

SCC-240/540/640 Bases de Dados

Prof. Robson L. F. Cordeiro

Linguagem SQL - DML

Material original editado: Profa. Elaine Parros Machado de Sousa

DML - Introdução

- **Comandos da DML:**
 - **INSERT**
 - **UPDATE**
 - **DELETE**
 - **SELECT**

Comandos DML

- **INSERT** – insere uma ou mais tuplas em uma tabela

- Inserção de 1 tupla:

```
INSERT INTO tabela [(atrib1,atrib2,...)]  
VALUES (valor1, valor2,...)
```

- Inserção de múltiplas tuplas:

```
INSERT INTO tabela [(atrib1,atrib2,...)]  
<comando SELECT>
```

Exercício

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

- Inserir os seguintes dados:
 - aluna de nome Juliana, nro usp 222, nascida em 10 de abril de 1989, com cidade de origem *default*
 - disciplina SCC240, Banco de Dados, com 4 créditos.
 - matrícula da Juliana na disciplina SCC240, turma 1
- Criar uma tabela para os alunos menores de idade e alimentar com os alunos da tabela **Aluno** que têm menos de 18 anos.

Comandos DML

- **UPDATE** – modifica o valor de um ou mais atributos em uma ou mais tuplas da tabela

UPDATE *tabela* SET

atributo1 = <valor ou expressão> ,

atributo2 = <valor ou expressão> ,

...

WHERE <condição de localização>

Comandos DML

- **DELETE** – remove uma ou mais tuplas da tabela

DELETE FROM *tabela*
[WHERE <condição de localização>]

Exercícios

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

- Atualizar os seguintes dados:
 - alterar para 70% a frequência de todos os alunos com nota acima de 5.0 e frequência abaixo de 70%
 - acrescentar um crédito para as disciplinas do departamento de Ciências de Computação (SCC)
- Remover os seguintes dados
 - matrícula dos alunos da turma 2 de SCC240
 - disciplinas com número de créditos superior a 6

Comandos DML

- **SELECT** – comando de consulta
 - retorno \Rightarrow tabela resultado (**multiconjunto**)

```
SELECT [DISTINCT|ALL] <lista de atributos>
FROM <lista de tabelas>
[WHERE <condições>]
[GROUP BY atributo]
[HAVING <condições>]
[ORDER BY atributo [ASC|DESC]]
```


SELECT

- **SELECT** → O QUE se deseja na tabela resultado
 - *<lista de atributos>* ou
 - * (para todos os atributos)
 - **ALL** – resultado pode conter tuplas duplicadas (*default*)
 - **DISTINCT** – resultado contém somente tuplas distintas
- **FROM** → DE ONDE retirar os dados necessários
- **WHERE** → CONDIÇÕES de consulta
 - expressão condicional booleana
 - condições de seleção
 - condições de junção

Exercícios

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

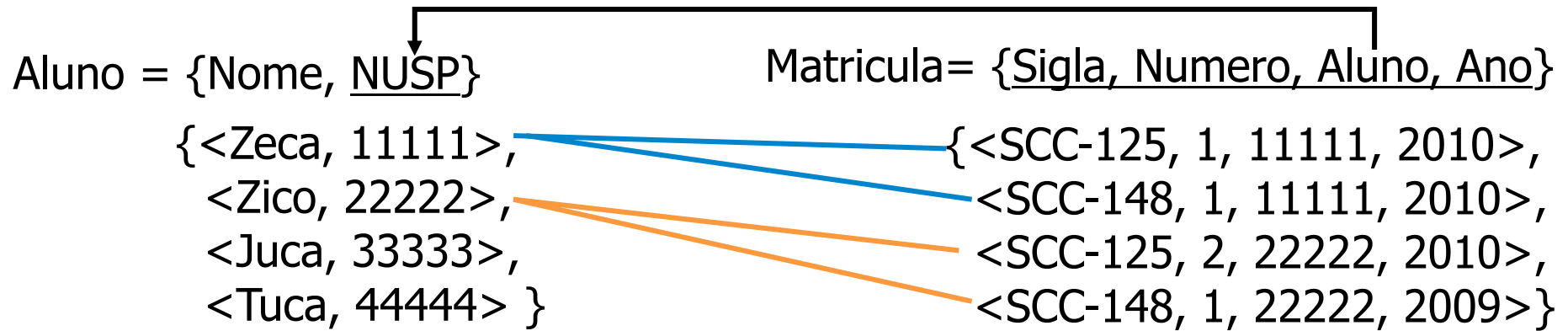
- 1) Selecionar nome, nusp e data de nascimento de todos os alunos que vieram de São Paulo.

SELECT

- Cláusula **FROM** com mais de uma tabela
 - **Junção Interna** (Inner Join)
 - **WHERE** \Rightarrow condição de junção

```
SELECT [DISTINCT|ALL] <atributos>
FROM tabela1, tabela2
WHERE tabela1.atributo1 =
      tabela2.atributo3
```

Exemplo: Junção



```
select A.nome, A.nusp, M.Sigla
from Aluno A, Matricula M
where A.nusp = M.aluno
```

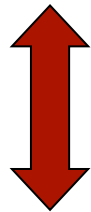
{Nome, NUSP, Sigla}
{<Zeca, 11111, SCC-125>, <Zeca, 11111, SCC-148>, <Zico, 22222, SCC-125>, <Zico, 22222, SCC-148 >}

Junção Interna – operador JOIN

```
SELECT [DISTINCT|ALL] <atributos>  
FROM tabela1 T1  
[INNER] JOIN tabela2 T2  
ON T1.atributo1 =  
   T2.atributo2
```

Junção Interna

```
SELECT <atributos>  
    FROM tabela1 T1 , tabela2 T2  
    WHERE T1.atributo1 =  
        T2.atributo2
```



```
SELECT <atributos>  
    FROM tabela1 T1 JOIN tabela2 T2  
    ON T1.atributo1 = T2.atributo2
```

Exercícios

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

1) ...

2) Selecionar, para todos os alunos matriculados em disciplinas do SCC, nome e nusp, nome e sigla das disciplinas em que estão matriculados

SELECT

■ Funções Agregadas

- entrada \Rightarrow conjunto de valores
- saída \Rightarrow valor
- Exemplos:
 - **AVG(*atributo*)** \rightarrow calcula a média da coluna ***atributo***
 - **COUNT()**
 - **count(*)** – retorna o número de tuplas de uma consulta
 - **count(*atributo*)** – retorna o número de valores da coluna ***atributo***

SELECT

- Funções Agregadas

- Exemplos

- **MAX**(*atributo*) → recupera o valor máximo da coluna *atributo*

- **MIN**(*atributo*) → recupera o valor mínimo da coluna *atributo*

- **SUM**(*atributo*) → obtém a soma de valores da coluna *atributo*

- ...

Exercícios

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

1)

2)

3) Selecionar a quantidade de matrículas nas disciplinas Banco de Dados ou Base de Dados

3.1) Selecionar a quantidade de alunos distintos que cursam ou já cursaram disciplinas Banco de Dados ou Base de Dados

SELECT

- **GROUP BY** → agrupamento de tuplas
 - para a aplicação de funções agregadas
 - **HAVING** → condições aplicadas **a grupos já formados** por **GROUP BY**
- **ORDER BY** → estabelece a ordenação lógica da tabela de resultados
 - **ASC** (*default*)
 - **DESC**

Exercícios

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

1) ...

2) ...

3) ...

- 4) Selecionar nome e nusp dos alunos, nome e sigla das disciplinas, e número da turma para todos os alunos matriculados em disciplinas do SCC. Ordenar o resultado por nusp do aluno e por sigla.

Exercícios

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

- 5) Selecionar, para cada aluno, seu nome, nusp e a média das notas das disciplinas em que foi aprovado (nota ≥ 5). Ordenar por nome de aluno

Consulta 5):

Aluno = {Nome, NUSP} Matricula = {Sigla, Numero, Aluno, Ano, Nota}

{<Zeca, 11111>, {<SCC-125, 1, 11111, 2010, 5.0>,
 <Zico, 22222>,
 <Juca, 33333>,
 <Tuca, 44444> } <SCC-148, 1, 11111, 2010, 7.0>,
 <SCC-125, 2, 22222, 2010, 5.0>,
 <SCC-148, 1, 22222, 2009, 4.0>}

- Selecionar, para cada aluno, seu nome e a média das notas das disciplinas em que foi aprovado (nota >= 5). Ordenar por nome de aluno

1º Passo: seleção e junção

```
SELECT ...  
  FROM Aluno A JOIN Matricula M  
           ON M.Aluno = A.NUSP  
 WHERE M.Nota BETWEEN 5.0 AND 10.0
```

{Nusp, Nome, Sigla, Nota}
{<11111, Zeca, SCC-125, 5.0>,
 <11111, Zeca, SCC-148, 7.0>,
 <22222, Zico, SCC-125, 5.0>}

(continuação)

2º Passo: agrupamento e agregação

```
SELECT A.Nome, AVG(M.Nota) as Media
FROM Aluno A JOIN Matricula M
      ON M.Aluno = A.NUSP
WHERE M.Nota BETWEEN 5.0 AND 10.0
GROUP BY A.Nome
ORDER BY A.Nome;
```

Grupo Zeca

<SCC125, 5.0>
<SCC148, 7.0>

Grupo Zico

<SCC125, 5.0>

Função **AVG** aplicada sobre cada grupo



{Nome, Media}
{<Zeca, 6.0>,
<Zico, 5.0>}

Exercícios

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

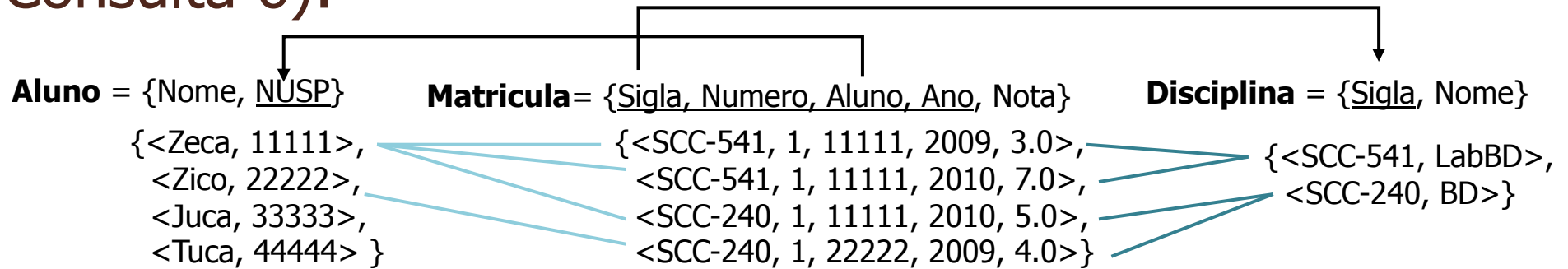
Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota, Frequencia}

- 6) Selecionar os nomes dos alunos que fizeram uma mesma disciplina mais de uma vez. Listar também o nome da disciplina, o número de vezes que cursou e a nota máxima que o aluno obteve (considerando todas as vezes que cursou).

Consulta 6):



- Selecionar os nomes dos alunos que fizeram uma mesma disciplina mais de uma vez.
Listar também o nome da disciplina, o nro de vezes que cursou e a nota máxima que o aluno obteve (considerando todas as vezes que cursou).

1º Passo: junção

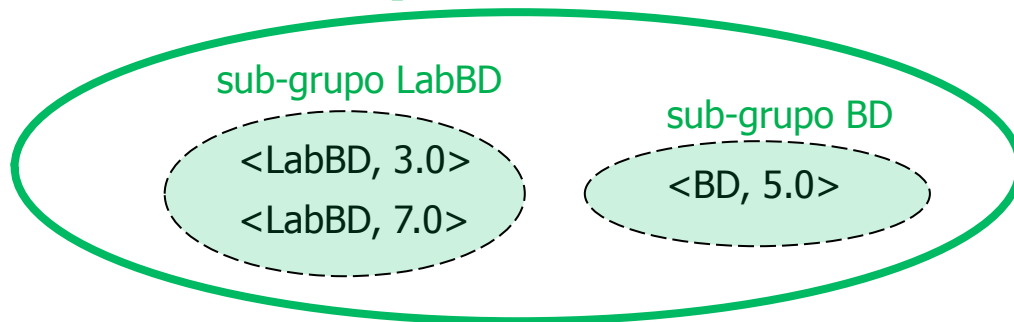
```
select ....  
  from Aluno A join Matricula M  
           on A.NUSP = M.Aluno  
  join Disciplina D  
  on D.Sigla = M.Sigla
```

(continuação)

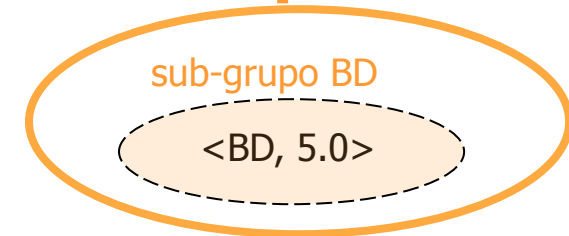
2º Passo: agrupamento e agregação

```
select A.Nome, D.Nome, count(*), max(M.Nota)
  from Aluno A join Matricula M
            on A.NUSP = M.Aluno
  join Disciplina D
    on D.Sigla = M.Sigla
 group by A.Nome, D.Nome
```

Grupo Zeca



Grupo Zico

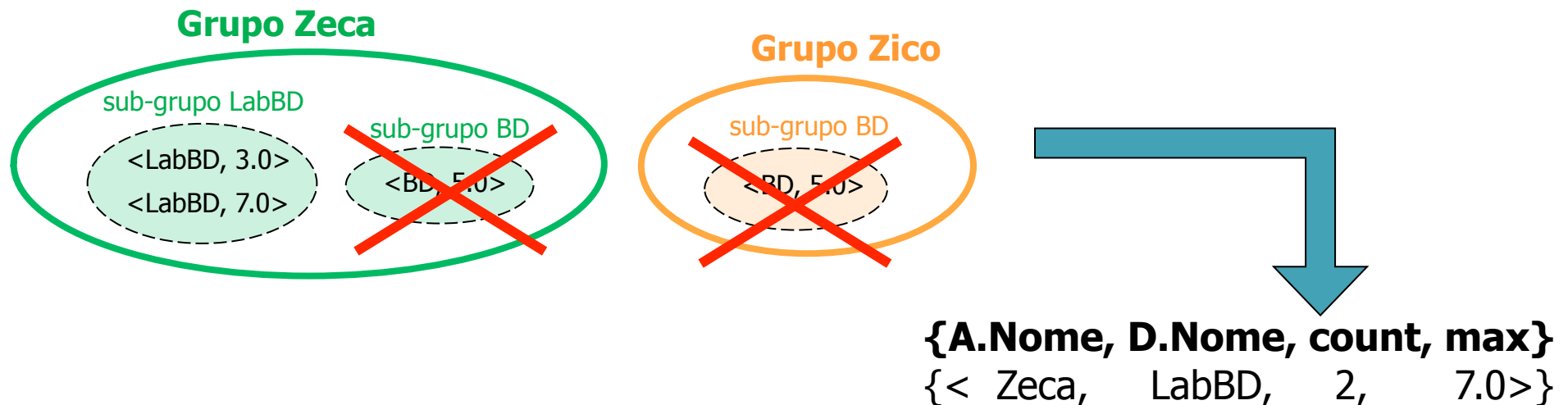


Funções **COUNT** e **MAX** aplicadas sobre cada sub-grupo

(continuação)

3º Passo: condição having

```
select A.Nome, D.Nome, count(*), max(M.Nota)
  from Aluno A join Matricula M
            on A.NUSP = M.Aluno
      join Disciplina D
            on D.Sigla = M.Sigla
  group by A.Nome, D.Nome
  having count(*) > 1
```



Leitura recomendada

- R. Elmasri, S. Navathe: *Fundamentals of Database Systems* – 4th Edition
 - Capítulo 8
- A. Silberschatz, H. F. Korth, s. Sudarshan: *Sistema de Banco de Dados*
 - Capítulo 4