE \*/left outer join cursos

on cursos.idcurso = gafanhotos.cursopreferido; /\* Na hora de mostrar, dê preferencia para a tabela gafanhotos(left outer join) \*/

select gafanhotos.nome, gafanhotos.cursopreferido, cursos.nome, cursos.ano from gafanhotos right outer join /\* RIGHT PREFERENCE -> \*/ cursos

on cursos.idcurso = gafanhotos.cursopreferido; /\* Na hora de mostrar, dê preferencia para a tabela cursos(right outer join) \*/