

DI-FCT-NOVA

8 de junho de 2022

Bases de Dados

2º teste, 2021/22 – Versão A

Duração: 1,5 horas/ 1 hora (UC 7747) (sem consulta)

- Os alunos da **UC 7747** (com 6 ECTS) só devem responder às **Questões 1-3**.
- Deve colocar nome e número em cada página das folhas de resposta pintando também os círculos correspondentes aos dígitos do número de aluno.
- Abaixo do nome deve também pintar os círculos correspondentes ao número de filas à sua frente e ao número de colunas (de estudantes) à direita (exemplo: se estiver sentado na segunda fila o mais à direita possível, deve indicar uma fila à frente e zero colunas à direita).
- Cada pergunta tem um identificador único - deve **responder a cada pergunta só dentro do espaço previsto para a mesma** – o que não estiver dentro do mesmo não será avaliado.
- Pode responder a lápis (desde que seja legível), e pode entregar após 45 minutos.
- A apresentação, simplicidade e clareza das respostas contarão na avaliação.

Questão 1 [7 valores]

Considere o seguinte esquema de base de dados, onde os atributos chave de cada relação se encontram sublinhados:

Cervejas(nome, fabricante)
Bares(nome, morada, licença)
Pessoas(nome, morada, telefone)

Gosta(nomeP, nomeC)
Vende(nomeB, nomeC, preço)
Frequenta(nomeP, nomeB)

Este esquema representa informação sobre *bares* (caracterizados por um nome, morada e licença), *cervejas* (caracterizadas por um nome e um fabricante) e *pessoas* (caracterizadas por um nome, morada e telefone), bem como informação sobre que bares (nomeB) vendem que cervejas (nomeC) a que preço, que pessoas (nomeP) gostam de que cervejas (nomeC), e quais os bares (nomeB) frequentados por que pessoas (nomeP).

Nenhum atributo pode conter nulos.

- [1]** Apresente uma consulta em datalog para encontrar as pessoas que frequentam bares que não vendem nenhuma cerveja que eles gostam.
- [3]** Apresente uma consulta em SQL para criar uma vista que mostra todos os fabricantes de cervejas junto com o número de cervejas produzido por eles que são as preferidas num dos bares, ou seja, as cervejas que entre as vendidas em cada bar tem o maior número de pessoas a frequentar o mesmo que gostam da cerveja (poderá criar vistas auxiliares).
- [1,5]** Apresente código SQL para assegurar que só estão à venda cervejas por um preço de pelo menos um euro e das quais se consegue saber o seu fabricante.
- [1,5]** Apresente código SQL para evitar que se insira na base de dados pessoas a frequentar bares onde não se vende nenhuma cerveja de que eles gostam.

Questão 2 [1 valor]

Assuma que o utilizador A é o dono da relação à qual o privilégio P se refere. A seguir à execução da seguinte sequência de comandos, pelos utilizadores indicados, quais os utilizadores que, para além de A, possuem o privilégio P? Justifique!

Passo	Utilizador	Comando
1	A	GRANT P TO B WITH GRANT OPTION
2	A	GRANT P TO C WITH GRANT OPTION
3	B	GRANT P TO D WITH GRANT OPTION
4	D	GRANT P TO B WITH GRANT OPTION
5	C	GRANT P TO D
6	A	REVOKE P FROM B CASCADE
7	A	REVOKE P FROM C RESTRICT

Questão 3 [Cotação: por cada alínea, 2 valores]

Considere o seguinte esquema de relação $R = (A, B, C, D, E, F)$ sobre o qual se deseja impor as seguintes dependências funcionais:

$$A \rightarrow C \ E \quad B \ F \rightarrow A \ D \quad C \rightarrow E \quad C \ E \rightarrow F$$

- Apresente as chaves candidatas de R , justificando adequadamente.
- Decomponha o esquema de R para uma forma normal Boyce-Codd, justificando cada passo.
- Decomponha o esquema de R para uma terceira forma normal, justificando cada passo.

Questão 4 [1 valor]

Considere a tabela t com schema $T = (a, b)$ com dois tuplos - (1,3) e (2,2) - e o seguinte escalonamento (em modo READ UNCOMMITTED - pode assumir que os resultados dos SELECT são visualizados no ecrã).

Transação 1	Transação 2	Transação 3
	INSERT INTO t VALUES (3,1);	
SELECT SUM(b) FROM t ;		
		UPDATE t SET $b=4$ WHERE $a=1$;
	commit;	
UPDATE t SET $b=1$ WHERE $a=2$;		
commit;		
		SELECT SUM(b) FROM t ;
		commit;

- Justifique apropriadamente se o escalonamento é serializável.

Questão 5 [5 valores]

Considere o seguinte XML DTD:

```
<!DOCTYPE bares [
  <!ELEMENT bares (bar*,cerveja*,cliente*)>
  <!ELEMENT bar (venda+)>
  <!ATTLIST bar nome ID #REQUIRED
              clientes IDREFS #REQUIRED>
  <!ELEMENT cerveja (fabricante)>
  <!ATTLIST cerveja nome ID #REQUIRED>
  <!ELEMENT cliente EMPTY>
  <!ATTLIST cliente nome ID #REQUIRED
                  gosta IDREFS #IMPLIED>
  <!ELEMENT venda (preco?)>
  <!ATTLIST venda cerveja IDREF #REQUIRED>
  <!ELEMENT fabricante (#PCDATA)>
  <!ELEMENT preco (#PCDATA)>]>
```

- [2] Escreva um documento XML de acordo com o DTD acima que contenha a seguinte informação:
 - A Luísa frequenta o bar Estrela onde gosta tanto da Bohemia como da Radler (ambas fabricadas pela SCC).
 - O Miguel para além do Estrela frequenta o bar Lua onde se vende a Stout por 2 euros (fabricada pela Super Bock).
- Apresente expressões XPATH sobre um ficheiro de acordo com a DTD acima para obter:
 - [1] Os nomes dos bares que vendem mais do que três cervejas por menos do que 2 euros.
 - [1] Os fabricantes das cervejas que os clientes do bar Lua gostam.
- [1] Apresente uma expressão XQuery que mostra os elementos cliente do DTD acima, colocando o nome do cliente e os nomes das cervejas que gosta como subelementos (em vez de atributos).

FIM