

Bases de Dados

SQL- Manipulação de Dados



Linguagem de Manipulação de Dados

- ▶ Uma instrução DML é executada quando se:
 - ▶ Acrescentam novas colunas a uma tabela
 - ▶ Modificam linhas **existentes** numa tabela
 - ▶ Removem linhas existentes de uma tabela



Acrescentar uma Nova Linha a uma Tabela

50	DESENVOLV	DETROIT
----	-----------	---------

Nova linha

DEP

NDEP	NOME	LOC
10	CONTABIL	NEW YORK
20	INVESTIGA	DALLAS
30	VENDAS	CHICAGO
40	OPERACOES	BOSTON

“...insere uma nova linha na tabela DEP...”

DEP

NDEP	NOME	LOC
10	CONTABIL	NEW YORK
20	INVESTIGA	DALLAS
30	VENDAS	CHICAGO
40	OPERACOES	BOSTON
50	DESENVOLV	DETROIT



A Instrução INSERT

- ▶ Acrescenta novas linhas a uma tabela usando a instrução INSERT.

```
INSERT INTO  table [(column [, column...])]  
VALUES      (value [, value...]);
```

- ▶ Somente uma linha é inserida em cada instante com esta sintaxe.



Inserir Novas Linhas

- ▶ Inserir uma nova linha contendo valores para cada coluna.
- ▶ Listar os valores, na ordem default das colunas da tabela.
- ▶ Opcionalmente listar as colunas na cláusula INSERT.

```
SQL> INSERT INTO  dep (ndep, nome, loc)  
      2 VALUES      (50, 'DESENVOLV', 'DETROIT');  
1 row created.
```

- ▶ Incluir os valores de strings e datas dentro de plicas.



Inserir Linhas com Valores Nulos

- ▶ Método implícito: Omitir a coluna da lista das colunas.

```
SQL> INSERT INTO    dep (ndep, nome )  
2 VALUES          (60, 'MIS') ;  
1 row created.
```

- ▶ Método explícito: Especificar a palavra chave NULL.

```
SQL> INSERT INTO    dep  
2 VALUES          (70, 'FINANCA', NULL) ;  
1 row created.
```



Inserir Valores Especiais

- ▶ A função SYSDATE grava a data e hora actual.

```
SQL> INSERT INTO    emp (nemp, nome, funcao,  
2 encar, data_entrada, sal, premios,  
3 ndep)  
4 VALUES          (7196, 'GREEN', 'VENDEDOR',  
5 7782, SYSDATE, 2000, NULL,  
6 10) ;  
1 row created.
```



Inserir Valores de Data Específicos

- ▶ Adicionar um novo empregado.

```
SQL> INSERT INTO emp
2  VALUES      (2296, 'AROMANO', 'VENDEDOR', 7782,
3                TO_DATE('FEB 3, 1997', 'MON DD, YYYY'),
4                1300, NULL, 10);
1 row created.
```

- ▶ Verificar a adição.

NEMP	NOME	FUNCAO	ENCAR	DATA_ENTRADA	SAL	PREMIOS	NDEP
2296	AROMANO	SALESMAN	7782	03-FEB-97	1300		10



Copiar Linhas de Outra Tabela

- ▶ Escrever a instrução INSERT com uma subquery.

```
SQL> INSERT INTO gestores(id, nome, salario, data_entrada)
2      SELECT nemp, nome, sal, data_entrada
3      FROM emp
4      WHERE funcao = 'GESTOR';
3 rows created.
```

- ▶ Não usar a cláusula VALUES.
- ▶ Satisfazer o número de colunas da clausula INSERT com as da subquery.



Mudar dados Numa Tabela

EMP

NEMP	NOME	FUNCAO	...	NDEP
7839	KING	PRESIDENTE		10
7698	BLAKE	GESTOR		30
7782	CLARK	GESTOR		10
7566	JONES	GESTOR		20
...				

“...actualiza uma linha na tabela EMP...”

EMP

NEMP	NOME	FUNCAO	...	NDEP
7839	KING	PRESIDENTE		10
7698	BLAKE	GESTOR		30
7782	CLARK	GESTOR		20
7566	JONES	GESTOR		20
...				



A Instrução UPDATE

- ▶ Modifica linhas existentes com a instrução UPDATE.

```
UPDATE      table
SET         column = value [, column = value, ...]
[WHERE      condition];
```

- ▶ Actualiza mais do que uma linha de cada vez, se necessário.



Actualizar Linhas numa Tabela

- ▶ Linhas ou linhas específicas são modificadas quando as especificamos na clausula WHERE.

```
SQL> UPDATE emp
2 SET ndep = 20
3 WHERE nemp = 7782;
1 row updated.
```

- ▶ Todas as linhas da tabela são modificadas se omitirmos a clausula WHERE.

```
SQL> UPDATE emp
2 SET ndep = 20;
14 rows updated.
```



Update com Subquery de Várias Colunas

- ▶ Actualizar a função e o departamento do empregado 7698 para os do empregado 7499.

```
SQL> UPDATE emp
2 SET (funcao, ndep) =
3     (SELECT funcao, ndep
4      FROM emp
5      WHERE nemp = 7499)
6 WHERE nemp = 7698;
1 row updated.
```



Actualizar Linhas Baseadas Noutra Tabela

- Usar subqueries nas instruções UPDATE para actualizar linhas da tabela, baseadas nos valores de outras tabelas.

```
SQL> UPDATE emp
2 SET ndep = (SELECT ndep
3             FROM emp
4             WHERE nemp = 7788)
5 WHERE funcao = (SELECT funcao
6                 FROM emp
7                 WHERE nemp = 7788);
2 rows updated.
```



Atualizar Linhas: Erro de restrição de Integridade

```
SQL> UPDATE emp
2 SET ndep = 55
3 WHERE ndep = 10;
```

```
UPDATE emp
*
ERROR at line 1:
ORA-02291: integrity constraint (USR.EMP_DEPTNO_FK)
violated - parent key not found
```



Remover uma Linha de uma Tabela

DEP

NDEP	NOME	LOC
10	CONTAB	NEW YORK
20	INVEST	DALLAS
30	VENDAS	CHICAGO
40	OPERACOES	BOSTON
50	DESENV	DETROIT
60	OUTRO	
...		

“...apaga um linha da tabela DEP...”

DEP

NDEP	NOME	LOC
10	CONTAB	NEW YORK
20	INVEST	DALLAS
30	VENDAS	CHICAGO
40	OPERACOES	BOSTON
60	OUTRO	
...		



A Instrução DELETE

- Pode remover as linhas existentes numa tabela usando a instrução DELETE.

```
DELETE [FROM]  table
[WHERE        condition];
```



Apagar Linhas de uma Tabela

- ▶ Linhas específicas são apagadas quando se especifica a clausula WHERE.

```
SQL> DELETE FROM    dep
      2  WHERE        nome = 'DESENV';
      1 row deleted.
```

- ▶ Todas as linhas na tabela são apagadas se for omitida a clausula WHERE.

```
SQL> DELETE FROM    dep;
      4 rows deleted.
```



Apagar Linhas Baseadas noutra Tabela

- ▶ Usar subqueries nas instruções DELETE para remover linhas de uma tabela, baseadas em valores de outra tabela.

```
SQL> DELETE FROM    emp
      2  WHERE        ndep =
      3
      4                (SELECT    ndep
      5                FROM      dep d
      6                WHERE      d.nome = 'VENDAS');
      6 rows deleted.
```



Apagar Linhas: erro de Restrição de Integridade

```
SQL> DELETE FROM    dep
      2  WHERE        ndep = 10;
```

```
DELETE FROM dep
          *
ERROR at line 1:
ORA-02292: integrity constraint (USR.EMP_DEPTNO_FK)
violated - child record found
```

Não pode apagar uma linha que contenha
uma chave primária que é usada como
chave estrangeira noutra tabela.

Resumo

Instrução	Descrição
INSERT	Adiciona uma nova linha a uma tabela
UPDATE	Modifica linhas existentes numa tabela
DELETE	Remove linhas existentes de uma tabela