



## Bases de Dados

ORACLE®



## SQL- Introdução

ORACLE®

## SELECT do SQL

Fernanda Brito Correia - DEIS/ISEC/IPC - BD - 18/19

### Seleccção

Tabela 1

### Projecção

Tabela 1

### Join

Tabela 1

Tabela 2



## Base de dados - Tabela DEP

### ▶ DEP (PK: NDEP)

NDEP    NUMBER(2) not null  
NOME    VARCHAR2(15) null  
LOC    VARCHAR2(15) null

Fernanda Brito Correia - DEIS/ISEC/IPC - BD - 18/19

## Base de dados - Tabela EMP

- ▶ EMP (PK: NEMP; NDEP FK DEP; ENCAR FK EMP)

|              |                    |
|--------------|--------------------|
| NEMP         | NUMBER(4) not null |
| NDEP         | NUMBER(2) not null |
| ENCAR        | NUMBER(4) null     |
| NOME         | VARCHAR2(20) null  |
| FUNCAO       | VARCHAR2(12) null  |
| DATA_ENTRADA | DATE null          |
| SAL          | NUMBER(7) null     |
| PREMIOS      | NUMBER(7) null     |

## Base de dados - Tabela DESCONTOS

- ▶ DESCONTOS (PK: ESCALAO)

|         |                    |
|---------|--------------------|
| ESCALAO | NUMBER(2) not null |
| SALINF  | NUMBER(7) null     |
| SALSUP  | NUMBER(7) null     |

- ▶ A relação entre empregados e descontos é calculada através do valor do salário do empregado.
- ▶ Assim, diz-se que um empregado, de número NEMP, pertence a um escalão de número ESCALAO se o seu valor de salário se situar entre os valores mínimo e máximo desse escalão.

## Instrução SELECT Básica

```
SELECT    [DISTINCT] {*, column [alias],...}  
FROM      table;
```

- ▶ SELECT identifica *quais* as colunas.
- ▶ FROM identifica *qual* a tabela.



## Escrever Instruções SQL

- ▶ As instruções SQL não são case sensitive.
- ▶ As instruções SQL podem ser numa ou mais linhas.
- ▶ As palavras chave não podem ser abreviadas ou serem divididas através das linhas.
- ▶ As clausulas são normalmente colocadas em linhas diferentes.
- ▶ Tabs e indentação são usados para maior legibilidade.

## Seleccionar as Colunas Todas

```
SQL> SELECT *  
2 FROM dep;
```

| NDEP | NOME          | LOC      |
|------|---------------|----------|
| 10   | CONTABILIDADE | NEW YORK |
| 20   | INVESTIGACAO  | DALLAS   |
| 30   | VENDAS        | CHICAGO  |
| 40   | OPERACOES     | BOSTON   |



## Seleccionar colunas específicas

```
SQL> SELECT ndep, loc  
2 FROM dep;
```

| NDEP | LOC      |
|------|----------|
| 10   | NEW YORK |
| 20   | DALLAS   |
| 30   | CHICAGO  |
| 40   | BOSTON   |



## Expressões Aritméticas

| Operador | Descrição     |
|----------|---------------|
| +        | Adição        |
| -        | Subtracção    |
| *        | Multiplicação |
| /        | Divisão       |



## Usar Operadores aritméticos

```
SQL> SELECT nome, sal, sal+300
2 FROM emp;
```

| NOME              | SAL   | SAL+300 |
|-------------------|-------|---------|
| -----             | ----- | -----   |
| KING              | 5000  | 5300    |
| BLAKE             | 2850  | 3150    |
| CLARK             | 2450  | 2750    |
| JONES             | 2975  | 3275    |
| MARTIN            | 1250  | 1550    |
| ALLEN             | 1600  | 1900    |
| ...               |       |         |
| 14 rows selected. |       |         |



## Precedência dos Operadores



- ▶ Multiplicação e divisão têm prioridade sobre a adição e subtração.
- ▶ Operadores com a mesma prioridade são calculados da esquerda para a direita.
- ▶ Parênteses são usados para forçar o cálculo da prioridade e para clarificar as instruções.



## Precedência dos Operadores

```
SQL> SELECT nome, sal, 12*sal+100
2 FROM emp;
```

| NOME   | SAL  | 12*SAL+100 |
|--------|------|------------|
| KING   | 5000 | 60100      |
| BLAKE  | 2850 | 34300      |
| CLARK  | 2450 | 29500      |
| JONES  | 2975 | 35800      |
| MARTIN | 1250 | 15100      |
| ALLEN  | 1600 | 19300      |
| ...    |      |            |

14 rows selected.



## Usar Parênteses

```
SQL> SELECT nome, sal, 12*(sal+100)
2 FROM emp;
```

| NOME   | SAL  | 12*(SAL+100) |
|--------|------|--------------|
| KING   | 5000 | 61200        |
| BLAKE  | 2850 | 35400        |
| CLARK  | 2450 | 30600        |
| JONES  | 2975 | 36900        |
| MARTIN | 1250 | 16200        |
| ...    |      |              |

14 rows selected.



## Definir um Valor Null

- Um null é um valor que está indisponível, não atribuído, desconhecido ou não aplicável.
- Um null não é o mesmo que um zero ou um espaço em branco.

```
SQL> SELECT nome, funcao, sal, premios
2 FROM emp;
```

| NOME   | FUNCAO     | SAL  | PREMIOS |
|--------|------------|------|---------|
| KING   | PRESIDENTE | 5000 |         |
| BLAKE  | GESTOR     | 2850 |         |
| ...    |            |      |         |
| TURNER | VENDEDOR   | 1500 | 0       |
| ...    |            |      |         |

14 rows selected.





## Valores Null em Expressões Aritméticas

- Expressões aritméticas contendo um valor null calculam null.

```
SQL> select nome, 12*sal+premios
2   from emp
3  WHERE nome='KING';
```

| MOME  | 12*SAL+PREMIOS |
|-------|----------------|
| ----- | -----          |
| KING  |                |



## Definir um Alias para uma Coluna

- Renomeia o cabeçalho de uma coluna
- É útil nos cálculos
- Segue imediatamente o nome da coluna; a palavra chave AS entre o nome da coluna e o alias é opcional
- Requer aspas se contém espaços ou caracteres especiais ou é case sensitive



## Usar Aliases em Colunas

```
SQL> SELECT nome AS ident, sal salario
2 FROM emp;
```

| IDENT | SALARIO |
|-------|---------|
| ----- |         |
| ...   |         |

```
SQL> SELECT nome "Nome",
2 sal*12 "Salario Anual"
3 FROM emp;
```

| Nome  | Salario Anual |
|-------|---------------|
| ----- |               |
| ...   |               |



## Operador de Concatenação

- ▶ Concatena colunas ou strings de caracteres a outras colunas
- ▶ É representado por duas barras verticais (||)
- ▶ Cria uma coluna resultante que é uma expressão de caracter



## Usar o Operador de Concatenação

```
SQL> SELECT  nome || funcao AS "Empregados"
2 FROM      emp;
```

```
Empregados
-----
KINGPRESIDENTE
BLAKEGESTOR
CLARKGESTOR
JONESGESTOR
MARTINVENDEDOR
ALLENVENDEDOR
...
14 rows selected.
```



## Strings de caracteres Literais

- ▶ Um literal é um caracter, um número, ou uma data incluída na lista do SELECT.
- ▶ Valores de data e caracteres literais devem ser incluídos dentro de plicas.
- ▶ Cada string de caracteres é escrita uma vez para cada linha retornada.



## Usar Strings de Caracter Literais

```
SQL> SELECT nome || ' e um ' || funcao
2          AS "Detalhes do Empregado"
3 FROM emp;
```

Detalhes do Empregado

```
-----
KING e um PRESIDENTE
BLAKE e um GESTOR
CLARK e um GESTOR
JONES e um GESTOR
MARTIN e um VENDEDOR
...
14 rows selected.
```



## Linhas Duplicadas

- ▶ O display default das queries é todas as linhas, incluindo linhas duplicadas.

```
SQL> SELECT ndep
2 FROM emp;
```

NDEP

```
-----
10
30
10
20
...
14 rows selected.
```



## Eliminar Linhas Duplicadas

Elimina linhas duplicadas usando a palavra chave **DISTINCT** na cláusula **SELECT**.

```
SQL> SELECT DISTINCT ndep  
2 FROM emp;
```

| NDEP |
|------|
| 10   |
| 20   |
| 30   |