



Agenda

1. Funções de Agrupamento

2. Funções de Agregações

3. Having

4. Junções

5. Join



Select / Order By

- **Classifica o resultado da consulta (em ordem ascendente, caso nada seja especificado).**

- ◆ **Itens especificados no ORDER BY não precisam aparecer na lista da seleção**

- ◆ **Com ORDER BY nulos são listados primeiros.**

- **Sintaxe**

```
select [distinct] lista from tabela  
[where condicao]  
[order by {coluna | expressao} [asc|desc]]
```

- **Exemplo**

- ◆ **Select matricula, nome from empregado order by nome**

- ◆ **Select nome, salario from empregado order by salario desc**



Funções de Agregações

- Funções comuns aplicadas a coleções de valores numérico, incluem SUM(soma), AVG(média), MAX (máximo) e MIN (mínimo);
- Função COUNT é utilizada para contagem de tuplas
- Função SUM e AVG só funcionam com valores numéricos;
- Não podem ser utilizadas num comando WHERE;
- Podem ser aplicados a todas as linhas em uma tabela ou num grupo de linhas de uma tabela.



Funções de agregações

- Utilizadas no SELECT e na cláusula HAVING;
- Funções de agregação, com exceção do COUNT(*), ignoram valores nulos;
- SUM e AVG só funcionam com valores numéricos;
- Apenas uma linha retorna (sem GROUP BY);



Funções de Agregações

■ Sintaxe

```
select funcao_agregada ([distinct] expressao)  
from tabela [where condicoes]
```

■ Cont

- ◆ Soma o número de linhas resultante de uma condição estabelecida.
- ◆ Determinar o número de linhas da tabela
 - `select count(*) from empregado`
- ◆ Número de valores não nulos de um atributo
 - `select count(atributo) from empregado`



Funções de Agregações

■ Max

◆ Determinar o maior valor

- `select max(salario) from empregado`

■ Min

◆ Determina o menor valor

- `select min(salario) from empregado`



Funções de Agregações

■ SUM

- ◆ Determinar o somatório dos valores da coluna especificada

- `select sum(salario) from empregado where coddep = 4`

■ AVG

- ◆ Determina a média dos valores da coluna especificada

- `select avg(salario) from empregado where funcao = 'analista'`



Função de Agregação

■ DISTINCT

- ◆ Opcional com sum, avg e count;
- ◆ Não permitida com min, max e count (*);
- ◆ Permitida com count (atributo);
- ◆ Usado apenas com atributos, não com expressões aritméticas

■ Exemplo

- ◆ Média dos salarios distintos
 - `select avg(distinct salario) from empregado`
- ◆ Conta as funções distintas
 - `Select count (distinct funcao) from empregado`



Função de Agrupamento

- Divide os dados em grupos;
- Normalmente utilizada com uma função de agregação na lista de seleção;
- Os valores nulos na coluna do group by são tratados como um grupo;

- Sintaxe

- ◆ select [distinct] lista from tabela group by expressao

- Exemplo

- ◆ Select codproj, avg(horas) from Trabalhaem group by codproj



Having

- Estabelece condições para a sentença GROUP BY.

- Sintaxe

select [distinct] lista from tabela
group by expressao
having condicoes

- Exemplo

- ◆ Select codproj, avg(horas) from Trabalhaem group by
codproj having avg(horas) > 30



Exercício

1. Qual é a média de salários dos empregados:
 - a) por departamento?
 - b) por funcao?
2. Qual é o salario mais baixo e mais alto de cada departamento?
3. Quantas horas cada empregado trabalhou em projetos ?
4. Para os empregado que trabalharam menos de 10 horas por projeto, calcule a média de horas trabalhadas em projetos destes funcionários.
5. Quantos empregados existem por funcao ?
6. Qual o número de empregados por departamento?
7. Quantas horas foram gastas por projeto?



Exercício

8. Quantas horas cada empregado trabalhou em projetos?
9. Quantos salarios distintos existem na tabela de empregados?
10. Por codigo do projeto recupere quantos funcionários trabalharam nele e qual a valor minimo e máximo do número de horas trabalhadas em cada projeto.
11. Sem utilizar AVG calcule a média de salário dos empregados por departamento.
12. Listar o codigo do departamento e nome de todos os empregados classificados por codigo do departamento e nome.
13. Qual o número de dependentes filhos de sexo feminino?
14. Selecione as matriculas dos empregados que trabalham em mais de 2 projetos.
15. Selecione os projetos onde trabalham mais de 3 empregados.



Junções

- Pode-se utilizar tabelas do mesmo banco ou de banco diferentes;
- Atributos comparados devem ter valores similares;
- Valores nulos não participam da operação de junção;
- Os atributos na condição de junção não precisam estar na lista de seleção;
- Recupera dados de duas ou mais tabelas;
- Combina tabelas através da correspondência de valores de linhas em cada tabela;
- Qualificar os atributos que têm o mesmo nome, usando o nome do atributo precedido pelo nome da tabela;



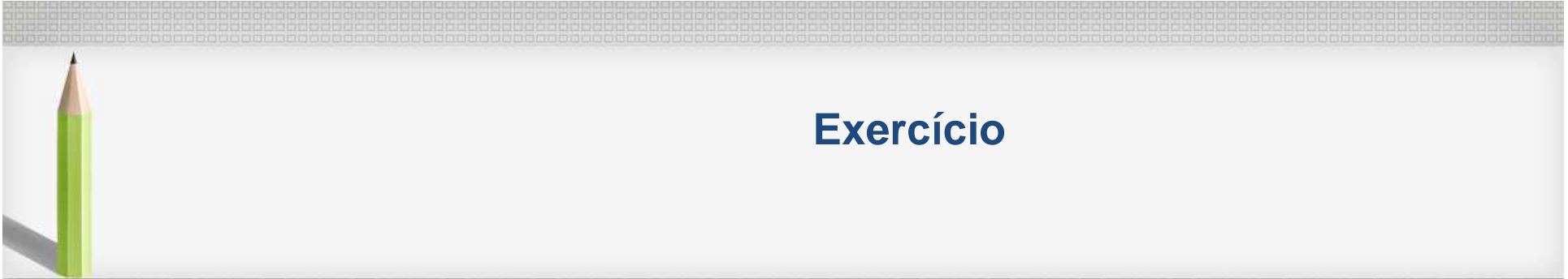
Junções

■ Idênticas

- ◆ Os atributos de junção são do mesmo tipo e fazem parte de um relacionamento entre as tabelas. (integridade referencial, chaves primária e estrangeira).

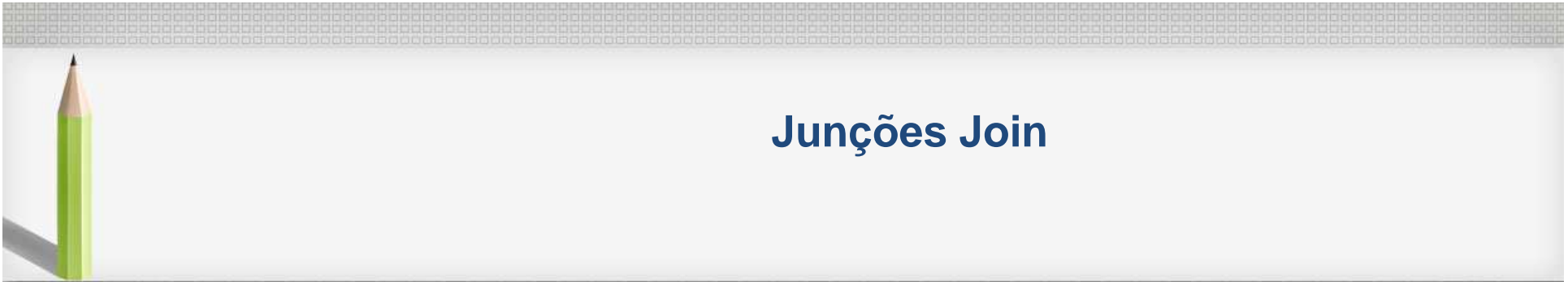
■ Não idênticas

- ◆ Os atributos de junção podem não ser do mesmo tipo e nem fazer parte de um relacionamento entre as tabelas.
- ◆ Muito utilizada para associar o resultado de uma função de agregação com outras tabelas.



Exercício

- 16. Mostre nome do departamento e todos os seus empregados classificado por nome do departamento e nome empregado;**
- 17. Mostre o nome e salario de cada empregado que trabalha no projeto Gestão do Conhecimento.**
- 18. Recupere os nome departamento, nome empregado e nome do projeto ordenados por departamento e nome;**



Junções Join

■ Cross Join

- ◆ Produto cartesiano

■ Join – Junção Natural

- ◆ Junta tabelas sem precisar declarar a condição seleção. Para isso as relações devem possuir atributos com mesmo nome. Será executada uma igualdade para cada par de atributos com o mesmo nome.

■ Inner Join – Junção Interna

- ◆ Padrão para join. Junta tabelas a partir de uma condição especificada. Uma tupla será incluída no resultado somente se existir uma tupla que combine na outra relação.



Junções Join

- **Outer Join – Junção Externa**
 - ◆ Junta tabelas a partir de uma condição especificada. Uma tupla será incluída no resultado se existir uma tupla que combine na outra relação, caso não combine, a tupla será incluída para a relação sinalizada (esquerda ou direita) e os atributos da outra relação serão preenchidos com nulo.
- **LEFT OUTER JOIN**
 - ◆ Junção externa à direita
- **RIGHT OUTER JOIN**
 - ◆ Junção externa à esquerda
- **FULL OUTER JOIN**
 - ◆ Junção externa completa



Exemplo Join

■ Produto cartesiano

```
SELECT * FROM departamento, empregado;
```

pode ser :

```
SELECT * FROM departamento CROSS JOIN empregado;
```

■ Junções interna – Expressão INNER JOIN

◆ Selecione o nome e função dos empregados que trabalham no departamento de Pesquisa.

```
SELECT empregado. nome, empregado.funcao FROM  
(empregado INNER JOIN departamento ON  
empregado.codepart = departamento.cod )  
WHERE departamento.nome = 'Pesquisa';
```

semelhante a :

```
SELECT empregado. nome, empregado.funcao FROM  
empregado, departamento WHERE empregado.codepart =  
departamento.cod )  
AND departamento.nome = 'Pesquisa';
```



Exemplo Join

■ Junções interna – Expressão INNER JOIN

◆ Para os projetos concluídos, liste o número do projeto, departamento controlador e o nome do gerente do departamento.

```
SELECT projeto.cod, departamento.nome, empregado.  
nome  
FROM ((projeto JOIN departamento ON projeto.coddepar =  
departamento.cod )  
JOIN empregado on  
departamento.matgerente=empregado.mat)  
WHERE projeto.situacao='Concluido';
```

semelhante a :

```
SELECT projeto.cod, departamento.nome, empregado.  
nome  
FROM projeto, departamento, empregado  
WHERE projeto.coddepar = departamento.cod AND  
departamento.matgerente=empregado.mat AND  
projeto.situacao='Concluido';
```