

1) A base de dados de uma dada empresa contém as tabelas correspondentes às seguintes definições:

```
create table Produto (CodProd numeric(6) primary key,  
                     NomeProd char(50), QtStock numeric(8) not null  
)  
  
create table Fatura(IdFat numeric(6), Ano numeric(4),  
                   NomeCliente char(50) not null,  
                   NIFCli numeric(9) not null,  
                   primary key(IdFat,Ano))  
  
create table Item(IdFat numeric(6), AnoFat numeric(4), NumItem numeric(4),  
                 CodProd numeric(6) not null references Produto,  
                 Quantidade numeric(8) not null,  
                 primary key(IdFat,AnoFat,NumItem),  
                 foreign key(IdFat,AnoFat) references Fatura  
)
```

Tendo-se concluído que na maior parte das vezes o acesso às faturas envolve também o acesso aos respetivos itens, foi decidido criar a seguinte vista:

```
create view FatItem as  
select f.*, i.NumItem, i.CodProd, i.Quantidade from Fatura as f  
inner join Item as i on f.IdFat = i.IdFat and f.Ano = i.AnoFat
```

Porém, concluiu-se que o acesso à vista demora demasiado tempo quando se fazem acessos a gamas de faturas extensas. Alguém sugeriu que, dado que todas as faturas têm de ter pelo menos um item, se poderia substituir a vista FatItem pela seguinte tabela:

```
create table FatItemTab(IdFat numeric(6), AnoFat numeric(4),  
                       NomeCliente char(50) not null, NIFCli numeric(9) not null,  
                       NumItem numeric(4),  
                       CodProd numeric(6) not null references Produto,  
                       Quantidade numeric(8) not null,  
                       primary key(Idfat,Anofat,NumItem),  
                       foreign key(Idfat,AnoFat) references Fatura,  
                       foreign key(IdFat,AnoFat,NumItem) references Item  
)
```

Mas é necessário garantir que esta tabela está sempre sincronizada com as tabelas Fatura e Item.

- a) Discuta uma solução que o permita conseguir.
- b) Implemente essa solução contemplando apenas as inserções e remoções sobre a tabela Item.

2) O que entende por “*predicate locking*”? Como é que o Sql Server lida com este conceito?

3) Considere o seguinte código TSQL:

```
create table t(i int primary key, j int)
insert into t values (1,1)
insert into t values (2,2)
insert into t values (3,3)

set transaction isolation level read committed
begin tran
select * from t where i=2
--ponto 1
update t set j = j*10 where i = 2
--ponto 2
commit
--ponto 3
```

Indique, justificando, quais os *locks* colocados por esta transação sobre cada um dos registos, bem como o tipo de *lock* em cada um dos pontos assinalados. Identifique os registos pela respetivo valor da chave.

4) Considere o seguinte código TSQL:

```
create table t(i int primary key, j int)
insert into t values (1,1)
insert into t values (2,2)
insert into t values (3,3)
```

E o seguinte conjunto de instruções:

```
I1: select * from t where i = 3
I2: select * from t where i < 3
I3: update t set j = j*10 where i = 1
I4: update t set j = j*100 where i > 2
I5: update t set j = j*1000
```

Considere também duas transações com os seguintes esqueletos:

T1	T2
set transaction isolation level ?1	set transaction isolation level ?3
begin transaction	begin transaction
?2	?4
commit	commit

Substitua os ?N nas duas transações de forma a que, usando apenas subconjuntos das instruções indicadas, se produzam os seguintes efeitos potenciais quando T1 e T2 se executam concorrentemente (indique também os escalonamentos que a isso conduziriam):

- a) *Dirty reads*, sem *non repeatable reads* nem *phantoms*.
- b) *Non repeatable reads* sem *dirty reads* nem *phantoms*.
- c) *Phantoms* sem *dirty reads* nem *non repeatable reads*.
- d) *Deadlock*.

Nota: Caso não seja possível verificarem-se os efeitos indicados, justifique-o.

5) Compare os objetos **DataReader** do **ADO.Net** com os cursores SQL.

6) Um exame é identificado pela época em que ocorre (normal, recurso, especial), pelo código da unidade curricular e pela data em que ocorre. Cada exame possui um júri composto por 3 docentes e um conjunto de zero ou mais alunos. Um docente é caracterizado pelo número mecanográfico, pelo nome e categoria (assistente, prof. adjunto, prof. coordenador). No contexto de um exame, cada aluno é caracterizado pelo seu número, nome e pela nota que obtém nesse exame.

a) Apresente um documento xml que represente o exame de época especial da UC de código SI2, realizado no dia 15/9/1017, cujo júri é composto pelo professor coordenador Walter Vieira (668), pelo professor adjunto Nuno Datia (1474) e pelo assistente Afonso Remédios (1601). O exame foi realizado pelos seguintes alunos:

Número	Nome	Nota
1111	Ana Rita	14
2222	Pedro Silva	8
3333	Rui Pinto	14

b) Apresente um *xml schema* que permita validar documentos do tipo da alínea anterior e diga quais as alterações que deveria realizar ao documento que desenvolveu na alínea anterior para o associar a este *xml schema*.

Cotação:

alínea	1.a	1.b	2	3	4.a	4.b	4.c	4.d	5	6.a	6.b	Total
cotação	2	2	1,5	2	2	2	2	2	1,5	1	2	20