

SQL - Structured Query Language : DML – Data Manipulation Language

Comandos de Manipulação de dados em SQL

Manipulação de Dados em SQL

- Existem três comandos SQL para modificar o banco de dados: **INSERT**, **DELETE**, e **UPDATE**

INSERT

- Em sua forma simplificada, é utilizado para adicionar uma ou mais tuplas a uma relação
- Os valores dos atributos devem ser listados na mesma ordem como foram especificados os atributos no comando **CREATE TABLE**

INSERT (cont.)

- Exemplo:

```
U1:INSERT INTO EMPREGADO VALUES ('Richard', 'K',  
    'Marini', '653298653', '30-DEC-52', '98 Oak Forest,Katy,TX',  
    'M', 37000,'987654321', 4 )
```

- Uma forma alternativa de INSERT especifica explicitamente os nomes dos atributos que correspondem aos valores na nova tupla

- Atributos com valores NULL podem ser deixados de fora

- Exemplo: Insira uma tupla para um novo EMPREGADO para o qual só se conheça o PNOOME, UNOME e SSN.

```
U1A: INSERT INTO EMPREGADO (PNOOME, UNOME, SSN)  
    VALUES ('Richard', 'Marini', '653298653')
```

INSERT (cont.)

- Nota Importante: Somente as restrições especificadas em comandos DDL são automaticamente impostas pelo SGBD quando atualizações são aplicadas ao banco de dados
- Outra variação de INSERT permite a inserção de múltiplas tuplas

```
INSERT INTO EMPREGADO (PNOOME, UNOME, SSN)  
    VALUES ('Richard', 'Marini', '653298653'),  
           ('John', 'New York', '234976391')
```

INSERT (cont.)

- Exemplo: Suponha que seja criada uma tabela temporária que tem o nome, número de empregados e total de salários para cada departamento.

- Uma tabela DEPTS_INFO é criada pela U3A, e é carregada com o sumário de informações recuperadas do banco de dados pela consulta U3B.

```
U3A:      CREATE TABLE DEPTS_INFO
           (DEPT_NOME      VARCHAR(10),
            NUM_EMPS       INTEGER,
            TOTAL_SAL      INTEGER);
```

```
U3B:      INSERT INTO DEPTS_INFO (DEPT_NOME,
                                NUM_EMPS, TOTAL_SAL)
           SELECT DNAME, COUNT (*), SUM (SALARIO)
           FROM   DEPARTAMENTO, EMPREGADO
           WHERE  DNUMERO=DNO
           GROUP BY DNAME ;
```

INSERT (cont.)

- Nota: A tabela DEPTS_INFO pode não ser atualizada se mudar as tuplas das relações em DEPARTAMENTO ou em EMPREGADOS após usar U3B. É necessário criar uma visão para manter tal tabela atualizada.

DELETE

- Exclui tuplas de uma relação
 - Contém uma cláusula WHERE para selecionar as tuplas a serem excluídas
 - A integridade referencial deve ser mantida
 - As tuplas são deletadas de apenas uma tabela por vez (a menos que um CASCADE seja especificado em uma restrição de integridade referencial)
 - A ausência da cláusula WHERE especifica que todas as tuplas da relação serão excluídas; a tabela então se tornará vazia
 - O número de tuplas excluídas depende do número de tuplas da relação que satisfaçam a cláusula WHERE

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

9

DELETE (cont.)

- Examples:

U4A:	DELETE FROM WHERE	EMPREGADO UNOME='Brown'
U4B:	DELETE FROM WHERE	EMPREGADO SSN='123456789'
U4C:	DELETE FROM WHERE	EMPREGADO DNO IN (SELECT DNUMERO FROM DEPARTAMENTO WHERE DNOME='Pesquisa')
U4D:	DELETE FROM	EMPREGADO

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

10

UPDATE

- Usada para modificar valores de atributos de uma ou mais tuplas selecionadas
- Uma cláusula WHERE seleciona as tuplas a serem modificadas
- Uma cláusula SET adicional especifica os atributos a serem modificados e seus novos valores
- Cada comando modifica tuplas da mesma relação
- A integridade referencial deve ser mantida

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

11

UPDATE (cont.)

- Exemplo: Mude a localização e o número do departamento de controle do projeto número 10 para 'Bellaire' e 5, respectivamente.

```
U5:      UPDATE  PROJETO
          SET      PLOCALIZACAO = 'Bellaire',
                  DNUM = 5
          WHERE    PNUMERO=10
```

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

12

UPDATE (cont.)

- Exemplo: Dê a todos os empregados do departamento de 'Pesquisa' um aumento de 10%.

```
U6: UPDATE      EMPREGADO
      SET        SALARIO = SALARIO * 1.1
      WHERE      DNO IN (SELECT DNUMERO
                           FROM    DEPARTAMENTO
                           WHERE    DNOME='Pesquisa')
```

- Aqui, a modificação do valor do SALARIO depende do valor original do SALARIO em cada tupla
 - A referência ao atributo SALARIO do lado direito do = se refere ao valor antigo do SALARIO **antes** da modificação
 - A referência ao atributo SALARIO do lado esquerda do = se refere ao valor do novo SALARIO **depois** da modificação

Comandos de Consulta em SQL Parte 1/2

Consultas de Dados em SQL

- Serve para buscar dados armazenados no BD;
- É altamente flexível, o que facilita a filtragem dos dados recuperados em uma consulta;
- É baseada na álgebra relacional e no cálculo relacional de tuplas.

Resultado de Consultas em SQL

- A SQL tem um comando básico para retornar informações de um banco de dados; o comando **SELECT**
 - Esta não é a mesma operação **SELEÇÃO** da álgebra relacional
- Distinções importantes entre a SQL e o modelo relacional formal:
 - A SQL permite que uma tabela tenha duas ou mais tuplas que são idênticas em todos os seus valores de atributos
 - Consequentemente, uma relação SQL (tabela) é um **multi-conjunto** (também chamado **saco**) de tuplas; não um conjunto de tuplas
- Relações SQL podem ser restringidas a conjuntos pela especificação de **CHAVES PRIMÁRIAS** ou atributos **ÚNICOS**, ou usando a operação **DISTINCT** na consulta

Resultado de Consultas em SQL (cont.)

- Um **saco** ou **multiconjunto** é como um conjunto, mas um elemento pode aparecer mais de uma vez.
 - Exemplo: {A, B, C, A} é um saco. {A, B, C} é também um saco que também é um conjunto.
 - Sacos se assemelham a listas, porém a ordem não é relevante em um saco.
- Exemplo:
 - {A, B, A} = {B, A, A} como sacos
 - Entretanto, [A, B, A] não é igual a [B, A, A] como listas

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

17

Resultado de Consultas em SQL (cont.)

- Forma básica de um comando SQL SELECT é chamado um mapeamento ou um bloco SELECT-FROM-WHERE

SELECT <lista de atributos>
FROM <lista de tabelas>
WHERE <condição>

- <lista de atributos> é uma lista de nomes de atributos cujos valores serão recuperados pela consulta
- <lista de tabelas> é uma lista dos nomes das relações necessárias para o processamento da consulta
- <condição> é uma expressão condicional (booleana) que identifica as tuplas que serão recuperadas pela consulta

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

18

Esquema do Banco de Dados Relacional



Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

19

Banco de Dados Populado

EMPREGADO	PNUME	MINICIAL	UNOME	SSN	DATANASC	ENDERECO	SEXO	SALARIO	SUPERSSN	DNO
John	B	Smith	123456789	1955-01-09	731 Fondren, Houston, TX	M	30000	333445555	5	
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5	
Alicia	J	Zelaysa	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4	
Jennifer	S	Walace	987654321	1941-06-20	291 Berry, Bellare, TX	F	43000	888665555	4	
Ramesh	K	Narsyan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5	
Joyce	A	Englah	453453453	1972-07-31	5831 Rice, Houston, TX	F	25000	333445555	5	
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4	
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	null	1	

DEPT LOCALIZACOES	DNUMERO	DLOCALIZACAO
	1	Houston
	4	Stafford
	5	Bellare
	5	Sugarland
		Houston

DEPARTAMENTO	DNOME	DNUMERO	GERSSN	GERDATAINICIO
Pesquisa		5	333445555	1989-05-22
Administracao		4	987654321	1995-01-01
Sede administrativa		1	888665555	1981-06-19

TRABALHA_EM	ESSN	PNO	HORAS
	123456789	1	32.5
	123456789	2	7.5
	666884444	3	40.0
	453453453	1	20.0
	453453453	2	20.0
	333445555	2	10.0
	333445555	3	10.0
	333445555	10	10.0
	333445555	20	10.0
	999887777	30	30.0
	999887777	10	10.0
	987987987	10	35.0
	987987987	30	5.0
	987654321	30	20.0
	987654321	20	15.0
	888665555	20	null

PROJETO	PJNOME	PNUMERO	PLOCALIZACAO	DNUM
	ProdutoX	1	Bellare	5
	ProdutoY	2	Sugarland	5
	ProdutoZ	3	Houston	5
	Automatizagao	10	Stafford	4
	Reorganizagao	20	Houston	1
	Novos Beneficios	30	Stafford	4

DEPENDENTE	ESSN	NOME_DEPENDENTE	SEXO	DATANASC	PARENTESCO
	333445555	Alice	F	1985-04-05	FILHA
	333445555	Theodore	M	1983-10-25	FILHO
	333445555	Joy	F	1958-05-03	CONJUGE
	987654321	Abner	M	1942-02-28	CONJUGE
	123456789	Michael	M	1989-01-04	FILHO
	123456789	Alice	F	1988-12-30	FILHA
	123456789	Elizabeth	F	1967-05-05	CONJUGE

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

20

Consulta SQL Simples

- Consultas SQL básicas usam as seguintes operações da álgebra relacional:
 - SELEÇÃO
 - PROJEÇÃO
 - JUNÇÃO
- Todos os exemplos subsequentes usam o banco de dados EMPRESA

Consulta SQL Simples (cont.)

- Exemplo de uma consulta simples em uma relação
- Consulta 0: Recupere o aniversário e o endereço dos empregados cujo o nome seja 'John B. Smith'.

```
Q0:SELECT      DATANASC, ENDERECO
FROM          EMPREGADO
WHERE         PNOME='John' AND MINICIAL='B'
AND          UNOME='Smith'
```
- Similar a um par de operações SELECT-PROJECT da álgebra relacional:
 - A cláusula SELECT especifica os atributos de projeção e a cláusula WHERE especifica a condição de seleção
- Entretanto, o resultado de uma consulta pode conter tuplas duplicadas

Consulta SQL Simples (cont.)

- Consulta 1: Recupere o nome e o endereço de todos os funcionários que trabalham no departamento de 'Pesquisa'.

```
Q1:SELECT      PNAME, UNOME, ENDERECO  
      FROM      EMPREGADO, DEPARTAMENTO  
      WHERE     DNAME='Pesquisa' AND DNUMERO=DNO
```

- Similar a uma sequência de operações SELECT-PROJECT-JOIN da álgebra relacional
- (DNAME='Pesquisa') é uma condição de seleção (corresponde a uma operação de SELECAO na álgebra relacional)
- (DNUMERO=DNO) é uma condição de junção (corresponde a uma operação de JUNCAO na álgebra relacional)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

23

Consulta SQL Simples (cont.)

- Consulta 2: Para cada projeto localizado em 'Stafford', liste o número do projeto, o número do departamento responsável e o último nome do gerente do departamento, seu endereço e data de nascimento.

```
Q2: SELECT  PNUMERO, DNUM, UNOME, DATANASC, ENDERECO  
      FROM    PROJETO, DEPARTAMENTO, EMPREGADO  
      WHERE   DNUM=DNUMERO AND GERSSN=SSN  
              AND PLOCALIZACAO='Stafford'
```

- Em Q2, existem duas condições de junção
- A condição de junção DNUM=DNUMERO relaciona um projeto a seu departamento de controle
- A condição de junção GERSSN=SSN relaciona o departamento de controle com o empregado que administra esse departamento

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

24

Aliases, * e DISTINCT, cláusula WHERE ausente

- Em SQL, pode-se usar o mesmo nome para dois ou mais atributos sendo que os atributos estejam em relações diferentes
- Uma consulta que se refere a dois ou mais atributos com o mesmo nome deve qualificar o nome do atributo com o nome da relação através da prefixação do nome da relação ao nome do atributo
- Exemplo:
 - EMPREGADO. NOME, DEPARTAMENTO. NOME

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

25

ALIASES

- Algumas consultas precisam se referenciar a mesma relação mais de uma vez
 - Nesse caso, *alias*es (pseudônimos) são dados aos nomes das relações
- Consulta 8: Para cada empregado, recupere seu nome e o nome de seu superior imediato.

```
Q8: SELECT      E.PNOME, E.UNOME, S.PNOME, S.UNOME
      FROM      EMPREGADO E, EMPREGADO S
      WHERE     E.SUPERSSN=S.SSN
```

- Em Q8, os nomes alternativos E e S são chamados *alias*es ou *variáveis de tupla* para a relação EMPREGADO
- Pode-se pensar em E e S como duas cópias diferentes de EMPREGADO; E representa os empregados no papel de *supervisionados* e S representa os empregados no papel de *supervisores*

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

26

ALIASES (cont.)

- Aliases pode ser também utilizados em qualquer consulta SQL por conveniência
- Pode-se também utilizar a palavra chave AS para especificar aliases

```
Q8:SELECT  E.PNOME AS PRIMEIRONOME,  
          S.PNOME, S.UNOME  
FROM      EMPREGADO AS E,  
          EMPREGADO AS S  
WHERE     E.SUPERSSN=S.SSN
```

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

27

Cláusula WHERE Ausente

- A ausência da cláusula *WHERE* indica que não há nenhuma condição; consequentemente, todas as tuplas das relações na cláusula FROM serão selecionadas
 - Isso é equivalente a condição WHERE ser verdadeira
- Consulta 9: Recupere o SSN para todos os empregados.

```
Q9:      SELECT      SSN  
FROM      EMPREGADOS
```

- Se mais de uma relação for especificada na cláusula FROM e não existir condição de junção, então será obtido o PRODUTO CARTESIANO dessas relações

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

28

Cláusula WHERE Ausente (cont.)

■ Exemplo:

Q10: SELECT SSN, DNOME
 FROM EMPREGADO,
 DEPARTAMENTO

- É extremamente importante especificar todas as condições de seleção e de junção na cláusula WHERE; senão, podem ocorrer resultados incorretos ou muito grandes

Uso do * (Asterisco)

- Para recuperar os valores de todos os atributos das tuplas selecionadas, o * é usado, que significa selecionar todos os atributos

Exemplos:

Q1C: SELECT *
 FROM EMPREGADO
 WHERE DNO=5

Q1D: SELECT *
 FROM EMPREGADO, DEPARTAMENTO
 WHERE DNOME='Pesquisa' AND
 DNO=DNUMERO

Uso do DISTINCT

- A SQL não trata uma relação como um conjunto; assim, tuplas duplicadas podem aparecer
- Para eliminar tuplas duplicadas no resultado de uma consulta, a palavra chave **DISTINCT** é usada
- Por exemplo, o resultado de Q11 pode ter valores de SALARIO duplicados, enquanto que Q11A não tem nenhum valor duplicado

Q11: SELECT SALARIO
 FROM EMPREGADO

Q11A: SELECT **DISTINCT** SALARIO
 FROM EMPREGADO

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

31

Operações de Conjuntos

- A SQL tem diretamente incorporada algumas operações de conjunto
- Existe uma operação de união (UNION), e em algumas versões da SQL há operações de diferença de conjuntos (EXCEPT) e interseção (INTERSECT)
- As relações resultantes dessas operações de conjunto são conjuntos de tuplas; tuplas duplicadas são eliminadas no resultado
- Operações de conjuntos são aplicáveis apenas a relações união compatíveis; as duas relações devem ter os mesmos atributos e os atributos devem aparecer na mesma ordem

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

32

Operações de Conjuntos (cont.)

- Consulta 4: Faça uma lista de todos os números de projetos para projetos que envolvam um empregado cujo último nome seja 'Smith', como trabalhador ou como gerente do departamento que controla o projeto.

```
Q4:      (SELECT      PNUMERO
          FROM        PROJETO, DEPARTAMENTO,
                    EMPREGADO
          WHERE       DNUM=DNUMERO AND
                    GERSSN=SSN AND UNOME='Smith')

          UNION
          (SELECT      PNUMERO
          FROM        PROJETO, TRABALHA_EM,
                    EMPREGADO
          WHERE       PNUMERO=PNO AND
                    ESSN=SSN AND NOME='Smith')
```

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

33

Junção de Relações

- Pode-se especificar uma "junção de relações" na cláusula FROM
 - Se parece com qualquer outra relação, mas é o resultado de uma junção
 - Existem dois tipos de junções:
 - Internas (INNER JOIN)
 - Externas (OUTER JOIN)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

34

Junção de Relações (cont.)

- As junções internas podem ser feitas com:
 - **NATURAL JOIN**: Nesta caso, a junção é feita através dos nomes dos campos que forem iguais nas duas relações;
 - **JOIN**: Aqui a junção deve ser especificada através de:
 - **ON**: usada quando os nomes dos campos de junção das duas relações foram diferentes;
 - **USING**: usado com o nomes dos campos de junção das duas relações foram iguais.
- A palavra **INNER** é opcional.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

35

Junção de Relações (cont.)

- **Exemplos:**
Q1: SELECT PNOOME, UNOME, ENDERECO
FROM EMPREGADO, DEPARTAMENTO
WHERE DNOME='Pesquisa' AND DNUMERO=DNO
- **Pode ser escrito como:**
Q1: SELECT PNOOME, UNOME, ENDERECO
FROM EMPREGADO **JOIN** DEPARTAMENTO
ON (DNUMERO=DNO)
WHERE DNOME='Pesquisa'
- **ou como:**
Q1: SELECT PNOOME, UNOME, ENDERECO
FROM EMPREGADO **NATURAL JOIN**
DEPARTAMENTO
AS DEPT(DNOME, DNO, GERSSN, GERDATA)
WHERE DNOME='Pesquisa'

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

36

Junção de Relações (cont.)

- Outro exemplo: Q2 pode ser escrito como segue; Isso ilustra múltiplas junções

```
Q2:      SELECT      PNUMERO, DNUM, UNOME,  
                  ENDERECO, DATANASC  
FROM PROJETO JOIN DEPARTAMENTO  
           ON      (DNUM=DNUMERO) JOIN  
                  EMPREGADO ON  
                  (GERSSN=SSN)  
WHERE    PLOCALIZACAO='Stafford'
```