acessar todas as tabela

> show databases;

cria tabela aluno;

> create database cadastro

default character set utf8

default collate utf8\_general\_ci;

acessar aluno

> use empresa

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

criar table pessoas

> 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)

) ENGINE = InnoDB DEFAULT CHARACTER SET = utf8;﻿

acresentar um coluna

> alter table pessoas

add column profissao varchar(10) after nome;

Para coloca na primeira linha "first"

deletar coluna

> drop column profissao;

add column sem nome not null pareca

alter column profissao;

add column profissao varchar(30) not null default '' alter nome;

midificar coluna

> alter table pessoas

modify column profissao varchar(30);

renomear a coluna

>alter table pessoas

change column profissao prof varchar(20);

alterar o nome da tabela

> alter table pessoas

rename to gafanhotos;

> create table if not exists cursos( -> se a tabela existir o BD

não deixa criar

nome varchar(30) not null unique, -> nao deixa sobre escrever

descrição text,

carga int unsigned -> nao deixa colcar sinal

totaulas int,

ano year default '2018'

) default charset = utf8;

adiciona chave primeria

> alter table cursos

add primary key (idcurso);

adicionar um coluna

> alter table gafanhotos

add column cursopreferido int;

troca o nome do valor da tabela

> update cursos

set nome = 'Java', carga = '40', ano = '2015'

where idcurso = '5'

limit 1;

apagar linha

> delete from cursos

where idcurso='8';

apagar todas as linha da tabela

> truncate cursos

acessar tabela criada do aluno

> show columns in aluno;

acessar tabela criada

> describe funcionarios;

coloca os dados na tabela criada

> insert into funcionarios (nome, idade) values ('usuarios1', '30');

olha os valores

> select \* from funcionarios;

acessar tabela pela idade > 40

> select \* from funcionarios where idade > 40;

deleta usuarios pelo id

> delete from funcionarios where id '1';

deleta tabele inteira

> drop table funcionarios;

deleta database inteira

> drop database empresa;

selecionar curso entre 2014 e 2016

> select nome, ano from cursos

where ano between 2014 and 2016;

selecionar ano <= 2015 por ordem de nome

> select nome, descricao from cursos

where ano <= 2015

order by nome;

selecionar os curso de 2014 e 2016 apenas

> select nome, descricao, ano from cursos

where ano in (2014, 2016)

order by ano;

operador

> select nome, carga, totaulas from cursos

where carga > 36 and totaulas < 30;

seleciona no campo nome todo letra que começa com P

> select \* from cursos

where nome like 'P%';

seleciona no campo nome todo letra que tem A na palavra

> select \* from cursos

where nome like '%a%';

Não tem letra A no nome

> select \* from cursos

where nome not like '%a%';

selecionar carga do curso não deixado repetir as horas

por ordem de carga

> select distinct carga from cursos

order by carga;

conta quantidade de cursos;

> select count(\*) from cursos;

conta quantidade de cursos que tem carga maior que 40

> select count(\*) from cursos

where carga > 40;

selecionar maior carga entre os cursos

> select max(carga) from cursos;

vai somar todas os cursos feito no ano de 2016

> select sum(totaulas) from cursos

where ano = '2016';

media

select avg(totaulas) from cursos

where ano = '2016';

vai conta quanto curso tem determinada hora

> select carga, count(nome) from cursos

group by carga;

1)selecionar todos do sexo feminino

> select \* from gafanhotos

where sexo = 'f';

2)lista com os dados de todos aqueles que nasceram entre

01/jan/2000 e 31/dez/2015

> select \* from gafanhotos

where nascimento between '2000-01-01' and '2015-12-31';

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

Programadores

> select \* from gafanhotos

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

4)Um lista com os dados de todos as mulheres que nasceram no

Brasil e que tem seu nome iniciado com a letra J

> select \* from gafanhotos

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

5) Uma lista com o nome e nacionalidade de todos os homens

que tem silva no nome, nao 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 maior altura entre gafanhotos homens que moram no

brasil

> select max(altura) from gafanhotos

where sexo = 'M' and nacionalidade = 'Brasil';

7) Qual é a media de peso dos gafanhotos cadastrados

> select avg(peso) from gafanhotos;

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';

9) Quantas gafanhotos mulheres tem mais de 1.90m de altura

> select \* from gafanhotos

where sexo = 'F' and altura > '1.90';

exercio 2

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

> select profissao, count(\*) from gafanhotos

group by profissao;

2) Quantos gafanhotos homens e quantas mulheres nasceram após 01/jan/2005?

> select sexo, count(\*) from gafanhotos

where nascimento > '01-01-2005'

group by sexo;

3) Uma lista com os gafanhotos que naceram fora do brasil, mostrando o país de origem e o 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 altura, count(\*) from gafanhotos

where peso > '100'

group by altura

having altura > (select avg(altura) from gafanhotos);

tabela gafanhotos vai add cursopreferido na referencia curso(idcurso)

> alter table gafanhotos

add foreign key (cursopreferido)

references cursos(idcurso);

na tabela gafanhotos vai ter cursospreferido na linha 6

com primeiro id

> update gafanhotos set cursopreferido = '6' where id = '1';

vai relacionar duas tabela aparecendo nome do usuário, nome do curso e ano.

> select gafanhotos.nome, cursos.nome, cursos.ano

from gafanhotos join cursos

on cursos.idcurso = gafanhotos.cursopreferido;

mostro agora todos os alunos da left esquerda

> select g.nome, c.nome, c.ano

from gafanhotos as g left join cursos as c

on c.idcurso = g.cursopreferido;

mostro agora todos os cursos da right esquerda

> select g.nome, c.nome, c.ano

from gafanhotos as g right join cursos as c

on c.idcurso = g.cursopreferido;

criar tabela referencia "FOREING KEY"

> create table gafanhotos\_assiste\_cruso(

id int not null auto\_increment,

data date,

idgafanhoto int,

idcurso int,

primary key (id),

foreign key (idgafanhoto) references gafanhotos(id),

foreign key (idcurso) references cursos(idcurso)

) default charset = utf8;

puxa dados de 3 tabela

> select g.nome, c.nome from gafanhotos g

join gafanhoto\_assiste\_curso a

on g.id = a.idgafanhoto

join cursos c

on c.idcurso = a.idcurso

order by g.nome;