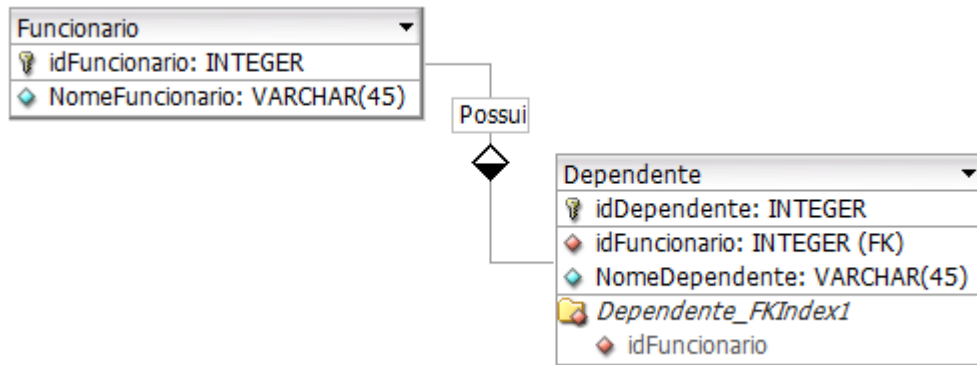


Exercícios - Trigger

Dado o seguinte modelo...



... criar as tabelas (em uma database qualquer) e inserir alguns registros nas tabelas FUNCIONARIO e DEPENDENTE de maneira a relacioná-los utilizando, por exemplo, o script abaixo:

```
CREATE TABLE Funcionario (  
    idFuncionario INTEGER NOT NULL,  
    NomeFuncionario VARCHAR(45) NULL,  
    PRIMARY KEY(idFuncionario)  
);  
  
CREATE TABLE Dependente (  
    idDependente INTEGER NOT NULL,  
    idFuncionario INTEGER NOT NULL,  
    NomeDependente VARCHAR(45) NULL,  
    PRIMARY KEY(idDependente),  
    INDEX Dependente_FKIndex1(idFuncionario),  
    FOREIGN KEY(idFuncionario)  
        REFERENCES Funcionario(idFuncionario)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION  
);  
  
insert into funcionario values (1,'José Antonio da Silva');  
insert into funcionario values (2,'Luis Augusto dos Anjos');  
insert into funcionario values (3,'Pedro Caetano de Souza');  
insert into funcionario values (4,'Ana Maria Ribeiro');  
insert into funcionario values (5,'Lucia da Costa');  
  
insert into dependente values (1,1,'Jose Antonio da Silva Filho');  
insert into dependente values (2,1,'Jose Luis da Silva Jr');  
insert into dependente values (3,1,'Josilda da Silva');  
insert into dependente values (4,2,'Luisa Augusta dos Anjos');
```

```
insert into dependente values (5,2,'Luis Augusto dos Anjos Jr');
insert into dependente values (6,3,'Pedro Caetano de Souza Filho');
insert into dependente values (7,5,'Lucio da Costa');
insert into dependente values (8,5,'Lucia Maria da Costa');
insert into dependente values (9,5,'Luciano da Costa');
insert into dependente values (10,5,'Luciana da Costa');
```

Depois, criar uma TRIGGER que faça a seguinte função: No instante que for excluído um registro da Tabela FUNCIONÁRIO, também seja(m) excluído(s) o(s) registro(s) da Tabela DEPENDENTE. Ou seja, quando ocorrer uma demissão na empresa e os dados do funcionário forem excluídos da base de dados da empresa (Tabela FUNCIONÁRIO), também seja(m) excluído(s) todo(s) o(s) seu(s) dependente(s) (Tabela DEPENDENTE).

Resolução:

Depois de criar as tabelas do modelo e inserir os dados (registros) nessas tabelas, criar uma trigger (usando o HeidiSQL, por exemplo) dentro desta database utilizando o seguinte comando:

```
delimiter |
create trigger exclui_dep
before delete on funcionario
for each ROW
begin
delete from dependente where idFuncionario = old.idFuncionario;
end
|
delimiter ;
```

Depois de criada a trigger, execute o seguinte comando:

```
delete from funcionario where idfuncionario = 2;
```

Será excluído o funcionário “Luis Augusto dos Anjos” da tabela funcionário.

Depois, verifique os dados dos dependentes utilizando o seguinte comando:

```
Select * from dependente;
```

Foram excluídos também os registros relacionados ao registro deste funcionário. Ou seja, os dependentes “Luisa Augusta dos Anjos” e “Luis Augusto dos Anjos Jr”, da tabela dependentes, foram excluídos com o recurso da trigger.

Como funciona:

A trigger EXCLUI_DEP...

```
delimiter |
create trigger exclui_dep
before delete on funcionario
for each ROW
begin
delete from dependente where idFuncionario = old.idFuncionario;
end
|
delimiter ;
```

...foi criada com a finalidade de **excluir** um registro da tabela **DEPENDENTE ANTES** (BEFORE) que seja **excluído** um registro da tabela **FUNCIONARIO**. Ou seja, quando excluirmos um registro da tabela FUNCIONARIO também será(serão) deletado(s) o(s) registro(s) da tabela DEPENDENTE cuja Chave Primária da tabela FUNCIONARIO for igual ao atributo idFuncionario da tabela DEPENDENTE (Chave Estrangeira).

Mas como a trigger vai excluir um registro (da tabela dependente) **ANTES** de ter **excluído** um registro (da tabela funcionário)? Para isso usamos o argumento OLD que difere da opção NEW justamente por causa do registro já estar sendo considerado “inexistente” (“old.idfuncionario” refere-se ao campo “idfuncionario” – que será excluído – da tabela FUNCIONARIO).

Ou seja, se eu fosse realizar uma ação DEPOIS de um registro ter sido criado, pode-se considerar que este registro criado “passou a existir” sendo considerado um NOVO registro (nesse caso, seria mencionado “new.idfuncionario”).

E, ainda relativo a este caso, não podemos utilizar a opção AFTER na trigger por causa das CONSTRAINTS (Integridade Referencial). Ou seja, **como a trigger vai excluir um registro (da tabela dependente) DEPOIS de já ter excluído um registro da tabela funcionário?** Como a integridade referencial não permite isso devemos primeiro excluir os registros da tabela DEPENDENTE para depois excluir os registros da tabela FUNCIONARIO.

Pergunta:

Se eu especificasse a trigger com a opção “**AFTER delete on funcionário**” (ao invés de “**BEFORE delete on funcionário**”) na minha trigger e também descrevesse “**idfuncionario=idfuncionario**” (ao invés de “**idFuncionario = old.idFuncionario**”) ela funcionaria?

Resposta:

Não

Você saberia responder porque?