

Seleção de Dados

Professor: Henrique Delegrego

Seleção de Dados

SELECT

- Utilizado para recuperar dados de uma ou mais tabelas
- Principal comando da linguagem
- Estatísticas mostram que mais de 70% das operações em um SGBD relacional correspondem a consultas aos dados
- Seleciona, restringe, projeta e agrupa dados de uma ou mais tabelas
- Gera diferentes visões sobre os mesmos dados

Seleção de Dados

Comandos

- O comando **SELECT** pode apresentar até seis cláusulas, sendo que duas delas são obrigatórias (**SELECT E FROM**)

FROM: qual tabela(s) deve ser recuperada

ORDER BY:
ordenamento dos dados

```
SELECT  
{ *, coluna [apelido], coluna [apelido],... }
```

```
FROM  
{ tabela [apelido], tabela [apelido],... }
```

```
WHERE  
{ condição, condição,... }
```

```
GROUP BY  
{ coluna_agrupamento, coluna_agrupamento,... }
```

```
HAVING  
{ condição_função_agrupamento, condição_função_agrupamento,... }
```

```
ORDER BY  
{ coluna, coluna,... }
```

SELECT: informar quais colunas deseja recuperar

WHERE: informar as condições que os dados a serem recuperados devem atender

Seleção de Dados

SELECT

```
SELECT coluna1, coluna2, ...  
FROM tabela;
```

- Para selecionar todas as colunas de uma tabela, use **SELECT ***

```
SELECT *  
FROM tabela;
```

cd_departamento	nm_departamento	localizacao
1	Financeiro	Blumenau
2	Desenvolvimento	Florianopolis
3	Financeiro	Florianopolis
4	Desenvolvimento	Blumenau
5	RH	Curitiba

Seleção de Dados

SELECT

- Para selecionar somente algumas colunas

```
SELECT nm_departamento, localizacao (Colunas)  
FROM departamento; (Tabela)
```

nm_departamento	localizacao
Financeiro	Blumenau
Desenvolvimento	Florianopolis
Financeiro	Florianopolis
Desenvolvimento	Blumenau
RH	Curitiba

Agora:

- Vamos ver como é a seleção de dados
- Pratique o comando **SELECT** nas tabelas que você tem

Seleção de Dados

WHERE

- A cláusula **WHERE** é usada para filtrar os registros de uma tabela com base em uma condição
- Serve para dizer ao banco de dados quais linhas devem ser retornadas de acordo com critérios definidos

```
SELECT coluna1, coluna2, ...  
FROM tabela  
WHERE condição;
```

```
SELECT nm_departamento, localizacao  
FROM departamento  
WHERE localizacao = "Blumenau";
```

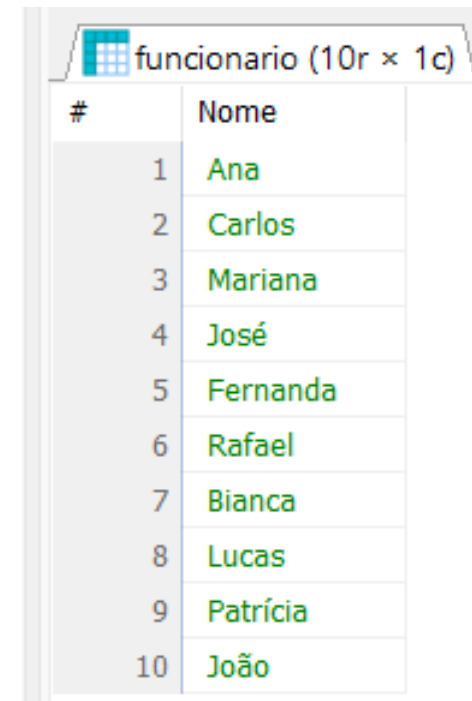
nm_departamento	localizacao
Financeiro	Blumenau
Desenvolvimento	Blumenau

Seleção de Dados

Apelido (Alias)

- Nome temporário que você pode atribuir a uma coluna ou tabela em um **SELECT**
- Usar aspas no apelido quando:
 - O apelido tiver mais de 1 palavra
 - Usa palavras reservadas do SGBD

```
SELECT nm_funcionario AS nome
FROM funcionario;
-- O comando "AS" é opcional
```



#	Nome
1	Ana
2	Carlos
3	Mariana
4	José
5	Fernanda
6	Rafael
7	Bianca
8	Lucas
9	Patrícia
10	João

Seleção de Dados

ORDER BY

- Para ordenar em ordem decrescente

```
SELECT nm_pessoa, salario
FROM pessoa
ORDER BY salario DESC;
```

- Para ordenar em ordem crescente
(padrão, podendo ser omitida)

```
SELECT nm_pessoa, salario
FROM pessoa
ORDER BY salario ASC;
```

nm_pessoa	salario
Ana	5,200
Eduardo	4,900
Juliana	4,700
Maria	4,500
Fernanda	3,900
Bruno	3,600
Camila	3,300
João	3,000
Carlos	2,800
Marcos	2,000

Agora:

- Faça a lista de exercícios

Seleção de Dados

Qualificação de colunas

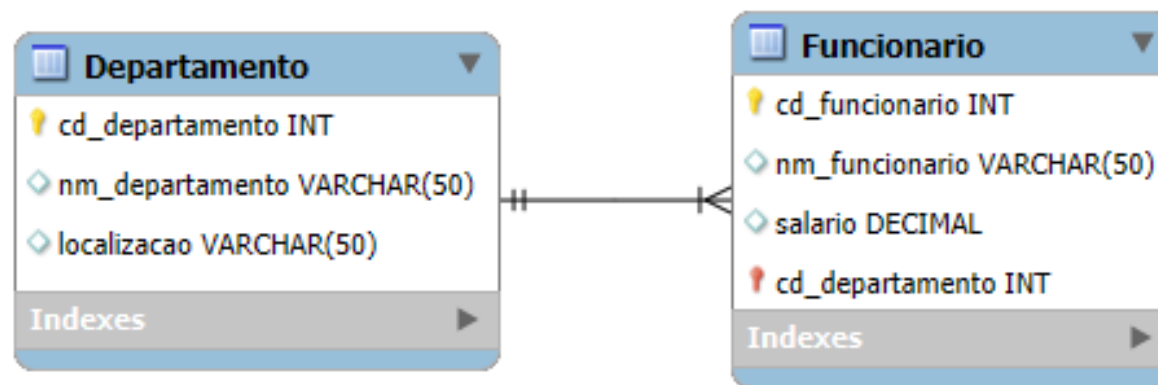
- Mencionar o nome de uma coluna com o nome da tabela em uma consulta SQL
- Faz o SGBD entender exatamente de onde vem cada coluna
- Obrigatório em seleção de mais de uma tabela. Se mais de uma tabela tiver uma coluna com o mesmo nome, a qualificação resolve o conflito

```
SELECT funcionario.nm_funcionario, funcionario.salario  
FROM funcionario;
```

Seleção de Dados

JOIN

- Usado para combinar linhas de 2 ou mais tabelas baseadas em uma coluna relacionada entre eles (normalmente atributos chaves)
- Por exemplo, se você possui uma tabela **Departamento** e uma tabela **Funcionario**, pode realizar um **join** entre elas usando a coluna **cd_departamento** para exibir os pedidos de cada funcionário juntamente com o nome do departamento que ele trabalha



Seleção de Dados

INNER JOIN

- Só retorna as linhas que possuem correspondência nas duas tabelas
- Se não houver correspondência, nada será exibido
- O comando **ON** define a condição de junção das tabelas

```
SELECT colunas
FROM tabela1
INNER JOIN tabela2
ON tabela1.coluna = tabela2.coluna;
-- O comando INNER é opcional
```

```
SELECT funcionario.nm_funcionario, departamento.nm_departamento
FROM funcionario
JOIN departamento
ON funcionario.cd_departamento = departamento.cd_departamento;
```



nm_funcionario	nm_departamento
Ana	Recursos Humanos
Carlos	Tecnologia da Informação
Mariana	Financeiro
José	Recursos Humanos
Fernanda	Tecnologia da Informação
Rafael	Marketing
Bianca	Vendas
Lucas	Marketing
Patrícia	Financeiro
João	Vendas

Seleção de Dados

Múltiplos INNER JOIN

- Usando o exemplo do exercício 1 de seleção:
 - Para retornar os alunos que são do professor “Henrique” da área de “Desenvolvimento de Software”

```
SELECT aluno.nm_aluno
FROM aluno
JOIN professor
ON aluno.cd_professor = professor.cd_professor
JOIN area
ON professor.cd_area = area.cd_area
WHERE professor.nm_professor = 'Henrique'
AND
area.nm_area = 'Desenvolvimento de Software';
```

