

SQL DML

Modificação de Bancos de Dados



INSERT DELETE UPDATE

Prof. Dr. Luis Mariano del Val Cura

Modificação de bancos de dados

- Um comando de *modificação* não retorna um resultado (como uma consulta) mas modifica o Banco de Dados de alguma forma.
- Três tipos de modificação:
 1. *Insert* Insere uma tupla ou tuplas.
 2. *Delete* Remoção de uma tupla ou tuplas.
 3. *Update* Atualiza o valor(es) de uma tupla ou tuplas existentes.

Inserção

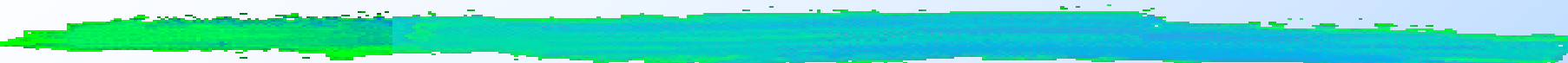
- Para inserir uma tupla simples:

```
INSERT INTO <relação>  
VALUES ( <lista de valores> );
```

- Exemplo: adicionar a Gostam(pessoa, cerveja) o fato que João gosta de Skoll Bud.

```
INSERT INTO Gostam  
VALUES ( 'João', 'Skoll' );
```

Especificação de atributos no INSERT



- Podemos adicionar ao nome da relação, uma lista de atributos.
- Duas razões:
 1. Esquecemos a ordem dos atributos na relação.
 2. Não temos valores para todos os atributos e desejamos que o sistema preencha com NULL os componentes que faltem.

Exemplo: Especificação de atributos

- Outra forma de adicionar que João gosta de Skoll em `Gostam(pessoa, cerveja)`

```
INSERT INTO  
  Gostam(cerveja,pessoa)  
VALUES ('Skoll', 'João');
```

Para inserir múltiplas tuplas.

- Podemos inserir em uma relação o resultado completo de uma consulta com a forma:

```
INSERT INTO <relação>  
( <sub-consulta> );
```

Exemplo: Inserção de uma sub-consulta

- Usando `Frequenta(pessoa, bar)` inserir em uma nova relação `Parceiros(nome)` todos os potenciais parceiros de João, i.e., aquelas pessoas que freqüentam pelo menos um dos bares que João freqüenta.

Solução

A outra
pessoa

Pares de tuplas de
Pessoa. O
primeiro é João e
o segundo é
qualquer um que
compartilhe o
mesmo Bar

INSERT INTO Parceiros

(SELECT d2.pessoa

FROM Frequenta d1, Frequenta d2
WHERE d1.pessoa = 'João' AND
d2.pessoa <> 'João' AND
d1.bar = d2.bar

);

Remoção

- Para remover tuplas que satisfazem uma condição da mesma relação:

```
DELETE FROM <relação>  
WHERE <condição>;
```

Exemplo: Remoção

- Eliminar de Gostam(pessoa, cerveja) o fato que João gosta de Soll:

DELETE FROM Gostam

WHERE pessoa = 'João' AND
cerveja = 'Skoll';

Exemplo: Remover todas as tuplas

- Esvaziar a relação **Gostam**:

DELETE FROM Gostam;

- Note que não precisa cláusula WHERE.

Exemplo: Remoção de várias tuplas.

- Remover de **Cervejas(nome, fabr)** todas as cervejas para as quais existe outra cerveja do mesmo fabricante.

DELETE FROM Cervejas c
WHERE EXISTS (

```
SELECT nome FROM Cervejas  
WHERE fabr = c.fabr AND  
nome <> c.nome);
```

Cervejas do mesmo fabricante e nome diferente do representada pela tupla b

Semântica da Remoção --- (1)

- Suponha que Ambev só faça Skoll e Brahma.
- Suponha primeiro se atribui à tupla c a Skoll.
- A sub-consulta não é vazia porque existe a tupla para Brahma, por tanto eliminamos a Skoll.
- Agora, quando c seja a tupla para a Brahma, eliminamos essa tupla também?

Semântica da Remoção --- (2)

- **Resposta:** eliminamos também a Brahma.
- Remoção se realiza em duas etapas:
 1. Marcar todas as tuplas para as que a condição de remoção na cláusula WHERE se satisfaz.
 2. Remover as tuplas marcadas.

Atualizações

- Para atualizar atributos em certas tuplas de uma relação: UPDATE
<relação>
SET <lista de atribuições>
WHERE <condição nas tuplas>;

Exemplo: Atualização

- Modificar o fone de Pedro para 019-32456734:

```
UPDATE Pessoas
```

```
SET fone = '019-32456734'
```

```
WHERE nome = 'Pedro';
```


Exemplo: modificar várias tuplas

- Em `Vendem(bar, cerveja, preço)` colocar como preço máximo das cervejas o valor de R\$4.00:

```
UPDATE Vendem  
SET preço = 4.00  
WHERE preço > 4.00;
```

Ordem nas atualizações

Para as cervejas do bar 'Vitoria' aumentar em 20%
preço das cervejas com preço menor que 2.00 e 5%
preço das cervejas com preço maior que 2.00.

```
UPDATE Vendem  
SET preço = 1.2 * preço  
WHERE preço < 2.00 and bar = 'Vitoria'
```

```
UPDATE Vendem  
SET preço = 1.05 * preço  
WHERE preço >= 2.00 and bar = 'Vitoria'
```