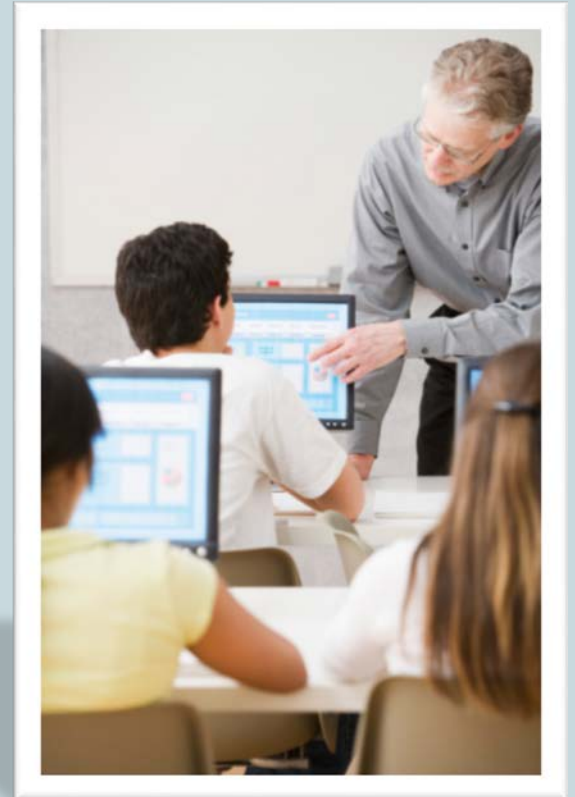




# Programação de Banco de Dados com SQL

5-3

Expressões de Condição



# Objetivos

Esta lição abrange os seguintes objetivos:

- Comparar e contrastar as funções DECODE e CASE
- Construir e executar uma consulta SQL que use corretamente as funções DECODE e CASE
- Construir e executar dois métodos para implementar a lógica condicional IF-THEN-ELSE

# Finalidade

- Ser capaz de tomar decisões é essencial na modelagem de dados.
- Os modeladores precisam decidir quais funções de negócios precisam ser modeladas ou não.
- O processo de modelagem de dados requer que os designers analisem informações para identificar entidades, resolver relacionamentos e selecionar atributos.
- Um decisão típica poderia ser:
  - SE uma empresa precisar rastrear dados com o passar do tempo, ENTÃO o tempo terá de ser uma entidade.  
DO CONTRÁRIO, deverá ser um atributo.

# Como as Funções são Avaliadas

- Em programação, esse processo de tomada de decisões não é muito diferente daquele que usamos na vida real.
- Pense na última vez que você teve que tomar uma decisão como a do exemplo anterior.
- SE eu terminar minha lição de casa antes das 21 horas, ENTÃO vou poder assistir televisão. DO CONTRÁRIO, não vou poder assistir televisão.
- Em SQL, esses tipos de escolhas envolvem métodos de processamento condicional.
- É mais fácil tomar decisões para obter os dados desejados quando se sabe usar o processamento condicional.

# Expressões de Condição

- As duas expressões de condição são CASE e DECODE.
- Você já estudou NULLIF, que é logicamente equivalente à expressão CASE, pois ela compara duas expressões.
- NULLIF compara duas expressões e, se elas forem iguais, retorna um valor nulo; se não forem iguais, retorna a primeira expressão.

# Expressões de Condição

- Existem dois conjuntos de comandos ou sintaxes que podem ser usados para gravar instruções SQL:
  - Instruções em conformidade com o padrão ANSI/ISO SQL 99
  - Instruções proprietárias da Oracle
- Os dois conjuntos de sintaxes são muito semelhantes, mas existem algumas diferenças.
- Neste curso, você aprenderá a usar os dois conjuntos de instruções SQL, mas recomenda-se a sintaxe do padrão ANSI/ISO SQL 99.

# Expressões de Condição

- CASE e DECODE são exemplos de algumas das diferenças.
- CASE é uma instrução que cumpre o padrão ANSI/ISO SQL 99.
- DECODE é uma instrução proprietária da Oracle.
- Ambas as instruções retornam as mesmas informações usando sintaxes diferentes.



# Expressão CASE

- Basicamente, a expressão CASE faz o trabalho de uma instrução IF-THEN-ELSE.
- Os tipos de dados das expressões CASE, WHEN e ELSE devem ser os mesmos.
- A sintaxe de uma expressão CASE é:

```
CASE expr WHEN comparison_expr1 THEN return_expr1  
  [WHEN comparison_expr2 THEN return_expr2  
   WHEN comparison_exprn THEN return_exprn  
   ELSE else_expr]  
END
```

# Sintaxe CASE

- A consulta verifica department\_id.
  - SE for 90, então retorna 'Management'
  - SE for 80, então retorna 'Sales'
  - SE for 60, então retorna 'IT'
  - DO CONTRÁRIO, retorna 'Other dept.'

```
SELECT last_name,  
CASE department_id  
  WHEN 90 THEN 'Management'  
  WHEN 80 THEN 'Sales'  
  WHEN 60 THEN 'It'  
  ELSE 'Other dept.'  
END AS "Department"  
FROM employees;
```

LAST_NAME	Department
King	Management
Kochhar	Management
De Haan	Management
Whalen	Other dept.
Higgins	Other dept.
Gietz	Other dept.
Zlotkey	Sales
Abel	Sales
Taylor	Sales
Grant	Other dept.
Mourgos	Other dept.
Rajs	Other dept.
Davies	Other dept.
Matos	Other dept.
Vargas	Other dept.
Hunold	It
Ernst	It
Lorentz	It
Hartstein	Other dept.
Fay	Other dept.

# Expressão DECODE

- A função DECODE avalia uma expressão de maneira semelhante à lógica IF-THEN-ELSE.
- DECODE compara uma expressão a cada um dos valores de pesquisa.
- A sintaxe para DECODE é:

```
DECODE(column1|expression, search1, result1  
      [, search2, result2,...,]  
      [, default])
```

- Se o valor padrão for omitido, um valor nulo será retornado quando o valor de pesquisa não corresponder a um dos valores

# Expressão DECODE

- Examine o exemplo:

```
SELECT last_name,  
       DECODE(department_id,  
              90, 'Management',  
              80, 'Sales',  
              60, 'It',  
              'Other dept.')  
AS "Department"  
FROM employees;
```

- Essa consulta retorna exatamente os mesmos resultados que o exemplo anterior com CASE, mas usa uma sintaxe diferente.

LAST_NAME	Department
King	Management
Kochhar	Management
De Haan	Management
Whalen	Other dept.
Higgins	Other dept.
Gietz	Other dept.
Zlotkey	Sales
Abel	Sales
Taylor	Sales
Grant	Other dept.
Mourgos	Other dept.
Rajs	Other dept.
Davies	Other dept.
Matos	Other dept.
Vargas	Other dept.
Hunold	It
Ernst	It
Lorentz	It
Hartstein	Other dept.
Fay	Other dept.

# Terminologia

Estes são os principais termos usados nesta lição:

- CASE
- Expressão de condição
- DECODE

# Resumo

Nesta lição, você deverá ter aprendido a:

- Comparar e contrastar as funções DECODE e CASE
- Construir e executar uma consulta SQL que use corretamente as funções DECODE e CASE
- Construir e executar dois métodos para implementar a lógica condicional IF-THEN-ELSE

