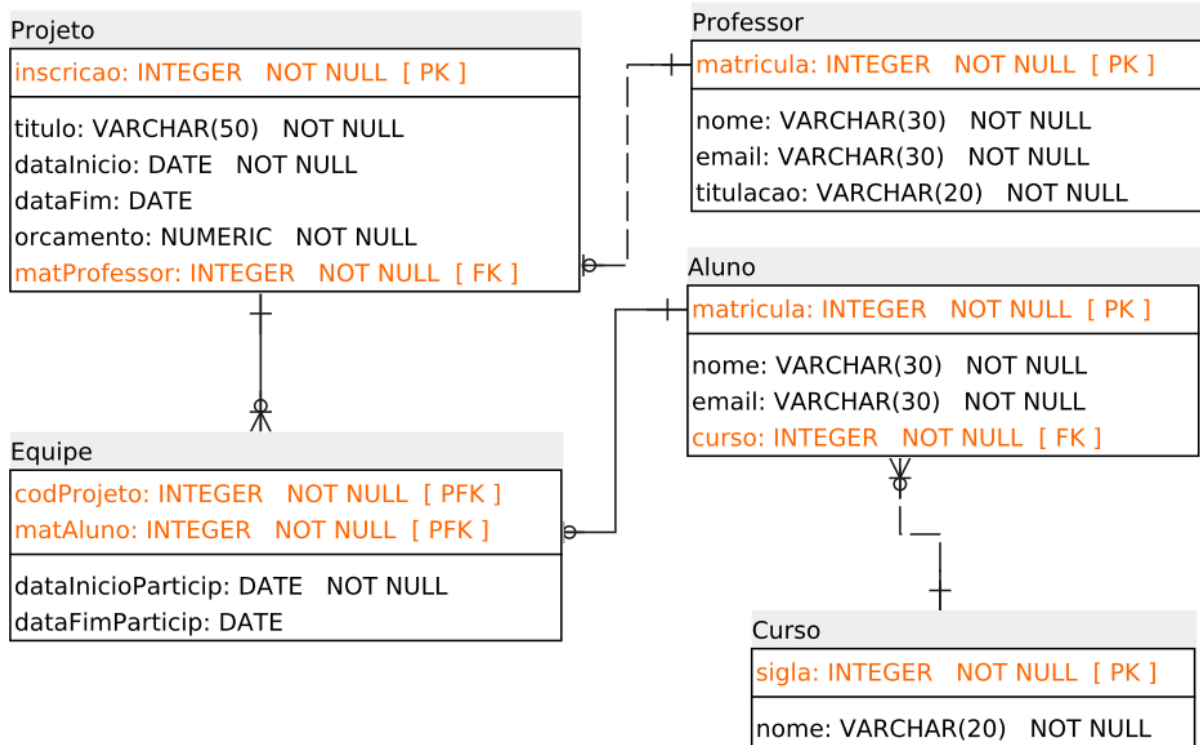


Exercícios de Banco de Dados II

Consultas com JOIN

Roberto Ribeiro Rocha – Univás

1) Utilizando o modelo físico a seguir, escreva as consultas para atender cada item abaixo:



1. Mostrar os projetos que começam no ano passado.
2. Mostrar a quantidade de projetos de cada ano.
3. Mostrar a quantidade de projetos e o valor do orçamento de cada professor.
4. Mostrar o nome e e-mail do professor e o título do projeto somente para os professores com titulação "doutor".
5. Mostrar a quantidade de projetos para cada titulação.
6. Mostrar o nome dos professores que possuem projetos que iniciaram no de 1º trimestre.
7. Mostrar o nome do projeto e a quantidade de alunos de cada projeto.
8. Mostrar o nome do curso e a média de duração dos projetos para cada curso.
9. Mostrar o nome dos alunos que ainda participam de algum projeto.
10. Mostrar a quantidade de alunos por projeto que ainda estão participando de algum projeto.
11. Mostrar, para cada mês, a quantidade de alunos que entraram nos projetos.
12. Mostrar, para cada mês, a quantidade de alunos que saíram dos projetos.
13. Mostrar o nome dos projetos e a quantidade de alunos desistentes de cada um, ordenado decrescente pela quantidade.
14. Mostrar o nome do projeto e a média de tempo que os alunos ficam no projeto antes de desistir.
15. Mostrar a quantidade média de alunos por projeto.

16. Mostrar o nome do projeto e o percentual de alunos que saíram do projeto em relação aos alunos que ainda continuam no projeto.
17. Mostrar o nome do professor e a quantidade de alunos que cada professor orienta atualmente.
18. Mostrar o nome dos professores e quantidade de desistência de alunos, ordenado decrescente pela quantidade.
19. Mostrar o nome dos alunos que são orientados por professores com titulação “doutor”
20. Mostrar o e-mail dos alunos que participam de qualquer projeto.
21. Mostrar o nome dos alunos que participam apenas de um projeto.
22. Mostrar, para cada projeto, o valor médio do orçamento para cada aluno, ou seja, o valor do orçamento dividido pelo número de alunos de cada projeto.
23. Mostrar o nome dos professores que não possuem nenhum projeto.
24. Mostrar o nome dos alunos que não participam de nenhum projeto.
25. Mostrar o nome e orçamento dos 3 projetos que tem maior orçamento e dos 3 projetos que possuem menor orçamento.
26. Mostrar o nome dos alunos que estão no projeto desde o início.
27. Mostrar o nome do professor que possui a titulação igual à titulação do professor chamado “Roberto”.

2) Executar o script fornecido pelo professor e testar as consultas, verificando se os resultados correspondem com cada item solicitado. Caso necessário, crie alguns inserts para que os resultados apareçam adequadamente.

Montar um script para fazer o insert dos dados para testes. Sugestão: a turma monta junto um script para todos utilizarem.

Script DDL:

```
--Script do Sistema de Aluno-Projeto-Professor
DROP TABLE Equipe;
DROP TABLE Projeto;
DROP TABLE Professor;
DROP TABLE Aluno;
DROP TABLE Curso;

CREATE TABLE Curso (
    sigla INTEGER NOT NULL,
    nome VARCHAR(20) NOT NULL,
    CONSTRAINT curso_pk PRIMARY KEY (sigla)
);

CREATE TABLE Aluno (
    matricula INTEGER NOT NULL,
    nome VARCHAR(30) NOT NULL,
    email VARCHAR(30) NOT NULL,
    curso INTEGER NOT NULL,
    CONSTRAINT aluno_pk PRIMARY KEY (matricula)
);

CREATE TABLE Professor (
    matricula INTEGER NOT NULL,
```

```

        nome VARCHAR(30) NOT NULL,
        email VARCHAR(30) NOT NULL,
        titulacao VARCHAR(20) NOT NULL,
        CONSTRAINT professor_pk PRIMARY KEY (matricula)
    );

CREATE TABLE Projeto (
    inscricao INTEGER NOT NULL,
    titulo VARCHAR(50) NOT NULL,
    dataInicio DATE NOT NULL,
    dataFim DATE,
    orcamento NUMERIC NOT NULL,
    matProfessor INTEGER NOT NULL,
    CONSTRAINT projeto_pk PRIMARY KEY (inscricao)
);

CREATE TABLE Equipe (
    codProjeto INTEGER NOT NULL,
    matAluno INTEGER NOT NULL,
    dataInicioParticip DATE NOT NULL,
    dataFimParticip DATE,
    CONSTRAINT equipe_pk PRIMARY KEY (codProjeto, matAluno)
);

ALTER TABLE Aluno ADD CONSTRAINT curso_aluno_fk FOREIGN KEY (curso) REFERENCES Curso (sigla) ON DELETE NO ACTION
ON UPDATE NO ACTION;

ALTER TABLE Equipe ADD CONSTRAINT aluno_equipe_fk FOREIGN KEY (matAluno) REFERENCES Aluno (matricula) ON DELETE NO
ACTION ON UPDATE NO ACTION;

ALTER TABLE Projeto ADD CONSTRAINT professor_projeto_fk FOREIGN KEY (matProfessor) REFERENCES Professor
(matricula) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Equipe ADD CONSTRAINT projeto_equipe_fk FOREIGN KEY (codProjeto) REFERENCES Projeto (inscricao) ON
DELETE NO ACTION ON UPDATE NO ACTION;

```

Script populate:

```

INSERT INTO curso (sigla,nome) VALUES (1, 'Banco de dados');
INSERT INTO curso (sigla,nome) VALUES (2, 'Engenharia');
INSERT INTO curso (sigla,nome) VALUES (3, 'Administração');
INSERT INTO curso (sigla,nome) VALUES (4, 'Biologia');
INSERT INTO curso (sigla,nome) VALUES (5, 'Sistemas');

INSERT INTO aluno (matricula, nome, email, curso) VALUES (1, 'João', 'aluno1@gmail.com', 2);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (2, 'José', 'aluno2@gmail.com', 3);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (3, 'Jonas', 'aluno3@gmail.com', 4);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (4, 'Jane', 'aluno4@gmail.com', 1);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (5, 'Júlia', 'aluno5@gmail.com', 2);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (6, 'Joaquim', 'aluno6@gmail.com', 3);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (7, 'Jandira', 'aluno7@gmail.com', 4);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (8, 'Júlia', 'aluno8@gmail.com', 4);
INSERT INTO aluno (matricula, nome, email, curso) VALUES (9, 'Juliano', 'aluno9@gmail.com', 1);

INSERT INTO professor (matricula,nome,email,titulacao) VALUES (1, 'Roberto', 'prof1@gmail.com', 'mestre');
INSERT INTO professor (matricula,nome,email,titulacao) VALUES (2, 'Rogério', 'prof2@gmail.com', 'doutor');
INSERT INTO professor (matricula,nome,email,titulacao) VALUES (3, 'Ronan', 'prof3@gmail.com', 'especialista');
INSERT INTO professor (matricula,nome,email,titulacao) VALUES (4, 'Ricardo', 'prof4@gmail.com', 'mestre');
INSERT INTO professor (matricula,nome,email,titulacao) VALUES (5, 'Robson', 'prof5@gmail.com', 'especialista');
INSERT INTO professor (matricula,nome,email,titulacao) VALUES (6, 'Rui', 'prof6@gmail.com', 'mestre');

INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (1, 'proj1', '2020-01-01', '2020-02-11',
90000, 1);
INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (2, 'proj2', '2020-02-01', '2020-03-12',
80000, 1);
INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (3, 'proj3', '2020-03-01', '2020-04-13',
70000, 2);
INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (4, 'proj4', '2020-04-01', '2020-05-14',
60000, 3);
INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (5, 'proj5', '2020-05-01', '2020-06-15',
50000, 4);

```

```

INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (6, 'proj6', '2020-06-01', '2020-07-16',
40000, 5);
INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (7, 'proj7', '2020-07-01', null,
20000, 5); --o projeto ainda não terminou
INSERT INTO projeto (inscricao,titulo,dataInicio,dataFim,orcamento,matProfessor) VALUES (8, 'proj8', '2020-08-01', null,
10000, 3); --o projeto ainda não terminou

INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (1, 1, '2020-01-02', '2020-01-21');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (1, 2, '2020-01-02', '2020-01-22');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (1, 3, '2020-01-02', '2020-01-23');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (1, 4, '2020-01-02', '2020-01-24');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (2, 5, '2020-02-02', '2020-02-25');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (2, 6, '2020-02-02', '2020-02-26');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (2, 7, '2020-02-02', '2020-02-27');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (3, 1, '2020-03-02', '2020-03-21');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (3, 2, '2020-03-02', '2020-03-22');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (4, 3, '2020-04-02', '2020-04-23');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (5, 4, '2020-05-02', '2020-05-24');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (6, 5, '2020-06-02', '2020-06-25');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (6, 6, '2020-06-02', '2020-06-26');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (7, 7, '2020-07-02', '2020-07-27');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (7, 1, '2020-07-02', null); -- ainda participa
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (7, 2, '2020-07-02', null); -- ainda participa
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (8, 3, '2020-08-02', '2020-08-23');
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (8, 4, '2020-08-02', null); -- ainda participa
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (8, 5, '2020-08-02', null); -- ainda participa
INSERT INTO equipe (codProjeto,matAluno,dataInicioParticip,dataFimParticip) VALUES (8, 8, '2020-08-02', null); -- ainda participa

```