

[< VOLTAR](#)

Criação de relatórios utilizando mais de uma tabela

Criar relatórios a partir de dados existentes no Banco de dados, utilizando mais de uma tabela a partir dos relacionamentos.

NESTE TÓPICO

[> Referências](#)[Marcar tópico](#)

Continuando a pesquisa em mais de uma tabela, conseguimos construir relatórios que buscam dados em diversas tabelas e por meio de uma das variações da junção – JOIN – INNER JOIN e OUTER JOIN.

Continuamos a utilizar essas consultas. Nosso trabalho agora é a utilização do SELF JOIN. Essa junção tem por objetivo realizar a pesquisa em um autorrelacionamento, que nada mais é que uma tabela relacionada a ela mesma.

Para exemplificar esse processo, vamos criar uma tabela que atenda a essa necessidade.

Criando a tabela empregado:

1. CREATE TABLE empregado
2. (cod_emp number(3) primary key, nome_emp varchar2(15) not null,
3. Cod_ger references empregado);

Dados da tabela empregado:

1. INSERT INTO empregado VALUES (1,'Marcos',6);
2. INSERT INTO empregado VALUES (2,'Lucas',1);
3. INSERT INTO empregado VALUES (3,'Ana',2);
4. INSERT INTO empregado VALUES (4,'Eva',1);
5. INSERT INTO empregado VALUES (5,'Juan',6);
6. INSERT INTO empregado VALUES (6,'Antonio',null);

Atualizando os dados dos gerentes:

1. UPDATE empregado SET cod_ger = 6 WHERE cod_emp IN (1,5);
2. UPDATE empregado SET cod_ger = 1 WHERE cod_emp IN (2,4);
3. UPDATE empregado SET cod_ger = 2 WHERE cod_emp = 3;

Preste atenção na instrução a seguir, ela irá verificar na tabela empregado o nome do empregado e quem é seu gerente.

Para entendermos melhor, vamos seguir passo a passo. Observe que esta instrução é a mais simples possível, e vai depender muito da leitura do relatório, pois ele não seria o ideal a ser realizado, como outros que já vimos.

1. SELECT * from empregado;



Perceba que é bem difícil a análise, agora se usarmos o conceito deste autorrelacionamento e aplicarmos o SELF JOIN:

1. SELECT E.Nome_emp Funcionario, Chefe.Nome_emp Gerente
2. FROM Empregado E, Empregado Chefe
3. WHERE E.cod_ger = Chefe.cod_emp;

Veja a diferença existente entre os dois relatórios.



Observe que o uso do SELF JOIN permite a busca do nome e não apenas ficar com o código.

Essa junção é uma técnica empregada para conseguir o produto cartesiano de uma tabela consigo mesma. Sua utilização não é muito frequente.

Mais Exemplos:

Suponha que a tabela a seguir já está criada e alimentada com os dados.

Autores	
Código (Código do livro)	Autor (Nome do Autor)
B0012	1. Francisco López
B0012	2. Javier Alonso
B0012	3. Marta Rebolledo
C0014	1. Francisco López
C0014	2. Javier Alonso
D0120	2. Javier Alonso
D0120	3. Marta Rebolledo

Queremos obter, para cada livro, pares de autores:

1. SELECT A.Codigo, A.Autor, B.Autor FROM Autores A, Autores B
2. WHERE A.Codigo = B.Codigo ;

O resultado é o seguinte:

Código	Autor	Autor
B0012	1. Francisco López	1. Francisco López
B0012	1. Francisco López	2. Javier Alonso
B0012	1. Francisco López	3. Marta Rebolledo
B0012	2. Javier Alonso	2. Javier Alonso
B0012	2. Javier Alonso	1. Francisco López
B0012	2. Javier Alonso	3. Marta Rebolledo
B0012	3. Marta Rebolledo	3. Marta Rebolledo
B0012	3. Marta Rebolledo	2. Javier Alonso
B0012	3. Marta Rebolledo	1. Francisco López
C0014	1. Francisco López	1. Francisco López
C0014	1. Francisco López	2. Javier Alonso
C0014	2. Javier Alonso	2. Javier Alonso
C0014	2. Javier Alonso	1. Francisco López
D0120	2. Javier Alonso	2. Javier Alonso
D0120	2. Javier Alonso	3. Marta Rebolledo
D0120	3. Marta Rebolledo	3. Marta Rebolledo
D0120	3. Marta Rebolledo	2. Javier Alonso

Como podemos observar, os pares de autores se repetem em cada um dos livros. Podemos omitir essas repetições da seguinte forma:

1. SELECT A.Codigo, A.Autor, B.Autor FROM Autores A, Autores B
2. WHERE A.Codigo = B.Codigo AND A.Autor < B.Autor ;

O resultado agora é o seguinte:

Código	Autor	Autor
B0012	1. Francisco López	2. Javier Alonso
B0012	1. Francisco López	3. Marta Rebolledo
C0014	1. Francisco López	2. Javier Alonso
D0120	2. Javier Alonso	3. Marta Rebolledo

Referências

BEIGHLEY, Lynn. *Use a cabeça SQL*. Rio de Janeiro: Altabooks.

FANDERUFF, Damaris. *Dominando o Oracle 9i: modelagem e desenvolvimento*, São Paulo: Makron.

GRAVES, Mark. *Projeto de banco de dados com XML*. 1. ed. São Paulo: Pearson Education do Brasil, 2003.

MORELLI, Eduardo Terra. *Oracle 9i fundamental: SQL, PL/SQL e administração*. São Paulo: Érica.

PRICE, Jason. *Oracle database 11g SQL*. Tradução: João Eduardo Nóbrega Tortello. Porto Alegre: Bookman, 2009.

SILVA, Robson S. *Oracle database 10g express edition*. 1. ed. São Paulo: Érica, 2007.



Avalie este tópico



ANTERIOR

←

Criação de relatórios utilizando mais de uma tabela

Índice

≡

Biblioteca

(<https://www.uninove.br/conhec-a-uninove/biblioteca/sobre-a-biblioteca/apresentacao/>)

Portal Uninove

(<http://www.uninove.br>)

Mapa do Site

Ajuda?

PRÓXIMO

(<https://ava.uninove.br/operacoes-com-conjuntos>)

Operações com conjuntos

→

© Todos os direitos reservados