

Grupo:

Gustavo Santos, Mauricio Santiago e Daniel Guedes.

Requisitos:

Nesta tarefa vocês devem colocar um requisito de negócios de algum domínio que desejem criar o Banco de Dados. O requisito de negócio deve ter no mínimo 1/2 página descrita, não pode ser composto somente por 1 parágrafo. Deve ser enriquecido de detalhes para que a modelagem seja realizada com riqueza.

Regras de Negócio

Sistema para controle de escolas particulares.

Uma escola tem como atributos um id único, data de fundação, nome e endereço (rua, cep, cidade estado número). Uma escola possui salas, As salas são identificadas pelo número delas e a qual escola pertence. Toda sala tem a sua capacidade de pessoas, está associada a uma lousa e a sala pode estar ativa ou não.

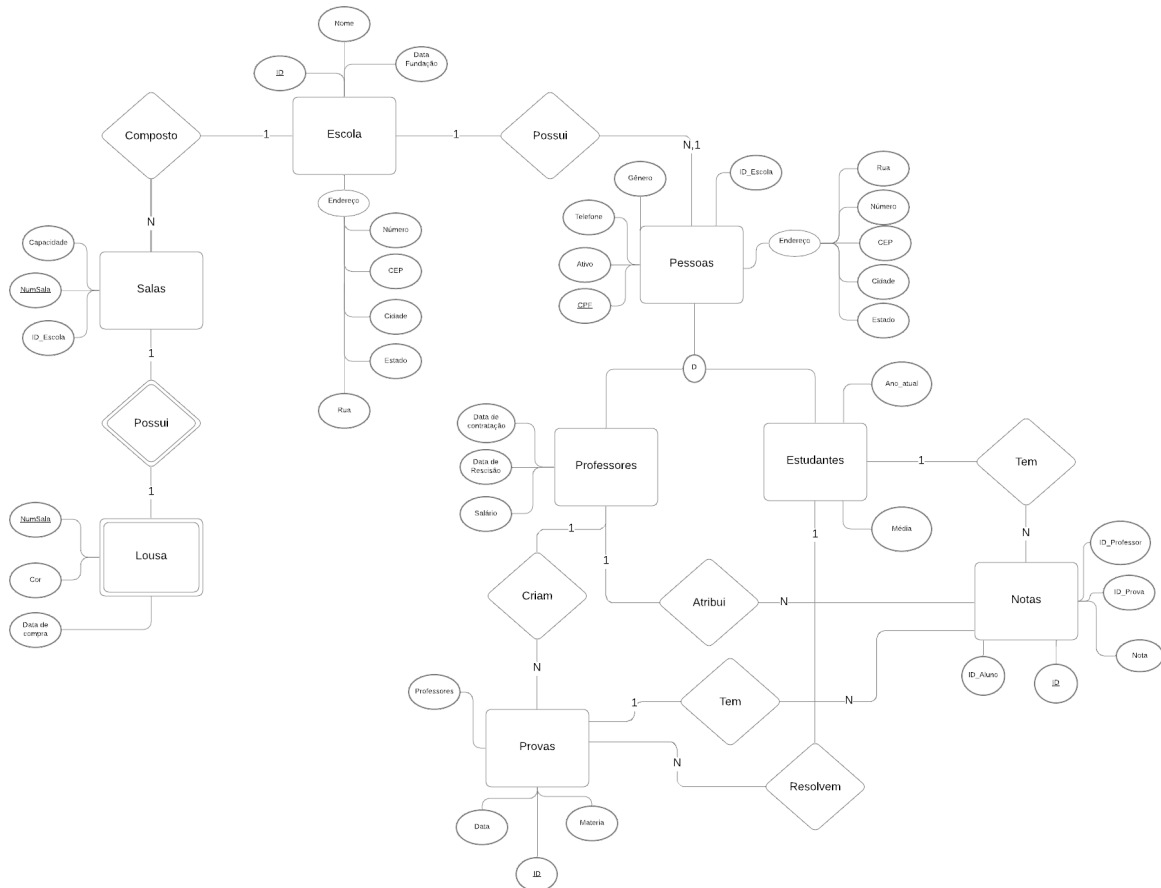
Uma escola pode possuir várias pessoas, porém as pessoas fazem parte apenas de uma escola, as pessoas têm como atributos: Gênero, Endereço (Rua, Número, CEP, Cidade, Estado), Telefone, CPF, Ativo, Id Único.

Cada pessoa pode ser ou professor ou aluno, sendo professor possui: Salário, Data de contratação, data de rescisão. Já o aluno possui um ano letivo e a sua média.

Toda escola possui provas que são aplicadas por um professor para uma matéria com uma data e tendo id único. Provas quando corrigidas pelos professores geram uma nota para um aluno a partir do id da prova

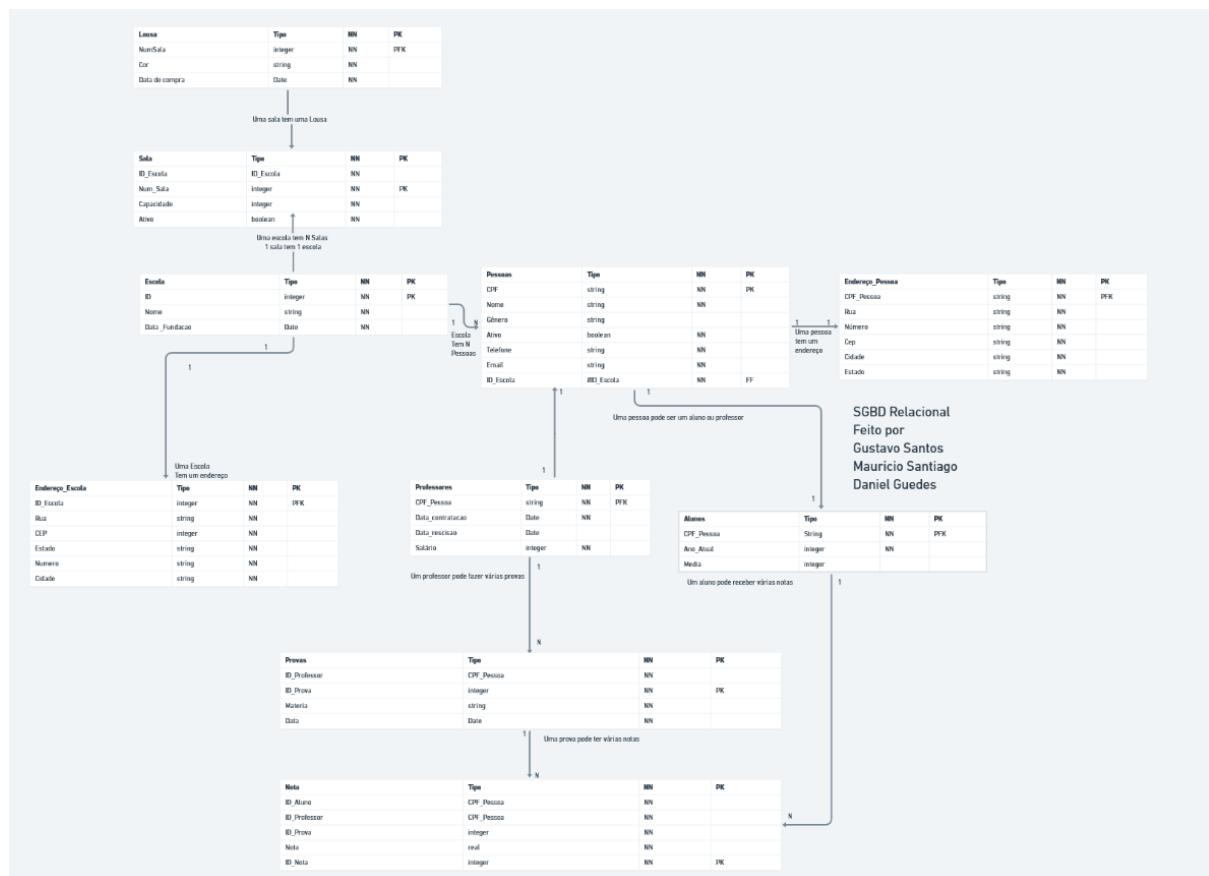
MER

<https://lucid.app/lucidchart/ab47c537-6866-4aab-9e0f-9f67b1727396/edit?shared=true>



MER Relacional

<https://whimsical.com/sghd-relacional-gustavo-mauricio-daniel-UGWYwFNxtkH7rN5HUDmxBk>



Relax

Construção do banco no relax e da álgebra para consultas e operações

Banco:

group: Escolas

```
Escola = {
  ID, Data_fundacao, nome
  1, '24-12-1993', 'Oicirum'
  2, '02-02-1992', 'Ovatsug'
  3, '02-02-1999', 'Leinad'
}
```

```
Endereco_Escola = {
  ID_escola, Rua, CEP, Estado, Numero, Cidade
  1, 'Mau maioral', 41810051, 'BA', 735, 'Pocoos'
```

```
2, 'Dan general', 12345678, 'BA', 782, 'Salvador'
3, 'Gustavo maior', 42706400, 'BA', 799, 'Lauro'
}
```

```
Pessoas = {
CPF, Nome, Genero, Ativo, Telefone, Email, ID_escola
'51007189002', 'Mau', 'Masculino', 'false', '71999037600', 'm@gmail.com', 3
'52290334014', 'Dan', 'Masculino', 'true', '71999037600', 'd@gmail.com', 2
'52290334013', 'Gustavo', 'Masculino', 'true', '71999037600', 'g@gmail.com', 1
'14593103061', 'Mia', 'Feminino', 'false', '71999037600', 'mia@gmail.com', 3
'62630974006', 'Mariana', 'Feminino', 'false', '71999037600', 'mari@gmail.com', 3
'92571181009', 'Luciana', 'Feminino', 'true', '71999037600', 'luci@gmail.com', 2
'90713100001', 'Seb', 'Masculino', 'false', '71999037600', 'seb@gmail.com', 1
'57941471051', 'Marivaldo', 'Masculino', 'false', '71999037600', 'mar@gmail.com', 2
'80259054038', 'Ligia', 'Feminino', 'true', '71999037600', 'lig@gmail.com', 2
'51768916039', 'Titia', 'Feminino', 'true', '71999037600', 'titi@gmail.com', 3
'51768916038', 'TioNorton', 'Masculino', 'true', '71999037600', 'norton@gmail.com', 2
'51768916037', 'Rogerio', 'Masculino', 'false', '71999037600', 'rog@gmail.com', 1
'25459220050', 'Nortinho', 'Masculino', 'true', '71999037600', 'jr@gmail.com', 1
'75070695088', 'Michelle', 'Feminino', 'true', '71999037600', 'miche@gmail.com', 3
'02318462070', 'Paula Mcarthy', 'Feminino', 'true', '71999037600', 'arthy@gmail.com', 2
'32041922028', 'Catherine zetaJones', 'Feminino', 'true', '71999037600', 'cat@gmail.com', 2
'88212687058', 'Kamilla', 'Feminino', 'true', '71999037600', 'Kat@gmail.com', 2
'78903119070', 'Jully', 'Feminino', 'true', '71999037600', 'jully@gmail.com', 3
'44763829084', 'Brenda', 'Feminino', 'true', '71999037600', 'b@gmail.com', 2
'70920653057', 'Pamela', 'Feminino', 'true', '71999037600', 'lombra@gmail.com', 3
}
```

```
Endereco_Pessoa = {
Cpf_pessoa, rua, numero, cep, cidade, estado
'51007189002', 'Maioral', 731, '41810051', 'Pocoas', 'BA'
'52290334013', 'Maioral', 123, '41810051', 'Salvador', 'BA'
'52290334014', 'Maioral', 823, '41810051', 'Salvador', 'BA'
'14593103061', 'Maioral', 745, '41810051', 'Sao paulo', 'SP'
'90713100001', 'minor', 861, '41810051', 'Sao Paulo', 'SP'
'57941471051', 'minor', 12, '41810051', 'Rio de Janeiro', 'SP'
'80259054038', 'ligia Maioral', 128, '41810051', 'Brasilia', 'DF'
'62630974006', 'Mari Maioral', 18, '41810051', 'Salvador', 'BA'
'92571181009', 'luciana Maioral', 1, '41810051', 'Fortaleza', 'PA'
'51768916039', 'Titia Maioral', 1234, '41810051', 'Belo Horizonte', 'MG'
'51768916038', 'minor', 246, '41810051', 'Manaus', 'AM'
'51768916037', 'minor', 3124, '41810051', 'Curitiba', 'PR'
'25459220050', 'Maioral', 36, '41810051', 'Recife', 'PE'
'75070695088', 'Miche Maioral', 765, '41810051', 'Goiania', 'GO'
'2318462070', 'Paula Maioral', 675, '41810051', 'Belem', 'PA'
'32041922028', 'Maioral', 763, '41810051', 'Salvador', 'BA'
'88212687058', 'Kami Maioral', 2345, '41810051', 'Salvador', 'BA'
'78903119070', 'Jully Maioral', 96, '41810051', 'Salvador', 'BA'
}
```

```
'44763829084', 'minoral', 901, '41810051', 'Salvador', 'BA'
'70920653057', 'minoral', 743, '41810051', 'Salvador', 'BA'
}
```

```
Sala = {
ID_escola, Num_sala, capacidade, ativo
1, 1, 10, 'false'
1, 2, 10, 'true'
1, 3, 10, 'false'
2, 101, 30, 'true'
2, 102, 30, 'true'
3, 201, 60, 'true'
3, 202, 100, 'true'
}
```

```
Lousa = {
    NumSala, Cor, Data_compra
    1, 'Branca', '1993-12-24'
    2, 'Branca', '1993-12-24'
    3, 'Branca', '1994-12-24'
    101, 'Preta', '1992-02-02'
    102, 'Preta', '1992-02-02'
    201, 'Azul', '1999-04-04'
    202, 'Azul', '1999-04-04'
}
```

```
Professores = {
CPF_pessoa, Data_contratacao, Data_recisao, Salario
'51007189002', 1993-12-24, '-', 15000
'52290334014', 1993-12-24, '-', 15000
'52290334013', 1993-12-24, '-', 15000
'14593103061', 1999-12-24, '-', 10000
'62630974006', 2000-09-24, '-', 16000
'92571181009', 2002-09-24, '-', 8000
}
```

```
Alunos = {
CPF_pessoa, ano_atual, Media
'90713100001', 1,5
'57941471051', 2,2
'80259054038', 3,7
'51768916039', 1,2
'51768916038', 2,6
'51768916037', 3,4
'25459220050', 1,9
'75070695088', 2,2
}
```

```
'02318462070',3,9
'32041922028',1,1
'88212687058',2,5
'78903119070',3,1
'44763829084',1,10
'70920653057',2,2
```

```
}
```

```
Provas = {
ID,ID_professor, Materia, Data_app
1,'51007189002','por','2000-12-11'
2,'52290334014','por','2000-09-09'
3,'52290334013','por','2000-02-13'
4,'51768916039','qui','2000-03-14'
5,'70920653057','qui','2000-08-15'
}
```

```
Notas = {
      ID,ID_aluno, ID_professor,ID_prova,Notas
      1,'51768916038','51007189002',1,9
      2,'90713100001','51007189002',1,5
      3,'57941471051','52290334014',2,1
      4,'80259054038','52290334014',2,2
      5,'51768916039','52290334014',2,3
}
```

Álgebra:

```
/* Lista de emails da escola:*/
 $\pi$ Email, Nome (Pessoas)
```

```
/* Lista de emails do departamento de educação*/
 $X = \sigma_{ID\_escola = 2}$  (Pessoas)
 $\pi$ Email, Nome (X)
```

```
/* Quem mora na rua maioral */
 $X = \sigma_{rua = 'Maioral'}$  (Endereco_Pessoa)
 $Y = \pi_{Cpf\_pessoa}$  (X)
 $W = (Y) \bowtie_{Pessoas.CPF = Cpf\_pessoa}$  (Pessoas)
 $\pi_{CPF, Nome, Email}$  (W)
```

```
/* Capacidade do colégio */
 $y ; \sum(Sala.capacidade) \rightarrow x$  ( Sala )
```

/* Media Salárial */

γ ; avg(Professores.Salario)->x (Professores)

/* Full Join que junta a pessoa a escola que ela vai */

((Pessoas) \bowtie Pessoas.CPF=Cpf_pessoa (Endereco_Pessoa)) \bowtie Endereco_Pessoa.cidade
=Cidade Endereco_Escola

/* Pessoas que não moram nas ruas minoral e Maioral*/

$X = \sigma \text{ rua} \neq \text{'minoral'} \wedge \text{ rua} \neq \text{'Maioral'} (\text{Endereco_Pessoa})$

$(X) \bowtie \text{Pessoas.CPF=Cpf_pessoa (Pessoas)}$

/*Quantas Louças estão em salas inativas? qual a cor delas? E qual a escola dessas
Lousas? */

$A = \sigma \text{ativo='false'} (\text{Sala})$

$B = \pi \text{ID_escola,Num_sala (A)}$

$C = \pi \text{NumSala,Cor (Lousa)}$

$D = (B) \bowtie \text{Sala.Num_sala=Lousa.NumSala (C)}$

$\pi \text{ID_escola,NumSala,Cor (D)}$

/* Capacidade por escola*/

$x = \gamma \text{ Sala.ID_escola; count(Sala.ID_escola)->NumeroSala (Sala)}$

$y = \gamma \text{ Sala.ID_escola ; sum(Sala.capacidade)->Capacidade (Sala)}$

$(x) \bowtie (y)$

/*Mulhere e Homens, quem está mais ativo? */

$Y = \sigma \text{Genero='Feminino'} \wedge \text{Ativo='true'} (\text{Pessoas})$

$X = \sigma \text{Genero='Masculino'} \wedge \text{Ativo='true'} (\text{Pessoas})$

$M = \gamma \text{ ;count(Pessoas.Ativo)→Quantidade_de_mulheres_ativas (Y)}$

$H = \gamma \text{ ;count(Pessoas.Ativo)→Quantidade_de_Homens_ativos (X)}$

$(M) \times (H)$

/*Media das notas dos alunos das provas aplicadas*/

$\text{media} = \gamma \text{ ID_prova,ID_professor;AVG(Notas)→media_notas (Notas)}$

$\text{media_nome} = ((\text{media}) \bowtie \text{Notas.ID_professor} = \text{Pessoas.CPF (Pessoas)})$

media_nome2 = π Notas.ID_prova,media_notas,Pessoas.Nome \rightarrow Professor_nome
(media_nome)

media_nome3 = ((media_nome2) \bowtie Notas.ID_prova=Provas.ID (Provas))

π Notas.ID_prova \rightarrow ID_prova
,media_notas
,Professor_nome
,Provas.Materia \rightarrow Materia
,Provas.Data_app \rightarrow Data_de_aplicacao
(media_nome3)

SQL

Banco:

```
CREATE TABLE Escolas
(
  ID INT not null,
  Nome char (20) not null,
  Data_fundacao Date not null,
  PRIMARY KEY(ID)
);
```

```
INSERT INTO Escolas
('ID`,`Nome`,`Data_fundacao`)
VALUES
('1','escola1','1993-12-24'),
('2','escola2','1992-02-02'),
('3','escola3','1999-02-02');
```

```
CREATE TABLE Endereco_Escola
(
  ID_escola INT not null,
  Rua char(200) not null,
  CEP char(8) not null,
  Estado char(3) not null,
  Numero char(5) not null,
  Cidade char(30) not null,
  PRIMARY KEY(ID_escola),
  FOREIGN KEY (ID_escola) REFERENCES Escolas(ID)
);
```

```
INSERT INTO Endereco_Escola
('ID_escola`,`Rua`,`CEP`,`Estado`,`Numero`,`Cidade`)
```


VALUES

```
(1, 'Mau maior', '41810051', 'BA', '735', 'Pocoas'),
(2, 'Dan general', '12345678', 'BA', '782', 'Salvador'),
(3, 'Gustavo maior', '42706400', 'BA', '799', 'Lauro');
```

CREATE TABLE Pessoas

```
(
CPF char(20) not null,
Nome char (20) not null,
Genero char (20),
Ativo boolean not null,
Telefone char(20) not null,
Email char(20) not null,
ID_escola INT not null,
PRIMARY KEY(CPF),
FOREIGN KEY (ID_escola) REFERENCES Escolas(ID)
);
```

INSERT INTO Pessoas

```
(`CPF`, `Nome`, `Genero`, `Ativo`, `Telefone`, `Email`, `ID_escola`)
```

VALUES

```
('51007189002', 'Mau', 'Masculino', FALSE, '71999037600', 'm@gmail.com', '3'),
('52290334014', 'Dan', 'Masculino', TRUE, '71999037600', 'd@gmail.com', '2'),
('52290334013', 'Gustavo', 'Masculino', TRUE, '71999037600', 'g@gmail.com', '1'),
('14593103061', 'Mia', 'Feminino', FALSE, '71999037600', 'mia@gmail.com', '3'),
('62630974006', 'Mariana', 'Feminino', FALSE, '71999037600', 'mari@gmail.com', '3'),
('92571181009', 'Luciana', 'Feminino', TRUE, '71999037600', 'luci@gmail.com', '2'),
('90713100001', 'Seb', 'Masculino', FALSE, '71999037600', 'seb@gmail.com', '1'),
('57941471051', 'Marivaldo', 'Masculino', FALSE, '71999037600', 'mar@gmail.com', '2'),
('80259054038', 'Ligia', 'Feminino', TRUE, '71999037600', 'lig@gmail.com', '2'),
('51768916039', 'Titia', 'Feminino', TRUE, '71999037600', 'titi@gmail.com', '3'),
('51768916038', 'TioNorton', 'Masculino', TRUE, '71999037600', 'norton@gmail.com', '2'),
('51768916037', 'Rogerio', 'Masculino', FALSE, '71999037600', 'rog@gmail.com', '1'),
('25459220050', 'Nortinho', 'Masculino', TRUE, '71999037600', 'jr@gmail.com', '1'),
('75070695088', 'Michelle', 'Feminino', TRUE, '71999037600', 'miche@gmail.com', '3'),
('02318462070', 'Paula MCarthy', 'Feminino', TRUE, '71999037600', 'arthy@gmail.com', '2'),
('32041922028', 'Catherine zetaJones', 'Feminino', TRUE, '71999037600', 'cat@gmail.com', '2'),
('88212687058', 'Kamilla', 'Feminino', TRUE, '71999037600', 'Kat@gmail.com', '2'),
('78903119070', 'Jully', 'Feminino', TRUE, '71999037600', 'jully@gmail.com', '3'),
('44763829084', 'Brenda', 'Feminino', TRUE, '71999037600', 'b@gmail.com', '2'),
('70920653057', 'Pamela', 'Feminino', TRUE, '71999037600', 'lombra@gmail.com', '3');
```

CREATE TABLE Endereco_pessoa (

```
CPF_pessoa char(20) not null,
```

```

Rua char(20) not null,
Numero char(200) not null,
Cep char(20) not null,
Cidade char(20) not null,
Estado char(20) not null,
PRIMARY KEY(CPF_pessoa),
FOREIGN KEY (CPF_pessoa) REFERENCES Pessoas(CPF)
);

```

```

INSERT INTO Endereco_pessoa
('CPF_pessoa', 'Rua', 'Numero', 'Cep', 'Cidade', 'Estado')
VALUES
('51007189002', 'Maioral', '731', '41810051', 'Pocoas', 'BA'),
('52290334014', 'Maioral', '123', '41810051', 'Salvador', 'BA'),
('52290334013', 'Maioral', '823', '41810051', 'Salvador', 'BA'),
('14593103061', 'Maioral', '745', '41810051', 'Sao paulo', 'SP'),
('90713100001', 'minoral', '861', '41810051', 'Sao Paulo', 'SP'),
('57941471051', 'minoral', '12', '41810051', 'Rio de Janeiro', 'SP'),
('80259054038', 'ligia Maioral', '128', '41810051', 'Brasilia', 'DF'),
('62630974006', 'Mari Maioral', '18', '41810051', 'Salvador', 'BA'),
('92571181009', 'luciana Maioral', '1', '41810051', 'Fortaleza', 'PA'),
('51768916039', 'Titia Maioral', '1234', '41810051', 'Belo Horizonte', 'MG'),
('51768916038', 'minoral', '246', '41810051', 'Manaus', 'AM'),
('51768916037', 'minoral', '3124', '41810051', 'Curitiba', 'PR'),
('25459220050', 'Maioral', '36', '41810051', 'Recife', 'PE'),
('75070695088', 'Miche Maioral', '765', '41810051', 'Goiania', 'GO'),
('02318462070', 'Paula Maioral', '675', '41810051', 'Belem', 'PA'),
('32041922028', 'Maioral', '763', '41810051', 'Salvador', 'BA'),
('88212687058', 'Kami Maioral', '2345', '41810051', 'Salvador', 'BA'),
('78903119070', 'Jully Maioral', '96', '41810051', 'Salvador', 'BA'),
('44763829084', 'minoral', '901', '41810051', 'Salvador', 'BA'),
('70920653057', 'minoral', '743', '41810051', 'Salvador', 'BA');

```

```

CREATE TABLE Salas (
ID_escola INT not null,
Num_sala INT not null,
Capacidade INT not null,
Ativo boolean not null,
PRIMARY KEY(Num_sala),
FOREIGN KEY (ID_Escola) REFERENCES Escolas(ID)
);

```

```

INSERT INTO Salas
('ID_escola', 'Num_sala', 'capacidade', 'ativo')
VALUES
(1, 001, '10', FALSE),
(1, 002, '10', TRUE),

```

```
(1, 003, '10', FALSE),  
(2, 101, '30', TRUE),  
(2, 102, '30', TRUE),  
(3, 201, '60', TRUE),  
(3, 202, '100', TRUE);
```

```
CREATE TABLE Lousas (  
  NumSala INTEGER,  
  Cor VARCHAR(20),  
  Data_compra Date not null,  
  PRIMARY KEY(NumSala),  
  FOREIGN KEY (NumSala) REFERENCES Salas(Num_sala)  
);
```

```
INSERT INTO Lousas  
  (`NumSala`, `Cor`, `Data_compra`)  
VALUES  
  ('1', 'Branca', '1993-12-24'),  
  ('2', 'Branca', '1993-12-24'),  
  ('3', 'Branca', '1994-12-24'),  
  ('101', 'Preta', '1992-02-02'),  
  ('102', 'Preta', '1992-02-02'),  
  ('201', 'Azul', '1999-04-04'),  
  ('202', 'Azul', '1999-04-04');
```

```
CREATE TABLE Professores (  
  CPF_pessoa char(20) not null,  
  Data_contratacao Date not null,  
  Data_rescisao Date,  
  Salario INT not null,  
  PRIMARY KEY(CPF_pessoa),  
  FOREIGN KEY (CPF_pessoa) REFERENCES Pessoas(CPF)  
);
```

```
INSERT INTO Professores  
  (`CPF_pessoa`, `Data_contratacao`, `Data_rescisao`, `Salario`)  
VALUES  
  ('51007189002', '1993-12-24', NULL, '15000'),  
  ('52290334014', '1993-12-24', NULL, '15000'),  
  ('52290334013', '1993-12-24', NULL, '15000'),  
  ('14593103061', '1999-12-24', NULL, '10000'),  
  ('62630974006', '2000-09-24', NULL, '16000'),  
  ('92571181009', '2002-09-24', NULL, '8000');
```

```
CREATE TABLE Alunos (  
  CPF_pessoa char(20) not null,  
  Ano_atual INT not null,
```

```
Media INT,  
PRIMARY KEY(CPF_pessoa),  
FOREIGN KEY (CPF_pessoa) REFERENCES Pessoas(CPF)  
);
```

```
INSERT INTO Alunos  
(`CPF_pessoa`, `ano_atual`, `Media`)  
VALUES  
( '90713100001', '1', '5'),  
( '57941471051', '2', '2'),  
( '80259054038', '3', '7'),  
( '51768916039', '1', '2'),  
( '51768916038', '2', '6'),  
( '51768916037', '3', '4'),  
( '25459220050', '1', '9'),  
( '75070695088', '2', '2'),  
( '02318462070', '3', '9'),  
( '32041922028', '1', '1'),  
( '88212687058', '2', '5'),  
( '78903119070', '3', '1'),  
( '44763829084', '1', '10'),  
( '70920653057', '2', '2');
```

```
CREATE TABLE Provas(  
    ID INT not null,  
    ID_professor char(20) not null,  
    Materia char(100) not null,  
    Data_app Date not null,  
    PRIMARY KEY(ID),  
    FOREIGN KEY (ID_professor) REFERENCES Pessoas(CPF)  
);
```

```
INSERT INTO Provas  
(`ID`, `ID_professor`, `Materia`, `Data_app`)  
VALUES  
( '1', '51007189002', 'por', '2000-12-11'),  
( '2', '52290334014', 'por', '2000-09-09'),  
( '3', '52290334013', 'por', '2000-02-13'),  
( '4', '51768916039', 'qui', '2000-03-14'),  
( '5', '70920653057', 'qui', '2000-08-15');
```

```
CREATE TABLE Notas(  
    ID INT not null,  
    ID_aluno char(20) not null,  
    ID_professor char(20) not null,  
    ID_prova INT not null,
```

```

        Notas INT not null,
        PRIMARY KEY(ID),
        FOREIGN KEY (ID_aluno) REFERENCES Pessoas(CPF),
        FOREIGN KEY (ID_professor) REFERENCES Pessoas(CPF),
        FOREIGN KEY (ID_prova) REFERENCES Provas(ID)
    );

```

```

INSERT INTO Notas
('ID`, `ID_aluno`, `ID_professor`, `ID_prova`, `Notas`)
VALUES
('1', '51768916038', '51007189002', '1', '9'),
('2', '90713100001', '51007189002', '1', '5'),
('3', '57941471051', '52290334014', '2', '1'),
('4', '80259054038', '52290334014', '2', '2'),
('5', '51768916039', '52290334014', '2', '3');

```

Consultas SQL:

/*Junta as escolas que tiveram data de fundação maior ou igual a 1997-02-02 ordenando da mais antiga para a mais nova*/

```

SELECT * FROM Escolas WHERE Data_fundacao >= '1997-02-02' ORDER BY
Data_fundacao ASC;

```

/*Junta CEP com pessoas*/

```

SELECT Pessoas.Nome, Endereco_pessoa.CEP FROM Pessoas INNER JOIN
Endereco_pessoa ON
Pessoas.CPF=Endereco_pessoa.cpf_pessoa ORDER BY Pessoas.Nome;

```

/* Junta Nome com o salario dela*/

```

SELECT Pessoas.Nome, Professores.Salario FROM Pessoas
INNER JOIN
Professores
ON
Pessoas.CPF=Professores.CPF_pessoa
ORDER BY Professores.Salario DESC;

```

/*Capacidade nas salas por escola*/

```

SELECT
id_escola, numsalas, capacidade
FROM
    (SELECT Salas.ID_escola, COUNT(Salas.ID_escola) as numSalas FROM Salas
    GROUP BY Salas.ID_escola) numSalas
LEFT JOIN
    (SELECT Salas.ID_escola as duplicata, SUM(Salas.Capacidade) as capacidade
    FROM Salas GROUP BY Salas.ID_escola) capacidade
ON numSalas.ID_escola = capacidade.duplicata;

```

```

/*Quantos homens e mulheres estão ativos?*/
SELECT t1.quantidade_Mulheres_ativas,
       t2.quantidade_Homens_ativos
FROM
(
    SELECT COUNT(ATIVO) AS quantidade_Mulheres_ativas
    FROM Pessoas
    WHERE Pessoas.Genero = 'Feminino'
    AND Pessoas.Ativo=true
) t1
CROSS JOIN
(
    SELECT COUNT(ATIVO) AS quantidade_Homens_ativos
    FROM Pessoas
    WHERE Pessoas.Genero = 'Masculino'
    AND Pessoas.Ativo=true
) t2;

```

```

/*Media das Notas dos alunos das provas aplicadas*/

SELECT
    ID_prova,
    Media_notas,
    Nome AS Nome_professor,
    Data_app AS Data_aplicacao,
    Materia
FROM
(
    SELECT *
    FROM
        (
            SELECT ID_prova, ID_professor,
                   AVG(Notas) AS Media_notas
            FROM Notas
            GROUP BY ID_prova, ID_professor
        ) AS t1
    LEFT JOIN
        (SELECT CPF, Nome FROM Pessoas) AS t2
    on(t1.ID_professor = t2.CPF)
    )AS t3
LEFT JOIN
    (
        SELECT ID, Materia, Data_app FROM Provas
    ) AS t4
ON(t3.ID_prova = t4.ID);

```

