

The logo for Oracle Academy is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by two horizontal dark gray bars, one at the top and one at the bottom.

# ORACLE

## Academy

# Database Foundations

6-4

Linguagem de Manipulação de Dados (DML)

**ORACLE**  
Academy



Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados.

# Roteiro

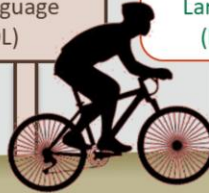
Introdução ao  
Oracle  
Application  
Express (APEX)

Structured  
Query  
Language  
(SQL)

Data  
Definition  
Language  
(DDL)

Data  
Manipulation  
Language  
(DML)

Transaction  
Control  
Language  
(TCL)



Recuperand  
o Dados com  
SELECT

Restringindo  
Dados com  
WHERE

Classificando  
Dados com  
ORDER BY

Unindo Tabelas  
com JOIN

Parte 6

**ORACLE**  
Academy

DFo 6-4  
Data Manipulation Language (DML)

Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados.

3

# Objetivos

- Esta lição abrange os seguintes objetivos:
  - Descrever a finalidade do DML (Data Manipulation Language)
  - Explicar as operações DML necessárias para gerenciar os dados de uma tabela de banco de dados:
    - INSERT
    - UPDATE
    - DELETE



# Data Manipulation Language

- Uma instrução DML é executada quando você:
  - Adiciona novas linhas a uma tabela (INSERT)
  - Modifica as linhas existentes em uma tabela (UPDATE)
  - Remove as linhas existentes de uma tabela (DELETE)
- Uma transação consiste em um conjunto de instruções DML que formam uma unidade de trabalho lógica



Considere um banco de dados bancário. Quando um cliente do banco transfere dinheiro de uma conta poupança para uma conta corrente, a transação pode consistir em três operações separadas: reduzir a conta poupança, aumentar a conta corrente e registrar a transação no diário de transações. O servidor Oracle deve garantir que as três instruções SQL sejam executadas para manter as contas no saldo adequado. Quando algo impede a execução de uma das instruções na transação, as outras instruções da transação devem ser desfeitas.

# Adicionando uma Nova Linha a uma Tabela

## DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500

...

## Nova linha

70   Public Relations   204  
2700

Insira uma nova linha  
na tabela  
DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
70	Public RElations	204	2700
80	Sales	149	2500

## Sintaxe da Instrução INSERT

- Adicione linhas a uma tabela usando a instrução INSERT:

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

- Com essa sintaxe, somente uma linha é inserida por vez

Na sintaxe:

- *table* é o nome da tabela.
- *column* é o nome da coluna da tabela que você deseja preencher.
- *value* é o valor correspondente da coluna.



## Sintaxe da Instrução INSERT

- Antes de executar operações DML em uma tabela, faça uma cópia da tabela:

```
CREATE TABLE copy_departments  
AS (SELECT * FROM departments);
```

- Lembre-se de que, quando você copia uma tabela dessa maneira, todas as restrições não são copiadas (somente NOT NULL)



## Inserindo Linhas

- Se você inserir uma linha que contenha valores para cada coluna, a lista de colunas não será necessária na cláusula INSERT
- Liste os valores na ordem padrão das colunas na tabela.
- Deve ser fornecido um valor para cada coluna

```
INSERT INTO copy_departments  
VALUES (40, 'Advertising', 201, 1800);
```

## Inserindo Linhas

- Como opção, liste as colunas na cláusula INSERT

```
INSERT INTO copy_departments (department_id,  
                               department_name, manager_id, location_id)  
VALUES (70, 'Public Relations', 100, 1700);
```

- Liste os valores na mesma ordem que os campos listados
- Coloque os valores de data e caractere entre aspas simples

## Cenário de Caso: Inserindo Linhas

**Entendo que a instrução INSERT é usada para adicionar linhas a uma tabela. Será possível inserir uma linha se algumas colunas não tiverem valores?**



Aluno



Corpo Docente

**Com certeza. Você pode usar a instrução INSERT para especificar as colunas para as quais não serão inseridos valores.**

## Inserindo Linhas com Valores Nulos

- Método explícito : Omita a coluna da lista de colunas

```
INSERT INTO copy_departments (department_id,  
                               department_name)  
VALUES (30, 'Purchasing');
```

- Método implícito : Especifique a palavra-chave NULL na cláusula VALUES

```
INSERT INTO copy_departments  
VALUES (100, 'Finance', NULL, NULL);
```

Certifique-se de que você possa usar valores nulos na coluna de destino verificando o status `Null` com o comando `DESCRIBE`.

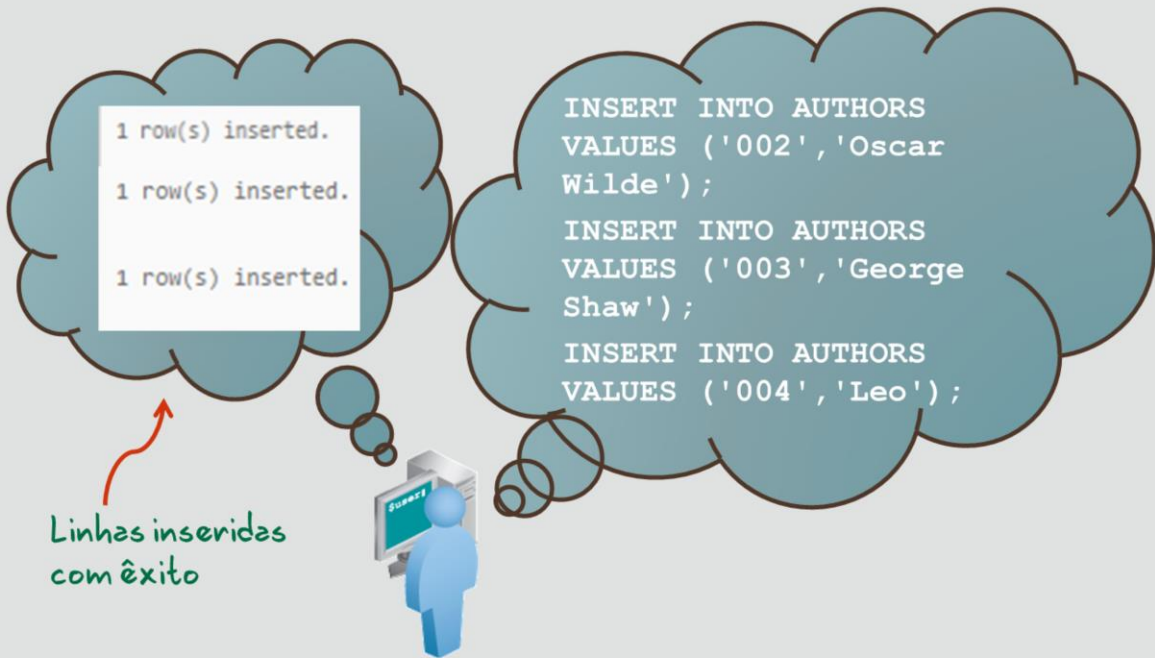
O servidor Oracle impõe automaticamente todos os tipos de dados, intervalos de dados e restrições de integridade de dados. Qualquer coluna que não esteja listada explicitamente obterá um valor nulo na nova linha, a menos que haja valores padrão para as colunas ausentes usadas.

Os erros comuns que podem ocorrer durante a entrada do usuário são verificados na seguinte ordem:

- Um valor obrigatório está ausente para uma coluna `NOT NULL`.
- Um valor duplicado viola uma restrição de chave primária ou exclusiva.
- O valor `Any` viola uma restrição `CHECK`.
- Uma chave estrangeira viola a restrição de integridade referencial.
- Incompatibilidade de tipos de dados ou valores muito grandes para caber na coluna.

**Observação:** o uso da lista comum é recomendável porque facilita a leitura da instrução `INSERT` e a torna mais confiável e menos suscetível a erros.

## Cenário de Caso: Inserindo Linhas



## Inserindo Valores Especiais

- Você pode usar funções para inserir valores especiais na tabela
- A função SYSDATE registra a data e a hora atuais

```
INSERT INTO copy_employees (employee_id, first_name,  
                             last_name, email, phone_number,  
                             hire_date, job_id, salary,  
                             commission_pct, manager_id,  
                             department_id)  
VALUES ( 113, 'Louis',  
        'Popp', 'LPOPP', '515.124.4567',  
        SYSDATE, 'AC_ACCOUNT', 6900,  
        NULL, 205,  
        110);
```

**ORACLE**  
Academy

DFo 6-4  
Data Manipulation Language (DML)

Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados. 14

Observação: em primeiro lugar, crie a tabela copy\_employees: CREATE TABLE copy\_employees AS (SELECT \* FROM employees);

O exemplo do slide registra informações do funcionário Popp na tabela EMPLOYEES. Ele fornece a data e a hora atuais na coluna HIRE\_DATE. A função SYSDATE é usada para retornar a data e a hora atuais do servidor de banco de dados.

Você também pode usar a função CURRENT\_DATE para retornar a data atual no fuso horário da sessão.

Ao inserir linhas em uma tabela, você também pode usar a função USER para registrar o nome de usuário atual.

## Inserindo Valores Específicos de Data e Hora

- Adicione um funcionário

```
INSERT INTO copy_employees
VALUES(114, 'Den', 'Raphealy', 'DRAPHEAL',
      '515.127.4561',
      TO_DATE('Dec 7, 2002', 'MON DD, YYYY'),
      'SA_REP', 11000, 0.2, 100, 60, NULL);
```

1 rows inserted

- Verifique sua inclusão

114	Den	Raphealy	DRAPHEAL	515.127.4561	07-Dec-2002
-----	-----	----------	----------	--------------	-------------



# Exercício 1 do Projeto

- DFo\_6\_4\_1\_Project
  - Banco de Dados da Oracle Baseball League
  - Usando operações DML para gerenciar tabelas de banco de dados: Inserindo registros



# Alterando Dados em uma Tabela

## EMPLOYEES

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17-Jun-1987	AD_PRES	24000	-	-	90
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-Sep-1989	AD_VP	17000	-	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	13-Jan-1993	AD_VP	17000	-	100	90
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-Sep-1987	AD_ASST	4400	-	101	10

...

Atualize as linhas da tabela EMPLOYEES:

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17-Jun-1987	AD_PRES	24000	-	-	60
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-Sep-1989	AD_VP	17000	-	100	60
102	Lex	De Haan	LDEHAAN	515.123.4569	13-Jan-1993	AD_VP	17000	-	100	60
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-Sep-1987	AD_ASST	4400	-	101	10

...

**ORACLE**  
Academy

DFo 6-4  
Data Manipulation Language (DML)

Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados. 17

## Sintaxe da Instrução UPDATE

É recomendável  
deixar a instrução  
UPDATE em uma  
linha própria

- Modifique os valores existentes em uma tabela com a instrução UPDATE:

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

- Em geral, use a coluna de chave primária na cláusula WHERE a fim de identificar uma única linha para atualização
- Atualize mais de uma linha por vez (se necessário)

Na sintaxe:

- `table` é o nome da tabela.
- `column` é o nome da coluna a ser preenchida na tabela.
- `value` é o valor correspondente ou a subconsulta para a coluna.
- `condition` identifica as linhas a serem atualizadas e consiste em nomes de colunas, expressões, constantes, subconsultas e operadores de comparação.

## Atualizando Linhas em uma Tabela

- Os valores de uma linha ou linhas específicas serão modificados se você especificar a cláusula WHERE:

```
UPDATE copy_employees
SET    department_id = 50
WHERE  employee_id = 113;
```

- Os valores de todas as linhas da tabela serão modificados se você omitir a cláusula WHERE:

```
UPDATE copy_employees
SET    department_id = 110;
```

Por exemplo, um funcionário que era um SA\_REP alterou sua função para IT\_PROG . Portanto, seu JOB\_ID precisa ser atualizado, e o campo de comissão precisa ser definido como NULL.

```
UPDATE copy_employees
SET job_id = 'IT_PROG', commission_pct = NULL
...
WHERE employee_id = 114;
```

**Observação:** a tabela copy\_employees tem os mesmos dados que a tabela EMPLOYEES.

## Atualizando Linhas em uma Tabela

- Especifique SET column\_name= NULL para atualizar um valor de coluna para NULL.

```
UPDATE    copy_employees
SET       department_id = NULL
WHERE     employee_id = 124;
```

# Violando Restrições

```
UPDATE employees
SET    department_id = 55
WHERE  department_id = 110;
```

Error starting at line 1 in command:

```
UPDATE employees
SET    department_id = 55
WHERE  department_id = 110
```

Error report:

```
SQL Error: ORA-02291: integrity constraint (ORA1.EMP_DEPT_FK) violated - parent key not found
02291. 00000 - "integrity constraint (%s.%s) violated - parent key not found"
*Cause:      A foreign key value has no matching primary key value.
*Action:     Delete the foreign key or add a matching primary key.
```

- O departamento 55 não existe na tabela pai – DEPARTMENTS

**ORACLE**  
Academy

DFo 6-4  
Data Manipulation Language (DML)

Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados. 21

**Observação:** observe que a consulta usa a tabela employees que tem restrições de integridade em vigor (copy\_employees – não tem essas restrições em vigor).

Quando houver restrições nas colunas, um erro será retornado se você tentar violar a regra da restrição. Por exemplo, um erro será retornado se você tentar atualizar um registro com um valor vinculado a uma restrição de integridade.

No exemplo do slide, como o departamento 55 não existe na tabela pai DEPARTMENTS, você recebe a violação "parent key not found" ORA-02291.

## Removendo uma Linha de uma Tabela

- Este slide mostra que o departamento Public Relations foi removido da tabela DEPARTMENTS (supondo-se que não haja restrições violadas nessa tabela)

**DEPARTMENTS**

DEPARTMENT _ID	DEPARTMENT _NAME	MANAGER _ID	LOCATION _ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
70	Public RElations	204	2700
80	Sales	149	2500

...

**ORACLE**  
Academy

DFo 6-4  
Data Manipulation Language (DML)

**Linha excluída da  
tabela DEPARTMENTS:**

DEPARTMENT _ID	DEPARTMENT _NAME	MANAGER _ID	LOCATION _ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500
90	Executive	100	1700

...

Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados. 22



# Instrução DELETE

- Você pode remover as linhas existentes de uma tabela usando a instrução DELETE:

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

Na sintaxe:

- `table` é o nome da tabela.
- `condition` identifica as linhas a serem excluídas e consiste em nomes de colunas, expressões, constantes, subconsultas e operadores de comparação.

**Observação:** se nenhuma linha for excluída, a mensagem "0 rows deleted" será retornada.

## Excluindo Linhas de uma Tabela

- Linhas específicas serão excluídas se você incluir a cláusula WHERE:

```
DELETE FROM copy_departments  
WHERE department_name = 'Purchasing';
```

- Todas as linhas da tabela serão excluídas se você omitir a cláusula WHERE.

```
DELETE FROM copy_departments;
```

Veja a seguir outro exemplo de remoção das linhas identificadas na cláusula WHERE:

```
DELETE FROM copy_departments WHERE department_id IN (30, 40);
```

## Violando Restrições

- Não é possível excluir uma linha que contenha uma chave primária usada como uma chave estrangeira em outra tabela

```
DELETE
FROM   departments
WHERE  department_id = 60;
```

```
Error starting at line 1 in command:
DELETE FROM departments
WHERE department_id = 60
```

Error report:

```
SQL Error: ORA-02292: integrity constraint (ORA1.JHIST_DEPT_FK) violated - child record found
02292. 00000 - "integrity constraint (%s.%s) violated - child record found"
```

```
*Cause:      attempted to delete a parent key value that had a foreign
              dependency.
```

```
*Action:     delete dependencies first then parent or disable constraint.
```

**ORACLE**  
Academy

DFo 6-4  
Data Manipulation Language (DML)

Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados. 25

**Observação:** observe que a consulta usa a tabela `departments` que tem restrições de integridade em vigor (`copy_departments` – não tem essas restrições em vigor).

Se você tentar excluir um registro com um valor vinculado a uma restrição de integridade, será retornado um erro.

O exemplo do slide mostra uma tentativa de excluir o departamento 60 da tabela `DEPARTMENTS`, mas isso resulta em um erro porque o número do departamento é usado como uma chave estrangeira na tabela `EMPLOYEES`. Se o registro pai que você tentar excluir tiver registros filhos, você receberá a violação "child record found" `ORA-02292`.

## Cenário de Caso: Excluindo Linhas

**É possível excluir todas as linhas de uma tabela, mas deixar a estrutura intacta?**



**Aluno**



**Corpo Docente**

**Com certeza! Você pode usar a instrução TRUNCATE para fazer isso.**

# Instrução TRUNCATE

- Remove todas as linhas de uma tabela, deixando-a vazia e a sua estrutura intacta
- É uma instrução DDL, e não uma instrução DML; não é possível desfazê-la facilmente
- Sintaxe:

```
TRUNCATE TABLE table_name;
```

- Exemplo:

```
TRUNCATE TABLE copy_employees;
```

- **\*\*** Consulte as observações sobre o truncamento das tabelas pai

**ORACLE**  
Academy

DFo 6-4  
Data Manipulation Language (DML)

Copyright © 2020, Oracle y/o sus filiales. Todos los derechos reservados. 27

A instrução `TRUNCATE` é um método mais eficiente para remover todas as linhas de uma tabela ou cluster.

Remover linhas com a instrução `TRUNCATE` é mais rápido do que removê-las com a instrução `DELETE` pelas seguintes razões:

- A instrução `TRUNCATE` é uma instrução DDL e não gera informações de rollback. Essas informações são abordadas mais adiante nesta lição.
- Truncar uma tabela não ativa os acionadores de exclusão da tabela.

Se a tabela for o pai de uma restrição de integridade referencial, não será possível truncá-la. É necessário desativar a restrição antes de emitir a instrução `TRUNCATE`. A desativação de restrições é abordada na lição "Introdução a Instruções DDL".

## Exercício 2 do Projeto

- DFo\_6\_4\_2\_Project
  - Banco de Dados da Oracle Baseball League
  - Usando operações DML para gerenciar tabelas de banco de dados: Atualizando e excluindo registros



# Resumo

- Nesta lição, você deverá ter aprendido a:
  - Descrever a finalidade do DML
  - Listar as operações DML necessárias para gerenciar os dados de uma tabela de banco de dados:
    - INSERT
    - UPDATE
    - DELETE





The logo for Oracle Academy is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by two horizontal dark gray bars, one at the top and one at the bottom.

# ORACLE

## Academy