Diagrama

Descrição gerada automaticamente

create database exercicio13;

use exercicio13;

create table Departamento

(

idDepto int primary key,

nomeDepto varchar(30),

fkGerente int,

dataInicioGer date

)

;

create table Funcionario

(

idFunc int primary key,

nomeFunc varchar(30),

salario decimal(7,2) check (salario > 0),

sexo char(1) check (sexo = 'm' or sexo = 'f' or sexo = 'n'),

fkSupervisor int,

dataNasc date,

fkDepto int,

foreign key(fkDepto) references Departamento(idDepto)

)

;

create table Projeto

(

idProj int primary key,

nomeProj varchar(30),

localProj varchar(30),

fkDepto int,

foreign key(fkDepto) references Departamento(idDepto)

)

;

create table FuncProj

(

fkFunc int,

foreign key(fkFunc) references Funcionario(idFunc),

fkProj int,

foreign key(fkProj) references Projeto(idProj),

primary key (fkFunc, FkProj),

horas decimal(3,1)

)

;

insert into Departamento values

(105, 'Pesquisa', 2, '2008-05-22'),

(104, 'Administração', 7, '2015-01-01'),

(101, 'Matriz', 8, '2001-06-19')

;

insert into Funcionario values

(1, 'Joao Silva', 3500, 'm', 2, '1985-01-09', 105),

(2, 'Fernando Wong', 4500, 'm', 8, '1975-12-08', 105),

(3, 'Alice Sousa', 2500, 'f', 7, '1988-01-19', 104),

(4, 'Janice Morais', 4300, 'f', 8, '1970-06-20', 104),

(5, 'Ronaldo Lima', 3800, 'm', 1, '1982-09-15', 105),

(6, 'Joice Leite', 2500, 'f', 1, '1992-07-31', 105),

(7, 'Antonio Pereira', 2500, 'm', 4, '1989-03-29', 104),

(8, 'Juliano Brito', 5500, 'm', NULL, '1957-11-10', 101)

;

insert into Projeto values

(1, 'Produto X', 'Santo André', 105),

(2, 'Produto Y', 'Itu', 105),

(3, 'Produto Z', 'São Paulo', 105),

(10, 'Informatização', 'Mauá', 104),

(20, 'Reorganização', 'São Paulo', 101),

(30, 'Benefícios', 'Mauá', 104)

;

insert into FuncProj values

(1, 1, 32.5),

(1, 2, 7.5),

(5, 3, 40.0),

(6, 1, 20.0),

(6, 2, 20.0),

(2, 2, 10.0),

(2, 3, 10.0),

(2, 10, 10.0),

(2, 20, 10.0),

(3, 30, 30.0),

(3, 10, 10.0),

(7, 10, 35.0),

(7, 30, 5.0),

(4, 30, 20.0),

(4, 20, 15.0),

(8, 20, null)

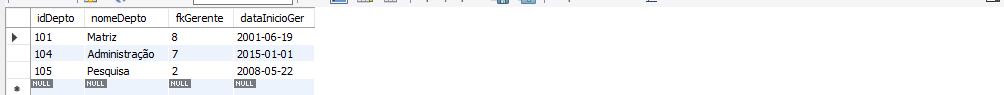
;

alter table Funcionario add foreign key (fkSupervisor) references Funcionario(idFunc);

alter table Departamento add foreign key (fkGerente) references Funcionario(idFunc);

-- 1. --

select \* from Departamento;



select \* from Funcionario;

Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamente

select \* from Projeto;

Interface gráfica do usuário

Descrição gerada automaticamente com confiança baixa

select \* from FuncProj;

Imagem em preto e branco

Descrição gerada automaticamente

-- 2. --

insert into Funcionario values

(null, 'Cecília Ribeiro', 2800, 'f', null, '1980-04-05', 104)

;

-- Não funcinou, pois idFunc está vazio

-- 3. --

insert into Funcionario values

(3, 'Alice Sousa', 2800, 'f', 4, '1980-04-05', 104)

;

-- Não funcinou, idFunc duplicada

-- 4. --

insert into Funcionario values

(9, 'Cecília Ribeiro', 2800, 'f', 4, '1980-04-05', 107)

;

-- Não funcinou, valor da foreign key fkDepto não existe

-- 5. --

insert into Funcionario values

(9, 'Cecília Ribeiro', 2800, 'f', 4, '1980-04-05', 104)

;

-- Sim, todos os valores certos

-- 6. --

delete from FuncProj where fkFunc = 3 and fkProj = 10;

-- Excluiu sim, pois há as duas primary keys informadas

-- 7. --

delete from Funcionario where idFunc = 4;

-- Não excluiu, pois esse funcionário é um supervisor, logo ele é necessário para outros regisros

-- 8. --

delete from Funcionario where idFunc = 2;

-- Não excluiu, pois esse funcionário é um supervisor, logo ele é necessário para outros regisros

-- 9. --

update Funcionario set salario = 2800 where idFunc = 3;

-- Foi alterado

-- 10. --

update Funcionario set fkDepto = 101 where idFunc = 3;

-- Foi alterado

-- 11. --

update Funcionario set fkDepto = 107 where idFunc = 3;

-- Não foi alterado, pois não existe a idDepto 107 na tabela Departamento

-- 12. --

select dataNasc, salario from Funcionario where nomeFunc = 'João Silva';



-- 13. --

select salario from Funcionario;

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

-- 14. --

select distinct(salario) as 'Salários Distintos' from Funcionario;

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente com confiança média

-- 15. --

select \* from Funcionario order by nomeFunc;

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

-- 16. --

select \* from Funcionario order by salario desc;

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

-- 17. --

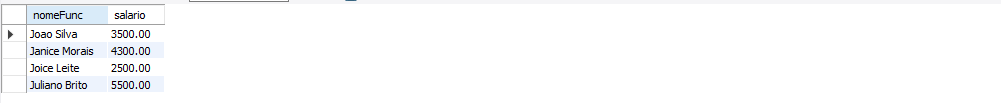
select \* from Funcionario where salario >= 2000 and salario <= 4000;

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

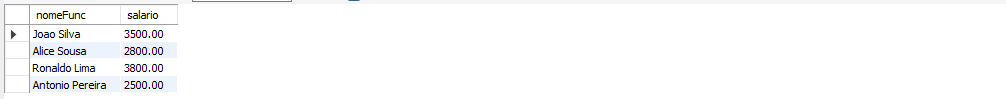
-- 18. --

select nomeFunc, salario from Funcionario where nomeFunc like 'J%';



-- 19. --

select nomeFunc, salario from Funcionario where nomeFunc like '%a';



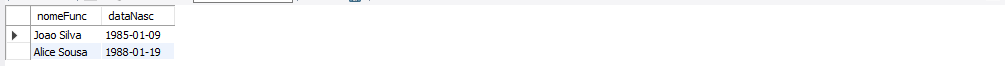
-- 20. --

select nomeFunc from Funcionario where nomeFunc like '\_\_n%';



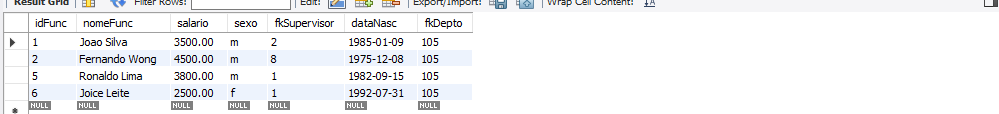
-- 21. --

select nomeFunc, dataNasc from Funcionario where nomeFunc like '%s\_\_\_\_';



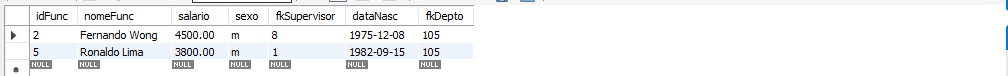
-- 22. --

select \* from Funcionario where fkDepto = 105;



-- 23. --

select \* from Funcionario where fkDepto = 105 and salario > 3500;



-- 24. --

select \* from Funcionario where fkDepto = 105 and nomeFunc like 'J%';



-- 25. --

select f.idFunc as 'idFuncionario', f.nomeFunc as 'Funcionário', s.idFunc as 'idSupervisor', s.nomeFunc as 'Supervisores'

from Funcionario as f join Funcionario as s

on f.fkSupervisor = s.idFunc;

Uma imagem contendo Forma

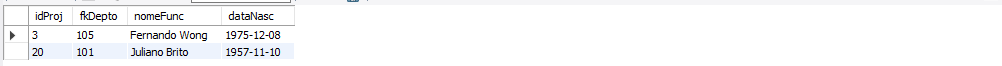
Descrição gerada automaticamente

-- 26. --

select idProj, Projeto.fkDepto, nomeFunc, dataNasc

from Projeto join Departamento on Projeto.fkDepto = idDepto and localProj = 'São Paulo'

join Funcionario on fkGerente = idFunc;



-- 27. --

select idFunc, nomeFunc, idProj, nomeProj, horas

from Funcionario join FuncProj on idFunc = fkFunc

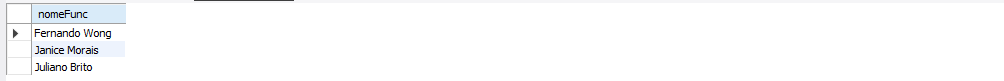
join Projeto on idProj = fkProj;

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

-- 28. --

select nomeFunc from Funcionario where dataNasc < '1980-00-00';



-- 29. --

select count(distinct(salario)) as 'Quantiadade de Salários Diferentes' from Funcionario;



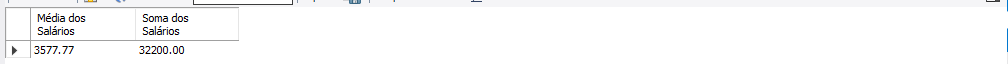
-- 30. --

select count(distinct(localProj)) as 'Quantiadade de Locais Diferentes do Projeto' from Projeto;



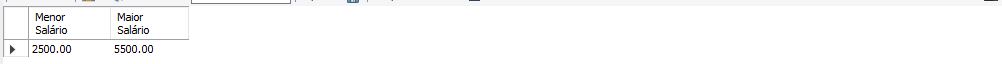
-- 31. --

select truncate(avg(salario),2) as 'Média dos Salários', sum(salario) as 'Soma dos Salários' from Funcionario;



-- 32. --

select min(salario) as 'Menor Salário', max(salario) as 'Maior Salário' from Funcionario;



-- 33. --

select idDepto, truncate(avg(salario),2) as 'Média dos Salários', sum(salario) as 'Soma dos Salários'

from Departamento join Funcionario on idDepto = fkDepto group by idDepto;



-- 34. --

select idDepto, min(salario) as 'Menor Salário', max(salario) as 'Maior Salário'

from Departamento join Funcionario on idDepto = fkDepto group by idDepto;



-- 35. --

insert into Funcionario values

(10, 'José da Silva', 1800, 'm', 3, '2000-10-12', null),

(11, 'Benedito Almeida', 1200, 'm', 5, '2001-09-01', null)

;

-- 36. --

insert into Departamento values

(110, 'RH', 3, '2018-11-10')

;

-- 37. --

select \* from Funcionario left join Departamento on fkDepto = idDepto;

Interface gráfica do usuário

Descrição gerada automaticamente

-- 38. --

select \* from Funcionario right join Departamento on fkDepto = idDepto;

Interface gráfica do usuário

Descrição gerada automaticamente