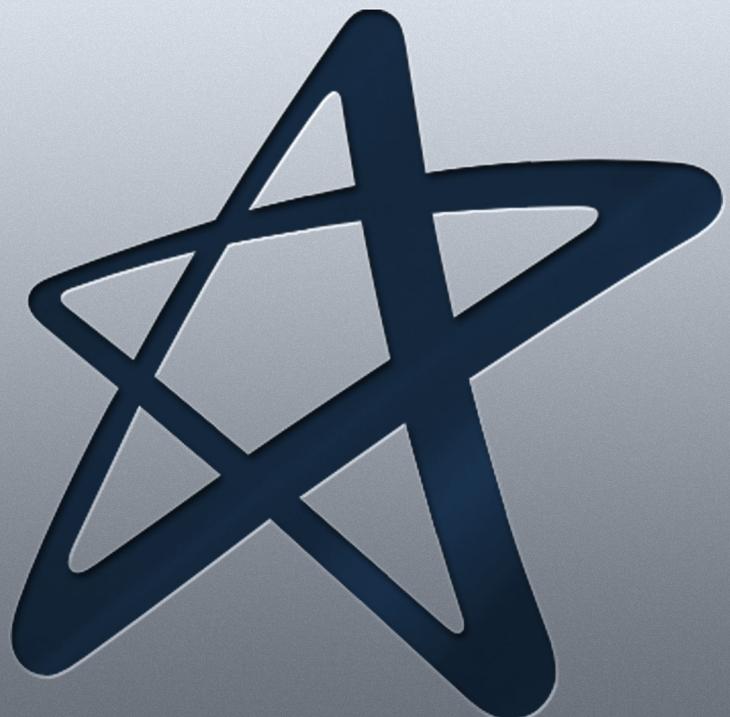


Banco de Dados



Educação a Distância
Cruzeiro do Sul Educacional
Campus Virtual

Material Teórico



Instruções de comando DML

Responsável pelo Conteúdo:

Prof. Ms. Alexander Gobbato Albuquerque

Revisão Textual:

Profa. Esp. Vera Lídia de Sá Cicarone



- Introdução
- UPDATE
- DELETE
- RENAME
- CREATE TABLE AS SELECT
- SELECT
- Cláusula WHERE



Hoje veremos alguns comandos para a manipulação de dados, ou seja, comando DDL (Data Definition Language).

Para obter um bom aproveitamento nesta unidade, vamos conferir sua estrutura:

- **Conteúdo Teórico:** neste link, você encontrará o material principal de estudos sob a forma de texto escrito.
- **Atividade de Sistematização:** os exercícios disponibilizados são de autocorreção e visam a que você pratique o que aprendeu na disciplina. Dessa forma, você poderá identificar os pontos aos quais precisa prestar mais atenção ou sobre ao quais deve pedir esclarecimentos a seu tutor. Além disso, a esses exercícios serão atribuídas notas que farão parte de sua média final na disciplina.
- **Atividade de Aprofundamento:** você deve verificar a atividade que será disponibilizada.
- **Material Complementar e Referências Bibliográficas:** nestes links, você receberá sugestões de leitura e mídias por meio das quais poderá ampliar seus conhecimentos.
- **Vídeo-aula:** aqui serão apresentadas, de forma prática, algumas ferramentas e também a resolução de alguns exercícios. (OBS: este item não aparecerá em todas as unidades.)

Lembramos a você a importância de realizar todas as atividades propostas dentro do prazo estabelecido para cada unidade. Dessa forma, evitará que o conteúdo se acumule e que você tenha problemas ao final do semestre.

Uma última recomendação: caso tenha problemas para acessar algum item da disciplina ou dúvidas com relação ao conteúdo, não deixe de entrar em contato com seu professor tutor através do botão mensagens.

Contextualização

Com o conhecimento adquirido nos comandos DML (Data Manipulation Language), aprenderemos, agora, como inserir, alterar e deletar informações.

Além de trabalharmos com os comandos acima, também veremos como podemos exibir as informações armazenadas.

Então está na hora de aprendermos os comandos DDL (Data Definition Language).

Introdução



Inserir linhas em tabelas

- INSERT
- UPDATE
- DELETE
- RENAME
- CREATE TABLE ... AS SELECT

Recuperar Dados de uma Tabela

- SELECT
- Distinct
- Expressões Aritméticas
- Cláusula Where
- Alias

Pré-requisito para este material:

Logar-se com o usuário HR e Selecionar SQL Commands.

ORACLE® Database Express Edition

Database Login

Enter your database username and password.

Username:

Password:

[Click here to learn how to get started](#)

Links

- License Agreement
- Documentation
- Forum Registration
- Discussion Forum
- Product Page

ORACLE® Database Express Edition

User: HR

Home

Administration Object Browser SQL Utilities Application Builder

```
CREATE TABLE pais
(
    cd_pais number (2),
    nm_pais char (45),
    ds_nacionalidade char(35),
    constraintpais_cd_pais_pk primary key (cd_pais),
    constraintpais_nm_pais_ukunique(nm_pais)
)
```

Inserindo registros nas tabelas

Sintaxe:

```
INSERT INTO nome_da_tabela
[(nome_da_coluna[, nome_da_coluna])]
VALUES
(valor_da_coluna[, valor_da_coluna])
```

Valores alfanuméricos e datas deverão estar entre aspas simples ('...')

- Ex 1:
INSERT INTO pais (cd_pais, nm_pais)VALUES (1,'Brasil')
- Ex 2:
INSERT INTO pais (cd_pais, nm_pais, ds_nacionalidade)VALUES (2, 'Argentina',null);
- Ex 3:
INSERT INTO paisVALUES (4, 'Chile', 'Chilena');
- Ex 4:
INSERT INTO pais (nm_pais, ds_nacionalidade, cd_pais)VALUES ('EUA', 'Americana' ,3);

Para incluir dados em uma tabela, pode-se, ou não, incluir valores em todas as suas colunas (caso não sejam obrigatórias).

A relação de colunas da tabela poderá ser omitida quando forem inclusos valores para todas as colunas dessa tabela.

UPDATE



Modificar a(s) linha(s) existentes com o comando UPDATE.

```
UPDATETabela
SET COLUNA = Valor
WHERE Condição
```

Update Paisset Nm_pais='Portugal',Ds_nacionalidade='Portug'
WhereCd_pais = 3;

Update Paisset Ds_nacionalidade='Portuguesa'
WhereCd_pais = 3;

DELETE



Remover a(s) linha(s) existente(s) com o comando DELETE.

```
DELETE FROM Tabela  
WHERE Condição
```



Observações:

- Verifique os nomes das colunas com o comando DESCRIBE.
- Confirme a operação de atualização pela visualização das linhas a serem atualizadas com o comando SELECT.
- Nunca omita a cláusula WHERE. No caso da omissão, todos os registros da tabela serão eliminados.

DELETE FROM Pais WHERE Cd_pais = 3;

RENAME



Este comando é utilizado nos casos de alteração de nome das tabelas.

```
RENAME nome_antigo_tabela TO nome_novo_tabela
```

RENAME Pais TO Tb_Pais

CREATE TABLE AS SELECT



Este comando cria uma tabela (estrutura e dados) com base em um comando SELECT.

```
create table espelhoteste as select * from employees
```

Results Explain Describe Saved SQL History					Results Explain Describe Saved SQL History				
EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER
100	Steven	King	SKING	515.123.4567	100	Steven	King	SKING	515.123.4567
101	Neena	Kochhar	NKOCHHAR	515.123.4568	101	Neena	Kochhar	NKOCHHAR	515.123.4568
102	Lex	De Haan	LDEHAAN	515.123.4569	102	Lex	De Haan	LDEHAAN	515.123.4569
103	Alexander	Hunold	AHUNOLD	590.423.4567	103	Alexander	Hunold	AHUNOLD	590.423.4567
104	Bruce	Ernst	BERNST	590.423.4568	104	Bruce	Ernst	BERNST	590.423.4568
105	David	Austin	DAUSTIN	590.423.4569	105	David	Austin	DAUSTIN	590.423.4569
106	Valli	Pataballa	VPATABAL	590.423.4560	106	Valli	Pataballa	VPATABAL	590.423.4560
107	Diana	Lorentz	DLORENTZ	590.423.5567	107	Diana	Lorentz	DLORENTZ	590.423.5567
108	Nancy	Greenberg	NGREENBE	515.124.4569	108	Nancy	Greenberg	NGREENBE	515.124.4569
109	Daniel	Faviet	DFAVIET	515.124.4169	109	Daniel	Faviet	DFAVIET	515.124.4169
More than 10 rows available. Increase rows selector to view more rows.					More than 10 rows available. Increase rows selector to view more rows.				

SELECT



O comando SELECT permite a seleção e a manipulação para visualização das informações armazenadas no banco de dados.

O comando SELECT permite selecionar:

(Para selecionar essas tabelas, é preciso logar-se com o usuário HR no Oracle)

- A) Todas as colunas

SELECT * FROM departments;

- B) Colunas específicas

SELECT FIRST_NAME, EMAIL, HIRE_DATE FROM employees;

- C) Expressões Aritméticas

SELECT FIRST_NAME, EMAIL, salary, salary*12 FROM employees;

- D) Colunas com Apelidos(alias)

SELECT FIRST_NAME, EMAIL, salary "Salário", salary*12 "SalarioAnual" FROM employees;

- E) Colunas concatenadas

SELECT FIRST_NAME, EMAIL, 'Salário: ' || salary "Salário", salary*12 "SalarioAnual" FROM employees;

Linhas Duplicadas

A exibição default de consultas são todas as linhas, incluindo as linhas duplicadas.

Para eliminar valores duplicados no resultado, incluímos o DISTINCT qualificador no comando SELECT.

```
SELECT distinct department_id FROM employees
ORDER BY department_id;
```

Cláusula WHERE



A Cláusula WHERE indica condição para um SELECT, o qual pode ter os seguintes operadores:

Operador	Descrição
=	Igual
<>	Diferente
>	Maior
<	Menor
>=	Maior ou igual
<=	Menor ou igual

Alfanuméricos e Datas na cláusula WHERE devem estar entre aspas simples.

Operadores SQL

Existem quatro operadores SQL os quais opera, com todos tipos de dados:

Operador	Significado
Between ...and ...	Entre dois valores (inclusive)
In (lista)	Comparar uma lista de valores
Like	Compara um parâmetro alfanumérico
Is Null	É um valor nulo

O Operador BETWEEN

Compara uma faixa de valores, inclusive o menor e o maior valor.

Suponhamos que nós queiramos ver aqueles empregados cujos salários estejam entre 1000 e 2000:

```
SELECT First_name, Last_name, Salary
FROM Employees
WHERE Salary BETWEEN 10000 AND 20000;
```

FIRST_NAME	LAST_NAME	SALARY
Neena	Kochhar	17000
Lex	De Haan	17000
Nancy	Greenberg	12000
Den	Raphaely	11000
John	Russell	14000
Karen	Partners	13500
Alberto	Errazuriz	12000
Gerald	Cambrault	11000
Eleni	Zlotkey	10500
Peter	Tucker	10000
More than 10 rows available. Increase rows selector to view more rows.		

Note que inclusive os valores especificados estão no resultado, e o menor precisa ser especificado primeiro.

O Operador IN

Compara os valores especificados dentro de uma lista.

Para encontrar empregados que tenham um dos três números de Manager, utilize o seguinte comando:

```
SELECT First_name, Last_name, Salary, Manager_id
FROM Employees
WHERE Manager_id IN (101, 145, 149);
```

FIRST_NAME	LAST_NAME	SALARY	MANAGER_ID
Nancy	Greenberg	12000	101
Jennifer	Whalen	4400	101
Susan	Mavris	6500	101
Hermann	Baer	10000	101
Shelley	Higgins	12000	101
Peter	Tucker	10000	145
David	Bernstein	9500	145
Peter	Hall	9000	145
Christopher	Olsen	8000	145
Nanette	Cambrault	7500	145
Oliver	Tuvault	7000	145
Ellen	Abel	11000	149
Alyssa	Hutton	8800	149
Jonathon	Taylor	8600	149
Jack	Livingston	8400	149
Kimberely	Grant	7000	149
Charles	Johnson	6200	149

Se alfanuméricos ou datas forem usados na lista, precisam ser colocados entre aspas simples ('...').

O Operador LIKE

Algumas vezes você precisa procurar valores que não conhece exatamente. Usando o operador LIKE, é possível selecionar linhas combinando parâmetros alfanuméricos. O caractere % é utilizado como coringa nas pesquisas. Veja exemplo abaixo:

Sintaxe	Descrição
Like '%cadeia'	Localiza valores que terminem com a cadeia de caracteres.
Like 'cadeia%'	Localiza valores que comecem com a cadeia de caracteres.
Like '%cadeia%'	Localiza valores que tenham a cadeia de caracteres em qualquer parte do campo.

O Operador LIKE

Para listar todos os empregados que tenham o nome que comecem com a letra S, faça:

```
SELECT First_name, Last_name, Salary
FROM Employees
WHERE First_name like 'S%';
```

O Operador IS NULL

Verifica quais campos estão com valores nulos.

Unicamente encontrar todos os empregados que não tenham gerente, você testará um valor nulo:

```
SELECT First_name, Last_name, Salary, Manager_id
FROM Employees
WHERE Manager_id IS Null;
```

Expressões Negativas

Podemos também negar todos os operadores estudados:

Sintaxe	Descrição
NOT Between	Tudo que estiver fora da faixa.
NOT In	Tudo que não estiver na lista.
NOT Like	Tudo que não contiver a linha de caracteres.
IS NOT Null	Tudo que não for nulo.

```
SELECT First_name, Last_name, Salary
FROM Employees
WHERE Salary NOT BETWEEN 10000 AND 20000;
```

```
SELECT First_name, Last_name, Salary, Manager_id  
FROM Employees  
WHERE Manager_id NOT IN (101, 145, 149);
```

```
SELECT First_name, Last_name, Salary  
FROM Employees  
WHERE First_name NOT Like 'S%';
```

```
SELECT First_name, Last_name, Salary, Manager_id  
FROM Employees  
WHERE Manager_id IS NOT Null;
```

Pesquisando dados com múltiplas condições

- Os operadores AND e OR devem ser usados para fazer composições de expressões lógicas.
- O predicado AND esperará que ambas as condições sejam verdadeiras;
- O predicado OR esperará que uma das condições seja verdadeira.

Nos dois exemplos seguintes, as condições são as mesmas, mas o predicado é diferente.

Veja como o resultado é dramaticamente alterado.

Para encontrar todos os escriturários do departamento de estoque que ganhem entre 2000 e 3000.

```
SELECT First_name, Last_name, Salary, Job_ID  
FROM Employees  
WHERE Salary BETWEEN 2000 AND 3000  
AND Job_id = 'ST_CLERK';
```

FIRST_NAME	LAST_NAME	SALARY	JOB_ID
Irene	Mikkilineni	2700	ST_CLERK
James	Landry	2400	ST_CLERK
Steven	Markle	2200	ST_CLERK
Mozhe	Atkinson	2800	ST_CLERK
James	Marlow	2500	ST_CLERK
TJ	Olson	2100	ST_CLERK
Michael	Rogers	2900	ST_CLERK
Ki	Gee	2400	ST_CLERK
Hazel	Philtanker	2200	ST_CLERK
John	Seo	2700	ST_CLERK
Joshua	Patel	2500	ST_CLERK
Randall	Matos	2600	ST_CLERK
Peter	Vargas	2500	ST_CLERK

Para encontrar todos os escriturários do departamento de estoque e os funcionários que ganhem entre 2000 e 3000.

```
SELECT First_name, Last_name, Salary
FROM Employees
WHERE Salary BETWEEN 2000 AND 3000
OR Job_id = 'ST_CLERK';
```

Perceba que a seleção foi bem maior, visto que existem funcionários que ganham entre 2000 e 3000 e não são escriturários do departamento de estoque.

FIRST_NAME	LAST_NAME	SALARY	JOB_ID
Shelli	Baida	2900	PU_CLERK
Sigal	Tobias	2800	PU_CLERK
Guy	Himuro	2600	PU_CLERK
Karen	Colmenares	2500	PU_CLERK
Julia	Nayer	3200	ST_CLERK
Irene	Mikkilineni	2700	ST_CLERK
James	Landry	2400	ST_CLERK
Steven	Markle	2200	ST_CLERK
Laura	Bissot	3300	ST_CLERK
Mozhe	Atkinson	2800	ST_CLERK
James	Marlow	2500	ST_CLERK
TJ	Olson	2100	ST_CLERK
Jason	Mallin	3300	ST_CLERK
Michael	Rogers	2900	ST_CLERK
Ki	Gee	2400	ST_CLERK
Hazel	Philtanker	2200	ST_CLERK
Renske	Ladwig	3600	ST_CLERK
Stephen	Stiles	3200	ST_CLERK
John	Seo	2700	ST_CLERK
Joshua	Patel	2500	ST_CLERK
Trenna	Rajs	3500	ST_CLERK
Curtis	Davies	3100	ST_CLERK
Randall	Matos	2600	ST_CLERK
Peter	Vargas	2500	ST_CLERK
Martha	Sullivan	2500	SH_CLERK
Girard	Geoni	2800	SH_CLERK
Anthony	Cabrio	3000	SH_CLERK
Timothy	Gates	2900	SH_CLERK
Randall	Perkins	2500	SH_CLERK
Vance	Jones	2800	SH_CLERK
Kevin	Feehey	3000	SH_CLERK

Você pode combinar AND e OR na mesma expressão lógica.

Quando AND e OR aparecerem na mesma cláusula WHERE, todos os ANDs serão feitos primeiro e, posteriormente, todos os “ORs” serão feitos.

```
SELECT First_name, Last_name, Salary, Job_ID  
FROM Employees  
WHERE Salary > 8000 and  
Job_id = 'ST_CLERK' OR  
Job_id = 'ST_MAN'
```

Se você quiser selecionar todos os Escriturários e Gerentes que ganhem acima de 8000, deverá fazer:

```
SELECT First_name, Last_name, Salary, Job_ID  
FROM Employees  
WHERE Salary > 8000 and  
(Job_id = 'ST_CLERK' OR  
Job_id= 'ST_MAN')
```

Os parênteses especificam a ordem na qual os operadores devem ser avaliados (prioridade).

No segundo exemplo, o operador OR é avaliado antes do AND.

Sempre que você tiver dúvida sobre qual dos dois operadores será feito primeiro, quando a expressão é avaliada, use sempre parênteses para definir a prioridade das expressões.

A cláusula ORDER BY

Normalmente a ordem das linhas retornadas de uma pesquisa é indefinida. A cláusula ORDER BY pode ser usada para ordenar as linhas.

Se usado, o ORDER BY deve ser sempre a última cláusula da declaração SELECT. Para ordenar essa consulta pelo campo Last_Name, faça:

```
SELECT First_name, Last_name, Salary  
FROM Employees  
WHERE Salary BETWEEN 10000 AND 20000  
ORDER BY Last_name;
```

Padrão da Ordenação dos Dados

O padrão da ordem de ordenação é ascendente:

- valores numéricos infinitos primeiro;
- valores de data primeiro;
- valores alfanuméricos.

Invertendo o padrão de ordenação

Para inverter essa ordem, acrescente o comando DESC (Decrescente) depois do nome das colunas da cláusula ORDER BY.

Para inverter a ordem da coluna Last_name, faça:

```
SELECT      First_name, Last_name, Salary
FROM        Employees
WHERE       Salary BETWEEN 10000 AND 20000
ORDER BY    Last_name DESC;
```

Ordenação por várias colunas.

É possível utilizar mais de uma coluna na cláusula ORDER BY.

O limite de colunas é o número de colunas da tabela. Na cláusula ORDER BY, especifiquem-se as colunas que serão ordenadas, separando-as por vírgula. Se algumas ou todas forem invertidas, especifique DESC depois de cada uma das colunas.

Para ordenar por duas colunas e mostrar ordem inversa do salário e ordem crescente do sobrenome faça:

```
SELECT      First_name, Last_name, Salary
FROM        Employees
WHERE       Salary BETWEEN 10000 AND 20000
ORDER BY    Salary Desc, Last_name;
```

Para ordenar por uma coluna, ela não precisa, necessariamente, estar declarada no SELECT.

A cláusula ORDER BY é usada, na pesquisa, quando você quer mostrar as linhas em uma ordem específica. Sem a cláusula ORDER BY, as linhas são retornadas, na ordem conveniente, para o ORACLE.

Esse comando não altera a ordem dos dados que estão armazenados no Banco de Dados.

Material Complementar

Consultando este material complementar, você verá que os mesmos comandos que aprendemos no Oracle podem ser utilizados em outros SGDB.

- <http://youtu.be/TBjsSrjuLgU>.

Referências

COSTA, Rogério Luis de C. **SQL : guia prático. 2.** ed. Rio de Janeiro: Brasport, 2006.

FANDERUFF, Damaris. **Dominando o Oracle 9i: Modelagem e desenvolvimento.** São Paulo: Pearson Education do Brasil, 2003.

MORELLI, Eduardo M. **Terra, 1996. Oracle 9i Fundamental: Sql, Pl/SQL e Administração.** São Paulo: Érica, 2002.

SILBERSCHATZ, A. **Sistema de bancos de dados.** São Paulo: Pearson Education do Brasil, 2004.

Anotações





Educação a Distância
Cruzeiro do Sul Educacional
Campus Virtual

www.cruzeirodosulvirtual.com.br
Campus Liberdade
Rua Galvão Bueno, 868
CEP 01506-000
São Paulo SP Brasil
Tel: (55 11) 3385-3000



Universidade
Cruzeiro do Sul



UNICID
Universidade
Cidade de S. Paulo



UNIFRAN
Universidade
de Franca



UDF
Centro
Universitário



Módulo
Centro
Universitário