

1º Exercício

Aluno: João Pedro Valladão Pinheiro

Email: jpinheiro@puc-rio.br

Descrição:

Síntese sobre limites das linguagens relacionais tradicionais, como Álgebra e SQL. Exemplos ilustrativos, lista de referências bibliográficas. Motivação para BD dedutivos (e datalog) mas também outros limites (e.g. BDs OO) que geraram BDs relacionais-objeto (ou objeto-relacionais). Entre 2 e 4 páginas no máximo, mais uma de referências bibliográficas. Colocar também as soluções em SQL das consultas dos (A) 3 maiores salários e (B) pessoas que gostam apenas das cervejas vendidas no Depósito.

Banco de Dados Dedutivos:

Em um sistema de banco de dados dedutivo, o modelo utilizado está relacionado ao modelo de dados relacional e ao campo da programação em lógica, através de uma variação da linguagem Prolog, chamada Datalog. A mesma é utilizada para definição de regras em forma de declaração, junto com um conjunto de relações existentes. Esses são seus dois principais tipos de especificação: fatos e regras.

O conjunto de fatos são os dados presentes no banco de dados, representados pelas tuplas. As tuplas, em suas relações, descrevem fatos do mundo real. Vale ressaltar que esses fatos são considerados universo - baseado na hipótese do "mundo fechado". As regras especificam relações virtuais que não estão armazenadas, mas podem ser formadas com base nos fatos, ao aplicar mecanismos de inferência baseados nas especificações da regra. Um ponto interessante com relação as regras é a possibilidade de recursão.

Com isso, podemos estabelecer como definição que banco de dados dedutivos são aqueles em que existe um mecanismo de inferência, incluindo recursão.

Banco de Dados Orientado a Objetos:

O termo orientação a objetos vêm das linguagens de programação com essa característica (LPOO). Um objeto possui dois componentes: estado (valor) e comportamento (operações). LPOO possui objetos transientes, que existem apenas durante a execução do programa. Já um banco de dados orientado a objetos (BDOO) podem adicionar a capacidade de persistir os objetos.

Outros conceitos importantes em LPOO são: herança e hierarquias entre dados e operações. Essas duas características, em especial, permitem e promovem o reuso das definições de tipos existentes em novos

tipos de objetos. Ainda há outras questões, tais como: identificação única dos objetos, construtores de tipo de dados, encapsulamento de operações, polimorfismo e sobrecarga de operador.

Para cobrir tais exigências demandadas pela OO, houve necessidade da extensão da linguagem SQL, conhecida como SQL/Object. Após, foram incorporadas na sintaxe mais recente da linguagem SQL:2008. Além da adequação em nível de linguagem, foi proposto o mapeamento de esquemas entidade relacionamento para esquemas banco de dados de objetos.

Referências:

- Elmasri, Navathe
Sistemas de banco de dados / Ramez Elmasri e Shamkant B. Navathe;
tradução Daniel Vieira; revisão técnica Enzo Scraphim e Thatyana de Faria Piola Scraphim.
6a ed. - São Paulo : Pearson Addison Wesley, 2011.

Apêndice (Consultas):

(A) Quais os nomes dos empregados que ganham os 3 maiores salários da empresa?

- Esquema: Empregados
 - emp (Id, Nome, Salario)

```
-- SQL Estendido

select id, nome from emp where salario in (
    select distinct salario from emp order by salario desc limit 3
);

-- SQL ANSI

select emp3.id, emp3.nome
from (
    select distinct
        emp1.salario, (
            select count(distinct emp2.salario)
            from emp emp2
            where emp1.salario < emp2.salario
        ) qtd
    from emp emp1
) tmp, emp emp3
where emp3.salario = tmp.salario
    and tmp.qtd < 3
;
```

(B)

- Esquema: Turma do Funil
 - Venda (Bar, Cerveja)
 - Gostam (Pessoa, Cerveja)

Listar todas as pessoas que gostam das cervejas vendidas no Pires

```
select distinct pessoa
from gostam where cerveja in (
    select cerveja from venda where bar = 'Pires'
);
```

Listar todas as pessoas que gostam apenas das cervejas vendidas no Pires

```
select distinct pessoa
from gostam
where pessoa not in (
    select pessoa
    from gostam
    where cerveja not in (
        select cerveja
        from venda
        where bar = 'Pires'
    )
);
```

Listar todas as pessoas que gostam das cervejas vendidas apenas no Pires

```
select distinct pessoa
from gostam
where cerveja not in (
    select cerveja from venda where bar <> 'Pires'
)
;
```