

Instituto Federal da Paraíba - IFPB

Banco de Dados

Pedro Henrique Guerra de Abreu

Data: 18/09/2021

Criação de tabelas:

```
CREATE DATABASE bancodedados;
```

```
CREATE TABLE DEPARTAMENTO(
```

```
    Nome varchar(50) NOT NULL,
```

```
    Sigla varchar(3) NOT NULL,
```

```
    Codigo int PRIMARY KEY
```

```
);
```

```
CREATE TABLE PROFESSOR(
```

```
    Nome varchar(50) NOT NULL,
```

```
    SNome varchar(50) NOT NULL,
```

```
    Matricula int NOT NULL,
```

```
    PRIMARY KEY(Matricula),
```

```
    DataNasc date,
```

```
    Sexo varchar(1),
```

```
    Salario decimal(8,2) CHECK(Salario>0),
```

```
    Matric_Coordenador_Area integer default null,
```

```
    Depto int
```

```
        FOREIGN KEY(Depto) REFERENCES departamento(Codigo)
```

```
);
```

Alterações na tabela para adicionar a chave primária de professor como chave estrangeira de departamento:

ALTER TABLE departamento ADD Coordenador INT

ALTER TABLE departamento ADD CONSTRAINT FOREIGN KEY(Coordenador) REFERENCES professor(Matricula)

Continuando criando as tabelas:

CREATE TABLE Projeto(

Nome varchar(50) NOT NULL,

Codigo int,

PRIMARY KEY(Codigo),

Depto int,

FOREIGN KEY (Depto) REFERENCES departamento (codigo)

);

CREATE TABLE Alocao(

MatricProf int,

FOREIGN KEY (MatricProf) REFERENCES professor (Matricula),

CodProj int,

FOREIGN KEY (CodProj) REFERENCES projeto (Codigo),

Horas time

);

CREATE TABLE Dependente(

MatricProf int,

FOREIGN KEY (MatricProf) REFERENCES professor (Matricula),

Nome varchar(50) NOT NULL,

RG int,

PRIMARY KEY(RG),

Sexo varchar(1),

DataNasc date

);

CREATE TABLE Telefone(

MatricProf int,

FOREIGN KEY (MatricProf) REFERENCES professor (Matricula),

Telefone int(12),

PRIMARY KEY(Telefone)

);

CREATE TABLE Email(

MatricProf int,

FOREIGN KEY (MatricProf) REFERENCES professor (Matricula),

Email varchar(50),

PRIMARY KEY(Email)

);

Inserindo dados nas tabelas:

INSERT INTO departamento(Codigo,Nome,Sigla) VALUES

(121,'Tec. Telemática','DTT'),

(122,'Tec. Construção de Edifícios','DCE'),

(125,'Eng. Computação','DEC');

**INSERT INTO professor(DataNasc,Matricula,Matricula_Coordenador_Area,Nome,SNome,S
alario,Sexo)VALUES**

('1982-03-25',121003,NULL,'Pedro','Pereira da Silva',4000,'M'),

('1978-04-10',121031,121003,'José Maria','Campos',7000,'M'),

('1980-11-27',122045,NULL,'Bento','Diniz Costa',4000,'M'),

('1994-12-30',122047,122045,'Ana Clara','Araujo Santos',7200,'F'),

('1976-07-22',125335,125332,'João Carlos','Matos Cavalcante',3700,'M'),

```
('1974-08-16',125331,125332,'Maria Luiza','Machado',6800,'F');  
( '1990-06-12',125332,NULL,'Joana Maria','Pereira',4500,'F',125);
```

Preenchendo as colunas Coordenador da tabela Departamento e Depto da tabela Professor:

```
UPDATE departamento SET Coordenador=121031 WHERE Codigo =121;
```

```
UPDATE departamento SET Coordenador=122047 WHERE Codigo =122;
```

```
UPDATE departamento SET Coordenador=125331 WHERE Codigo =125;
```

```
UPDATE professor SET Depto=121 WHERE Matricula =121041OR Matricula = 121031;
```

```
UPDATE professor SET Depto=122 WHERE Matricula =122040 OR Matricula = 122045;
```

```
UPDATE professor SET Depto=125 WHERE Matricula =125302 OR Matricula = 125305;
```

Inserindo dados nas tabelas:

```
INSERT INTO projeto(Codigo,Depto,Duracao_Dias,Nome) VALUES
```

```
(11,119,8,"Projeto Telemática 1"),  
(12,119,6,"Projeto Telemática 2"),  
(16,121,4,"Projeto Construção de Edifícios 1"),  
(17,121,5,"Projeto Construção de Edifícios 2"),  
(20,125,7,"Projeto Computação 1"),  
(21,125,9,"Projeto Computação 2");
```

```
INSERT INTO alocao(Codigo_Projeto,Horas,Matricula_Professor) VALUES
```

```
(11,'10:00:00',121031),  
(12,'15:00:00',121003),  
(16,'20:00:00',122045),  
(17,'12:00:00',122040),
```

(20,'16:00:00',125047),
(21,'20:00:00',125305);

INSERT INTO dependente(Nome,RG,DataNasc,Matricula_Professor,Sexo) VALUES

('Pedro',2715000,'2000-04-09',121031,'M'),
('Nayara',1787633,'2001-02-19',121031,'F'),
('Artur',9274589,'1995-12-11',122045,'M'),
('Maria',1257232,'1985-05-20',122045,'F'),
('Joao',3252783,'1974-06-29',125047,'M'),
('Manoel',5346373,'2003-08-15',125305,'M');

INSERT INTO telefone(Telefone,Matricula_Professor) VALUES

(988353781,121003),
(992375678,121031),
(995751400,122045),
(992346503,122047),
(999803530,125332),
(999140188,125335)
(998563320,125331);

INSERT INTO e_mail(Email,Matricula_Professor) VALUES

('Pedropereira@gmail.com',121003),
('Josemaria@gmail.com',121031),
('Bentodiniz@gmail.com',122045),
('Anaclara@gmail.com',122047),
('Joanamarca@gmail.com',125332),
('Joaocarlos@gmail.com',125335),
('Marialuiza@gmail.com',125331);

Consultas:

1. Exiba o nome, o RG e a data de nascimento de todos os dependentes do sexo feminino.

```
SELECT RG, Nome, DATA_NASCIMENTO FROM dependente WHERE Sexo = 'F'
```

2. Relacione em ordem decrescente os (diferentes) salários dos professores da instituição.

```
SELECT Nome, Salario FROM professor ORDER BY Salario DESC;
```

3. Modifique/atualize o nome do projeto “Aplicações do No SQL” para “Aplicações de Big Data”.

4. Exiba a matrícula e o nome dos professores que nasceram a partir de 1970, ordenados pelo nome em ordem ascendente.

```
SELECT Nome, Matricula FROM professor WHERE DataNasc >= '1970-00-00' ORDER BY DataNasc ASC;
```

5. Exiba em ordem decrescente da data de nascimento todos os dados dos professores que nasceram na década de 80 e que têm Pereira no sobrenome.

```
SELECT * FROM professor WHERE DataNasc >= '1980-00-00' AND DataNasc < '1990-00-00' AND sNome LIKE '%Pereira%' ORDER BY DataNasc DESC;
```

6. Exiba o nome e a data de nascimento do dependente mais jovem.

```
SELECT Nome, MAX(DataNasc) as DataNasc FROM dependente
```

7. Exibir a matrícula e o nome de todas as professoras que tenham Maria em qualquer parte do nome ordenados pelo nome em ordem decrescente.

SELECT Nome,Matricula FROM professor WHERE Nome LIKE '%Maria%' OR SNome LIKE '%Maria%' ORDER BY Nome DESC;

8. Para cada departamento, exiba o código do departamento e a quantidade de projetos lá alocados.

DEPARTAMENTO 121: TELEMATICA

SELECT departamento.Codigo,COUNT(projeto.Codigo) as Quantidade_Porjetos FROM departamento, projeto WHERE departamento.Codigo = 121 AND projeto.Depto = 121

DEPARTAMENTO 122: Tec. Construção de Edifícios

SELECT departamento.Codigo,COUNT(projeto.Codigo) as Quantidade_Porjetos FROM departamento, projeto WHERE departamento.Codigo = 122 AND projeto.Depto = 122

DEPARTAMENTO 125: Eng. Computação

SELECT departamento.Codigo,COUNT(projeto.Codigo) as Quantidade_Porjetos FROM departamento, projeto WHERE departamento.Codigo = 125 AND projeto.Depto = 125

9. Exibir a média de salário dos professores com salário entre R\$ 4000,00 e R\$ 7000,00 que não estejam lotados nos departamentos 121 e 125.

SELECT Nome,AVG(Salario) as Media_salarial FROM professor WHERE Depto <> 121 and Depto <> 125 and Salario BETWEEN 4000 AND 7000;

10. Exiba o nome e o código dos departamentos que têm mais de 2 professores nele lotados.

SELECT departamento.Nome,departamento.Codigo FROM professor, departamento GROUP BY departamento.Codigo, professor.Depto HAVING professor.Depto = departamento.Codigo AND COUNT(professor.Depto) >2;

11. Selecione o nome, o sobrenome e a matrícula dos professores que trabalham no Departamento de Tecnologia em Telemática.

SELECT professor.nome, professor.SNome, professor.Matricula FROM professor, departamento WHERE departamento.Codigo = professor.Depto AND departamento.Nome LIKE '%Telematica%'

12. Exiba o e-mail do professor cujo nome é João Carlos.

SELECT e_mail.Email FROM e_mail, professor WHERE professor.Matricula = e_mail.Matricula_Professor AND professor.Nome LIKE 'João Carlos'

13. Exiba o nome do(a) coordenador(a) do Departamento de Engenharia de Computação

SELECT professor.Nome FROM professor, departamento WHERE departamento.Coordenador = professor.Matricula AND departamento.Nome LIKE '%Computação%'

14. Exiba os nomes dos professores que não têm dependentes.

15. Relacione o nome dos professores que têm menos de 3 dependentes.

SELECT professor.Nome FROM professor, dependente GROUP BY professor.Matricula, dependente.Matricula_Professor HAVING professor.Matricula = dependente.Matricula_Professor AND COUNT(dependente.Matricula_Professor) < 3;

