



# Programação de Banco de Dados com SQL

1-3

Anatomia de uma Instrução SQL



# Objetivos

Esta lição abrange os seguintes objetivos:

- Relacionar projeção e seleção com seus recursos corretos
- Criar uma instrução SELECT básica
- Usar a sintaxe correta para exibir todas as linhas em uma tabela
- Usar a sintaxe correta para selecionar colunas específicas em uma tabela, modificar a maneira como os dados são exibidos e fazer cálculos usando expressões e operadores aritméticos

# Objetivos

Esta lição abrange os seguintes objetivos:

- Formular consultas usando a precedência de operadores correta para exibir os resultados desejados
- Definir um valor nulo
- Demonstrar o efeito que os valores nulos criam em expressões aritméticas
- Construir uma consulta usando um alias de coluna

# Palavra-chave SELECT

- SELECT é uma das palavras-chave mais importantes da linguagem SQL (se não for a mais importante).
- Usa-se SELECT para recuperar informações do banco de dados. Quando aprender a usar essa palavra-chave, você terá aberto a porta para o banco de dados.
- Imagine um banco de dados contendo informações sobre filmes, como título, gênero, estúdio, produtor, data de lançamento, sequências, país, idioma, avaliação, tempo de exibição e assim por diante.
- E se você quisesse apenas os títulos de filmes feitos na Índia?
- A instrução SELECT permite pesquisar dados específicos.

**ORACLE®**

# Instrução SELECT

- A instrução SELECT recupera informações do banco de dados.
- A sintaxe dessa instrução é a seguinte:

```
SELECT <column_name(s)>  
FROM <table_name>;
```

- Em sua forma mais simples, ela deve incluir o seguinte:
- Uma cláusula SELECT, que especifica as colunas a serem exibidas
- Uma cláusula FROM, que especifica a tabela que contém as colunas listadas na cláusula SELECT

# Convenções

- Ao longo deste curso, será usado o seguinte:

```
SELECT last_name  
FROM employees;
```

- Uma palavra-chave refere-se a um comando SQL individual.
- Por exemplo, SELECT e FROM são palavras-chave.
- Uma cláusula é parte de uma instrução SQL.
- Por exemplo, SELECT last\_name é uma cláusula.
- Uma instrução é uma combinação de duas ou mais cláusulas.
- Por exemplo, SELECT last\_name FROM employees é uma instrução.

# Recursos das Instruções SELECT

- Projeção: usada para escolher colunas em uma tabela
- Seleção: usada para escolher linhas em uma tabela

Tabela 2: Projeção


Tabela 2: Seleção




# Projeção e Seleção

ID	FIRST_NAME	LAST_NAME	SALARY
10	John	Doe	4000
20	Jane	Jones	3000
30	Sylvia	Smith	5000
40	Hai	Nguyen	6000

Projeção

Seleção

```
SELECT salary
FROM employees
WHERE last_name like 'Smith';
```

# Selecionando Todas as Colunas

- Você pode exibir todas as colunas de dados em uma tabela usando um asterisco (\*), em vez do nome de uma coluna, na cláusula SELECT.
- No exemplo mostrado, todas as colunas na tabela de países são selecionadas.

```
SELECT *  
FROM countries;
```

COUNTRY_ID	COUNTRY_NAME	REGION_ID
CA	Canada	2
DE	Germany	1
UK	United Kingdom	1
US	United States of America	2

# Selecionando Todas as Colunas

- Você também pode listar as colunas de uma tabela individualmente para exibir todas elas.

```
SELECT country_id, country_name, region_id  
FROM countries;
```

COUNTRY_ID	COUNTRY_NAME	REGION_ID
CA	Canada	2
DE	Germany	1
UK	United Kingdom	1
US	United States of America	2

# Projetando Colunas Específicas

- Se você quiser PROJETAR somente colunas específicas de uma tabela para serem exibidas, basta listar o nome de cada coluna desejada e separá-los com uma vírgula na cláusula SELECT.

```
SELECT location_id, city, state_province  
FROM locations;
```

LOCATION_ID	CITY	STATE_PROVINCE
1800	Toronto	Ontario
2500	Oxford	Oxford
1400	Southlake	Texas
1500	South San Francisco	California
1700	Seattle	Washington

# Usando Operadores Aritméticos

- Usando algumas regras e diretrizes simples, você pode construir instruções SQL que sejam fáceis de ler e editar.
- Conhecer as regras tornará o aprendizado de SQL mais fácil.
- É possível que você precise modificar a maneira como os dados são exibidos, fazer cálculos ou observar cenários hipotéticos.
- Por exemplo: "E se todos os funcionários recebessem 5% de aumento?"
- Como isso afetaria o nosso lucro anual?"

**ORACLE®**

Academy

# Usando Operadores Aritméticos

- Esse tipo de cálculo é possível com o uso de expressões aritméticas.
- Você já conhece as expressões aritméticas da matemática:
  - adição (+), subtração (-) , multiplicação (\*) e divisão (/)
- Observe que o exemplo não cria novas colunas nas tabelas nem altera os valores de dados reais.
- Os resultados dos cálculos vão aparecer somente na saída.

# Usando Operadores Aritméticos

- O exemplo mostrado usa o operador de adição para calcular um aumento de salário de 300 para todos os funcionários e exibe uma nova coluna, **SALÁRIO + 300**, na saída.

```
SELECT last_name, salary, salary + 300  
FROM employees;
```

- Colocar espaços em branco antes e depois de um operador aritmético não afetará a saída.

LAST_NAME	SALARY	SALARY+300
King	24000	24300
Kochhar	17000	17300
De Haan	17000	17300
Whalen	4400	4700
Higgins	12000	12300
Gietz	8300	8600
Zlotkey	10500	10800
Abel	11000	11300
Taylor	8600	8900
Grant	7000	7300

# Precedência dos Operadores Aritméticos

- Precedência é a ordem em que o Oracle avalia operadores diferentes na mesma expressão.
- Em uma expressão que contém vários operadores, ele avalia primeiro os que têm maior precedência.
- O Oracle avalia os operadores com precedência equivalente da esquerda para a direita em uma expressão.



# Precedência dos Operadores Aritméticos

- Os operadores aritméticos executam as operações matemáticas de Multiplicação, Divisão, Adição e Subtração.
- Se esses operadores aparecerem juntos em uma expressão, a multiplicação e divisão são avaliadas primeiro.
- Portanto, a ordem é:  $*$  /  $+$  -.
- Para lembrar facilmente a precedência dos operadores, use o processo mnemônico: MDAS

# Precedência dos Operadores Aritméticos

- Se os operadores de uma expressão tiverem a mesma prioridade, a avaliação será feita da esquerda para a direita.
- Você pode usar parênteses para forçar a expressão entre eles a ser avaliada primeiro.
- Nas tabelas de exemplo mostradas no próximo slide, observe as diferenças na saída entre a consulta que usou parênteses e a que não usou.

# Precedência dos Operadores Aritméticos

## Precedência dos Operadores

```
SELECT last_name, salary, 12*salary +100
FROM employees;
```

LAST_NAME	SALARY	12*SALARY+100
King	24000	288100
Kochhar	17000	204100
De Haan	17000	204100
Whalen	4400	52900
Higgins	12000	144100
Gietz	8300	99700

## Usando Parênteses

```
SELECT last_name, salary, 12*(salary +100)
FROM employees;
```

LAST_NAME	SALARY	12*(SALARY+100)
King	24000	289200
Kochhar	17000	205200
De Haan	17000	205200
Whalen	4400	54000
Higgins	12000	145200
Gietz	8300	100800



# Valores NULOS

- Em SQL, NULO é uma palavra interessante.
- Para entendê-la, você deve saber o que é e não é NULO.
- NULO é o valor que está indisponível, não foi atribuído, é desconhecido ou é inaplicável.
- NULO não é o mesmo que um zero ou espaço.
- Em SQL, zero é um número e espaço é um caractere.

# Valores NULOS

- Às vezes, você não sabe o valor de uma coluna.
- Em um banco de dados, é possível armazenar valores desconhecidos.
- Para representá-los, os bancos de dados relacionais usam um espaço reservado chamado NULO ou nulo.

# Valores NULOS

- Se qualquer valor de coluna em uma expressão aritmética for nulo, o resultado será nulo ou desconhecido.
- Se você tentar dividir por nulo, o resultado será nulo ou desconhecido.
- No entanto, se tentar dividir por zero, receberá um erro.

Salaries and Commissions

LAST_NAME	JOB_ID	SALARY	COMMISSION_PCT
King	AD_PRES	24000	-
Kochhar	AD_VP	17000	-
De Haan	AD_VP	17000	-
Whalen	AD_ASST	4400	-
Higgins	AC_MGR	12000	-
Gietz	AC_ACCOUNT	8300	-
Zlotkey	SA_MAN	10500	.2
Abel	SA_REP	11000	.3

# Valores NULOS

```
SELECT last_name, job_id, salary,  
commission_pct, salary*commission_pct  
FROM employees;
```

## Salaries and Commissions

LAST_NAME	JOB_ID	SALARY	COMMISSION_PCT	SALARY*COMMISSION_PCT
King	AD_PRES	24000	-	-
Kochhar	AD_VP	17000	-	-
De Haan	AD_VP	17000	-	-
Whalen	AD_ASST	4400	-	-
Higgins	AC_MGR	12000	-	-
Gietz	AC_ACCOUNT	8300	-	-
Zlotkey	SA_MAN	10500	.2	2100
Abel	SA_REP	11000	.3	3300
Taylor	SA_REP	8600	.2	1720

# Aliases

- Um Alias é uma maneira de renomear o título de uma coluna na saída.
- Sem os aliases, quando o resultado de uma instrução SQL for exibida, o nome das colunas exibidas será o mesmo que os nomes de coluna na tabela ou que um nome mostrando uma operação aritmética, como  $12 * (\text{SALARY} + 100)$ .
- Você provavelmente quer que a saída exiba um nome mais fácil de entender, mais "amigável".
- Os aliases de coluna permitem renomear colunas na saída.



# Aliases

- Várias regras se aplicam ao uso de aliases de coluna para formatar a saída.
- Um alias de coluna:
  - Renomeia um título de coluna
  - É útil para cálculos
  - Vem imediatamente após o nome da coluna
  - Pode ter a palavra-chave AS opcional entre o nome da coluna e o alias
  - Requer aspas duplas se o alias tiver espaços ou caracteres especiais ou fizer distinção entre letras maiúsculas e minúsculas

# Usando Aliases de Coluna

- A sintaxe para aliases é:

```
SELECT * |column|expr [ AS alias], .....  
FROM table;
```

- Exemplos:

```
SELECT last_name AS name,  
commission_pct AS comm  
FROM employees;
```

```
SELECT last_name "Name",  
salary*12 "Annual Salary"  
FROM employees;
```

NAME	COMM
King	-
Kochhar	-
De Haan	-

Name	Annual Salary
King	288000
Kochhar	204000
De Haan	204000

# Terminologia

Estes são os principais termos usados nesta lição:

- Expressão aritmética
- Operador aritmético
- Cláusula
- Coluna
- Alias de coluna
- Cláusula From
- NULO

# Terminologia

Estes são os principais termos usados nesta lição:

- Projeção
- Cláusula Select
- Seleção
- Instrução Select
- Instrução
- Cláusula WHERE
- \* (Asterisco)

# Resumo

Nesta lição, você deverá ter aprendido a:

- Relacionar projeção e seleção com seus recursos corretos
- Criar uma instrução SELECT básica
- Usar a sintaxe correta para exibir todas as linhas em uma tabela
- Usar a sintaxe correta para selecionar colunas específicas em uma tabela, modificar a maneira como os dados são exibidos e fazer cálculos usando expressões e operadores aritméticos

# Resumo

Nesta lição, você deverá ter aprendido a:

- Formular consultas usando a precedência de operadores correta para exibir os resultados desejados
- Definir um valor nulo
- Demonstrar o efeito que os valores nulos criam em expressões aritméticas
- Construir uma consulta usando um alias de coluna

