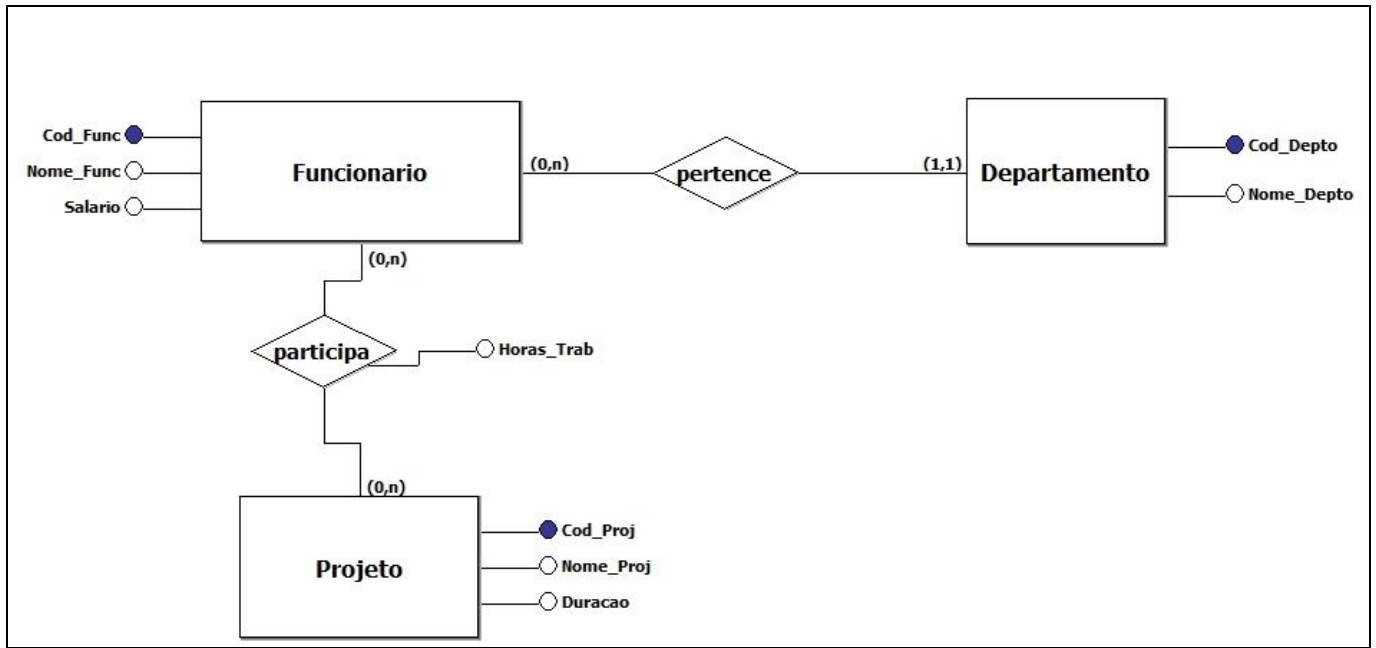


Exercícios da Aula 12, 13 e 14 – SELECT Básico, Funções Agregadas e GROUP BY/HAVING

Considere o seguinte Banco de Dados “Projeto” para responder as questões deste exercício, utilizando a Linguagem SQL (o script de criação deste banco de dados se encontra no final deste arquivo).

MER



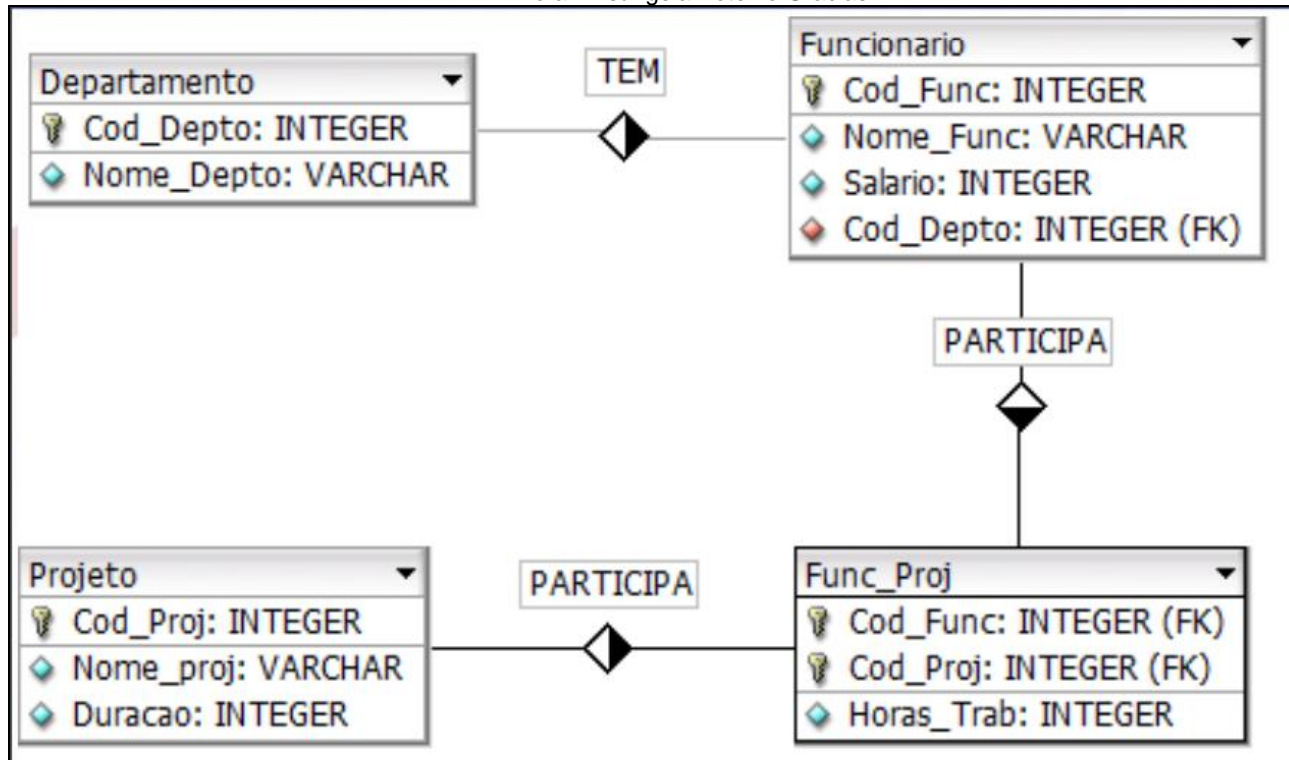
Modelo Relacional:

Departamento = {Cod_Depto, Nome_Depto}

Funcionário = {Cod_Func, Nome_Func, Salario, Cod_Depto} – Cod_depto é chave estrangeira que referencia o atributo Cod_depto da tabela Departamento.

Projeto = {Cod_Proj, Nome_Proj, Duracao}

Func_Proj = {Cod_Func, Cod_Proj, Horas_Trab} - Cod_Func é chave estrangeira que referencia o atributo Cod_Func da tabela Funcionário e Cod_Proj é chave estrangeira que referencia o atributo Cod_Proj da tabela Projeto.



Tabelas com Dados

Departamento

Cod_Depto	Nome_Depto
1	Marketing
2	Vendas
3	Dados
4	Pesquisa

Funcionario

Cod_Func	Nome_Func	Salario	Cod_Depto
101	Joao da Silva Santos	2000	2
102	Mario Souza	1500	1
103	Sergio Silva Santos	2400	2
104	Maria Castro	1200	1
105	Marcio Silva Santana	1400	4

Projeto

Cod_Proj	Nome_Proj	Duracao
1001	SistemaA	2
1002	SistemaB	6
1003	SistemaX	4

Func_Proj

Cod_Func	Cod_Proj	Horas_Trab
101	1001	24
101	1002	160
102	1001	56
102	1003	45
103	1001	86
103	1003	64
104	1001	46
105	1001	84
105	1002	86

1ª PARTE – SELECT BÁSICO

- 1) Obtenha o nome dos projetos que estão sem valor para a duração.
- 2) Obtenha o nome e salário dos funcionários que possuem um salário entre 1500 e 2400 (inclusive estes valores), em ordem decrescente de salário.
- 3) Obtenha o nome dos funcionários cujo nome não se inicie com a letra 'M' e cujo salário seja maior que 2000.
- 4) Obtenha o nome dos funcionários que possuem 'Silva' no nome, em ordem crescente do nome do funcionário.
- 5) Obtenha, sem repetição, os códigos dos funcionários que trabalharam no projeto de código 1001 mais de 50 horas ou que trabalharam no projeto de código 1003 mais de 60 horas.

2ª PARTE – SELECT COM FUNÇÕES AGREGADAS

- 1) Obtenha a média, o valor máximo e o valor mínimo dos salários pagos aos funcionários.
- 2) Obtenha a quantidade de funcionários que ganham um salário entre 2000 e 10000 (inclusive estes valores).
- 3) Obtenha a quantidade de funcionários cujo nome começa com a letra M.
- 4)
 - a) Obtenha a quantidade total de horas que foram utilizadas pelos funcionários no projeto de código 1001.
 - b) Obtenha a quantidade total de horas que foram utilizadas pelos funcionários no projeto de código 1002.
 - c) Obtenha a quantidade total de horas que foram utilizadas pelos funcionários no projeto de código 1003.
 - d) E se no banco de dados tivesse 1000 projetos diferentes, o que você faria para retornar a quantidade total de horas utilizadas em cada um dos 1000 projetos?
- 5) Obtenha o total que deverá ser pago ao funcionário de código 102, de acordo com o total de horas que ele trabalhou nos projetos, considerando que o valor de cada hora trabalhada deste funcionário é 100 reais.

3ª PARTE – SELECT COM GROUP BY e HAVING

- 1) Obtenha, para cada código de cada departamento, o total gasto com salário pago aos funcionários, bem como a média salarial de cada departamento.
- 2) Obtenha somente o código dos departamentos que possuem uma quantidade de funcionários entre 2 e 10 funcionários.
- 3) Obtenha, para cada funcionário (considere somente o código do funcionário), a quantidade total de horas trabalhadas nos projetos como um todo.
- 4) Obtenha o código de cada projeto e a quantidade total de horas trabalhadas pelos funcionários em cada um deles, em ordem decrescente do total de horas, mas desde que o total de horas trabalhadas no projeto seja maior que 200.
- 5) Obtenha, para cada projeto (considere somente o código do projeto), a quantidade de funcionários que trabalharam em cada um deles, mas desde que o número de funcionários por projeto seja superior a três funcionários.

Universidade Presbiteriana Mackenzie
Banco de Dados
Profa. Elisângela Botelho Gracias

-- SCRIPT COMPLETO DO BANCO

-- eliminação das tabelas

```
DROP TABLE Func_Proj;
DROP TABLE Projeto;
DROP TABLE Funcionario;
DROP TABLE Departamento;
```

-- criação das tabelas

```
CREATE TABLE Departamento
(Cod_Depto INTEGER,
 Nome_Depto VARCHAR(20) NOT NULL,
 PRIMARY KEY(Cod_Depto)
);
```

```
CREATE TABLE Funcionario
(Cod_Func INTEGER,
 Nome_Func VARCHAR(20) NOT NULL,
 Salario INTEGER,
 Cod_Depto INTEGER,
 PRIMARY KEY(Cod_Func),
 FOREIGN KEY (Cod_Depto) REFERENCES Departamento (Cod_Depto)
);
```

```
CREATE TABLE Projeto
(Cod_Proj INTEGER,
 Nome_Proj VARCHAR(20) NOT NULL,
 Duracao INTEGER,
 PRIMARY KEY(Cod_Proj)
);
```

```
CREATE TABLE Func_Proj
(Cod_Func INTEGER,
 Cod_Proj INTEGER,
 Horas_Trab INTEGER,
 PRIMARY KEY(Cod_Func, Cod_Proj),
 FOREIGN KEY (Cod_Func) REFERENCES Funcionario(Cod_Func),
 FOREIGN KEY (Cod_Proj) REFERENCES Projeto(Cod_Proj)
);
```

-- Inserção de dados na tabela Departamento

```
INSERT
INTO Departamento (Cod_Depto, Nome_Depto)
VALUES (1, 'Marketing');
```

```
INSERT
INTO Departamento (Cod_Depto, Nome_Depto)
VALUES (2, 'Vendas');
```

```
INSERT
INTO Departamento (Cod_Depto, Nome_Depto)
VALUES (3, 'Dados');
```

```
INSERT
INTO Departamento (Cod_Depto, Nome_Depto)
VALUES (4, 'Pesquisa');
```

-- Inserção de dados na tabela Funcionario

```
INSERT
INTO Funcionario (Cod_Func, Nome_Func, Salario, Cod_Depto)
VALUES (101, 'Joao da Silva Santos', 2000, 2);
```

Universidade Presbiteriana Mackenzie
Banco de Dados
Profa. Elisângela Botelho Gracias

```
INSERT
INTO Funcionario (Cod_Func, Nome_Func, Salario, Cod_Depto)
VALUES (102, 'Mario Souza', 1500, 1);
```

```
INSERT
INTO Funcionario (Cod_Func, Nome_Func, Salario, Cod_Depto)
VALUES (103, 'Sergio Silva Santos', 2400, 2);
```

```
INSERT
INTO Funcionario (Cod_Func, Nome_Func, Salario, Cod_Depto)
VALUES (104, 'Maria Castro', 1200, 1);
```

```
INSERT
INTO Funcionario (Cod_Func, Nome_Func, Salario, Cod_Depto)
VALUES (105, 'Marcio Silva Santana', 1400, 4);
```

-- Inserção de dados na tabela Projeto

```
INSERT
INTO Projeto (Cod_Proj, Nome_Proj, Duracao)
VALUES (1001, 'SistemaA', 2);
```

```
INSERT
INTO Projeto (Cod_Proj, Nome_Proj, Duracao)
VALUES (1002, 'SistemaB', 6);
```

```
INSERT
INTO Projeto (Cod_Proj, Nome_Proj, Duracao)
VALUES (1003, 'SistemaX', 4);
```

-- Inserção de dados na tabela Func_Proj

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (101, 1001, 24);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (101, 1002, 160);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (102, 1001, 56);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (102, 1003, 45);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (103, 1001, 86);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (103, 1003, 64);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (104, 1001, 46);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (105, 1001, 84);
```

```
INSERT
INTO Func_Proj (Cod_Func, Cod_Proj, Horas_Trab)
VALUES (105, 1002, 86);
```