E-fólio Global | Instruções para a realização do E-fólio

Critérios de Correção

SISTEMAS DE GESTÃO DE BASES DE DADOS | 21103 | ÉPOCA NORMAL

Período de Realização: decorre 09-02-2022 deste 15:00 com 2:30 horas de duração

Data de Limite de Entrega: até 17:30 de Portugal Continental

Temática / Tema / Conteúdos: Sistemas de gestão de bases de dados

Objetivos: Reconhecer formas de armazenamento de dados e formas de otimização de consultas; reconhecer o sistema transacional e formas de recuperação de dados; reconhecer ambientes de Data Warehouse, Data Mining e Information Retrieval.

Trabalho a desenvolver: Resolução de um conjunto de exercícios.

Critérios de avaliação e cotação: A cotação deste e-fólio é de 120 pontos = 12 valores, pode encontrar as cotações parciais junto de cada pergunta. A interpretação das perguntas também faz parte da sua resolução, se encontrar alguma ambiguidade deve indicar claramente como foi resolvida. Critérios de avaliação gerais: (i) para a dificuldade de leitura (linhas cruzadas, letras com fontes desadequadas) a penalização é de 20% a 100%; (ii) para erros e omissões a penalização é de 20% a 100%.

Normas a respeitar: Deve redigir o seu E-fólio na Folha de Resolução disponibilizada na turma e preencher todos os dados do cabeçalho. Podem ser incluídas imagens e digitalizações de conteúdos produzido manualmente pelo estudante. Todas as páginas do documento devem ser numeradas. O documento A4 deve ser redigido em Times New Roman, tamanho de letra 12. O espaçamento entre linhas deve corresponder a 1,0 ou 1,5 linhas. Nomeie o ficheiro com o seu número de estudante, seguido da identificação do E-fólio, segundo o exemplo apresentado: 000000efolioGlobal. Finalmente deve gerar um PDF do documento. Deve carregar o referido ficheiro para a plataforma no dispositivo E-fólio Global até à data e hora limite de entrega. Evite a entrega próximo da hora limite para se precaver contra eventuais problemas. O ficheiro a enviar não deve exceder 8 MB. Votos de bom trabalho!

A informação da avaliação do estudante está contida no vetor das cotações:

Questão:	1	2	3	4	5
Cotação:	2	2	2	4	2

Tópico 1 – Transações e Consultas

1. (2 valores) Relativo ao tema das Transações, considere o seguinte sequenciamento r1(x), r2(x), w2(x), r3(y), w4(x), r5(x), w5(z), w1(z).
a) O que entende por sequenciamento serializável a conflitos? O que entende por sequenciamento serializável a vistas? Justifique e exemplifique a resposta.
b) O referido sequenciamento é serializável a conflitos? O referido sequenciamento é serializável a vistas? Justifique e exemplifique a resposta.

Resposta:

1.a) Definições:

i) Definição escalonamento seriável a conflitos
Serialização a conflitos: um escalonamento é seriável por conflitos se o seu grafo de precedência não contiver ciclos.
Na construção do grafo de precedências as arestas são montadas a partir das observações das transações que participa, da escala sendo duas transações T_i e T_j haverá uma aresta: $T_i \rightarrow T_j$ se forem observadas as seguintes condições:

- T_i executa *write*(q) antes de T_j executar *read*(q)
- T_i executa *read*(q) antes de T_j executar *write*(q)
- T_i executa *write*(q) antes de T_j executar *write*(q)

ii) Definição de escalonamento seriável a vistas.
Serialização por vistas: Diz-se que "Ti lê x de Tj", se Wj(x) for a última operação de escrita antes de Ri(x). Dois escalamentos são equivalentes a vistas, se a relação "lê de" em ambas for a mesma.
Dois escalonamentos S1 e S2 são equivalentes se:

- Todos os elementos A têm o mesmo valor inicial
  Se S1: $r_i(A)$ lê o valor inicial, então S2: $r_i(A)$ também lê o mesmo valor inicial
- Um elemento A tem a mesma vista nos dois escalonamentos
  Se S1: $w_j(A) \Rightarrow r_i(A)$ então em S2: $w_j(A) \Rightarrow r_i(A)$; têm a mesma vista
- Todos os elementos A têm a mesma vista final
  Se S1: T_i termina $w_i(A)$, então S2: T_i também termina com $w_i(A)$; os valores finais são os mesmos.

1.b) Considerando o seguinte sequenciamento r1(x), r2(x), w2(x), r3(y), w4(x), r5(x), w5(z), w1(z)

i) Exemplo escalonamento seriável a conflitos:

Divisão por recurso/item:	As precedências são as seguintes:
Item x: R1, R2, W2, W4, R5	R1->W2, R1->W4, R2->W4, W2->W4, W2->W4, W2->R5, W4->R5
Item y: R3	N.A.
Item z: W5, W1	W5 -> W1

Existem três ciclos, 1->2->5->1, 1->4->5-> 1 e 1->2->4->5->1, pelo que o escalonamento não é seriável a conflitos.

ii) Consideremos o escalonamento e a tabelas de valores iniciais, update e finais:

T1	**T2**	T3	T4	T5
R(x)				
	R(x)			
	W(x)			
		R(y)		
			W(x)	
				R(x)
				W(z)
W(z)				

	x	y	z
inicial R	T1, T2, T5	T3	
update RW			
final W	T2, T4		T5, T1

Não existem sequências cíclicas. O escalonamento é seriável a vistas.

Critério de correção:
- 0,5 valores, definições (a)
- 0,5 valores, seriável a conflitos (b)
- 1,0 valor, seriável a vistas (b)
- erros, omissões, redundâncias: -20% a -100%

2. (2 valores) Relativamente à concorrência em bases de dados:
a) Defina o protocolo 2-PL relativamente à seriabilidade e à recuperação e ao 'deadlock'. Justifique a resposta.
b) Considere o protocolo 2-PL e acrescente os operadores X-lock(_), S-lock(_) e Unlock(_). Existe a possibilidade de 'deadlock'? Complete a tabela e justifique a resposta.

T1	T2	Justificação
Read (A) Read (B) . . Write (B) Write (A) . .	. . Read (B) Read (A) . . Write (A) Write (B)	

Resposta:

2.a) O protocolo 2-PL (*Two-Phase Locking Protocol*), requer que cada transação recorra a bloqueios, garantindo assim a seriabilidade, para aceder aos recursos em duas fases:

1- Fase de Crescimento: Uma transação pode obter bloqueios, mas não pode libertar nenhum;

2- Fase de Encolhimento: Uma transação pode libertar bloqueios, mas não consegue obter nenhum bloqueio novo

Em 2-PL usamos a seguinte matriz de compatibilidade de 'locks':

State of the lock \ *Lock request type* →	*Shared*	*Exclusive*
Shared	Yes	No
Exclusive	No	No

Este protocolo, apesar de garantir a serialização não garante a recuperação a falhas nem ao 'deadlocks'.

2.b) Tabelas com locks

T1 com locks/unlocks	T2 com locks/unlocks	Justificação
X-lock(A), Read(A)		lock A
X-lock(B), Read (B)		lock B
.	Read(B)	X-lock(B) não é possível, waiting B
.	Read (A)	X-lock(A) não é possível, waiting A
Write (B), Unlock(B)	.	liberta B
Write (A), Unlock(A)	.	liberta A
	X-lock(B), Read (B)	lock B
	X-lock(A), Read(A)	lock A
	Write (A), Unlock(A)	liberta A
	Write (B), Unlock(B)	liberta B
		termina com sucesso

As transações terminam com sucesso.

Critério de correção:
- 0,5 valores (2.a)
- 1,0 valor, tabelas com transações e justificações (2.b)
- 0,5 valores, resposta final (2.b)
- erros, omissões, redundâncias: -20% a -100%

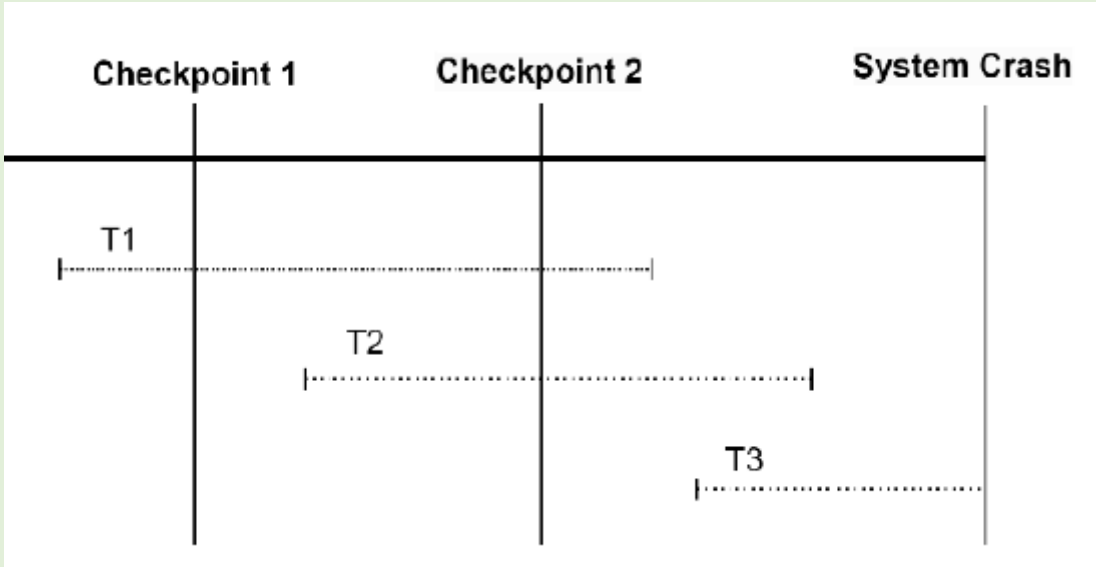
3. (2 valores) Considere a seguinte sequência de 'logs' das transações T1, T2 e T3.

Log	Redo Phase	Undo Phase
Start T1 Read T1, D Checkpoint Write T1, D, 20, 25 Start T2 Read T2, B Checkpoint Commit T1 Start, T3 Write T2, B, 12, 18 Commit T2 Read T3, D Write T3, D, 25, 15 System crash ←-----		

a) Represente as transações e os checkpoints na linha do tempo (diagrama temporal).
b) Acrescente os registos na recuperação. Complete a tabela e justifique a resposta.

Resposta:

3.a) Diagrama temporal



3.b) Tabela com fases e registos de recuperação

Log	Redo-Phase	Undo-Phase
	undo-list=[]	Fim Undo-Phase
Start T1	undo-list=[T1]	
Read T1, D		
Checkpoint 1	Checkpoint 1	
Write T1, D, 20, 25	Redo T1, D, 20, 25	
Start T2	undo-list=[T1, T2]	
Read T2, B		
Checkpoint 2	Checkpoint 2	
Commit T1	undo-list=[T2]	undo-list=[]
Start, T3	undo-list=[T2, T3]	undo T3
Write T2, B, 12, 18	Redo T2, B, 12, 18	
Commit T2	undo-list=[T3]	
Read T3, D		
Write T3, D, 25, 15	Redo T3, D, 25, 15	
System crash <-----	----------	undo-list=[T3]
		Início Undo-Phase
Redo T1, D, 20, 25		Registos acrescentados
Redo T2, B, 12, 18		na Redo-Phase
Redo T3, D, 25, 15		e Undo-Phase
Undo T3		
Abort T3		

Critério de correção:
- 0,5 valores, diagrama temporal
- 1,5 valores, registos acrescentados, Redo-Phase e Undo-Phase
- erros, omissões, redundâncias: -20% a -100%

Tópico 2 – Data Warehousing, Mining e Information Retrieval

4. (4 valores) Considere a seguinte base de dados que vai servir de fonte de dados a um "Data Warehouse".

a) Encontre a 1FD (1ª forma desnormalizada) e a 2FD (2ª forma desnormalizada).
1FD – obter uma poli-árvore, replicando as tabelas que forem necessárias;
2FD – obter árvores separadas, replicando as tabelas que forem necessárias.
Justifique a resposta.

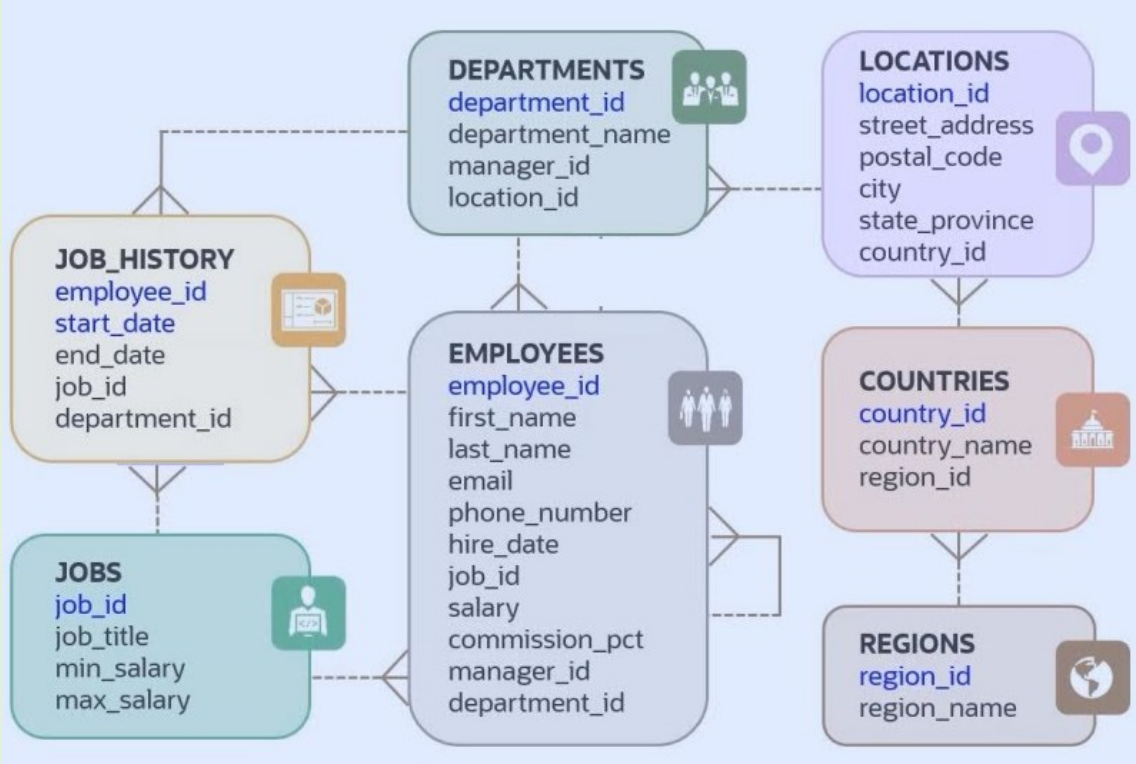
b) Verifique se a tríade (Departements, Employees, Job_History) é ou não redundante. Justifique a resposta.

c) Considerando que:

- Aditivos: são atributos que podem ser agregados (somados) por todas as dimensões, ex: valor da venda (usar Sum() sempre)
- Semi-aditivos: são atributos que podem ser agregados (somados) por algumas as dimensões, ex: quantidade (usar Sum() em condições particulares)
- Não-aditivos: são atributos que não podem ser agregados (somados), ex: preço unitário (usar Average() por exemplo)
- Sem factos: só existem identificadores (usar a função Count() dos identificadores).

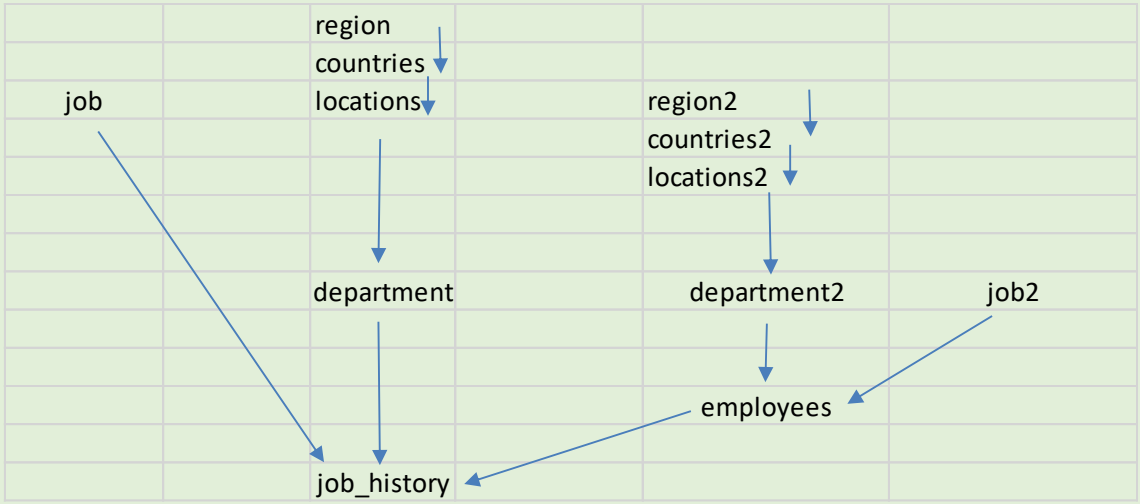
Para as tabelas de factos encontradas defina os atributos aditivos, semi-aditivos, não-aditivos e sem factos. Justifique a resposta.

d) De seguida, defina um DW, em estrela ou constelação, com pelo menos três dimensões para cada tabela factos.



Resposta:

4.a) 1FD (1ª forma desnormalizada) apresenta-se na figura. A 2FD é igual à 1FD porque só existe uma tabela de factos.



4.b) Verifique se a tríade (Departements, Employees, Job_History) é ou não redundante. Não é redundante porque Employees tem a informação no tempo corrente e o Job_history tem o histórico.

4.c) Tipos de atributos

Look Up	regions	
	jobs	
Intermédias	countries	
	locations	
	departments	
	employees	
Factos	job_history	sem factos

4.d) Informação DW

	employees	departments	jobs	time
Factos	X	X	X	X

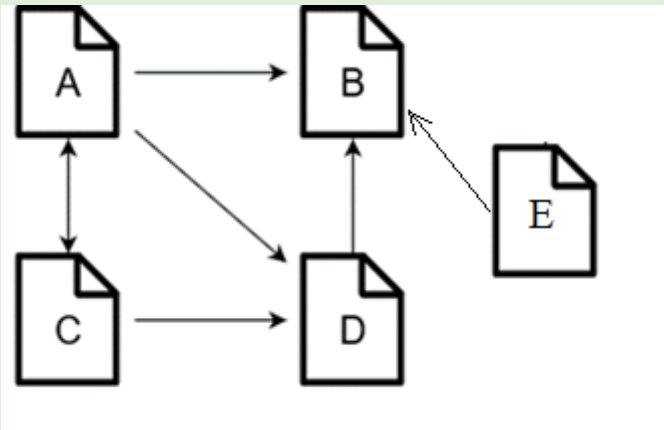
Critério de correção:
- 1,0 valor, 1FD e 2FD (4.a)
- 1,0 valor, redundância tríade (4.b)
- 1,0 valor, tipos atributos (4.c)
- 1,0 valor, informação DW (4.d)
- erros, omissões, redundâncias: -20% a -100%

5. (2 valor) Information Retrieval

Considere o algoritmo original de PageRank descrito por Lawrence Page and Sergey Brin em 1995 é dado por:

$$P[j] = \delta + (1 - \delta) * \sum_{i=1}^{N} (T[i,j] * P[i])$$

com $\delta = 0{,}5$.
Encontre a ordenação das páginas para o seguinte conjunto:



Resposta:

PR(A) = 0,5 + (1 - 0,5) * (1/2 * C)
PR(B) = 0,5 + (1 - 0,5) * (1/3 * A + 1 * D +1 * E)
PR(C) = 0,5 + (1 - 0,5) * (1/3 * A)
PR(D) = 0,5 + (1 - 0,5) * (1/3 * A + 1/2 * C)
PR(E) = 0,5 + (1 - 0,5) * (0)

T[i,j]						
	A	B	C	D	E	soma
A	0	1/3	1/3	1/3	0	1
B	0	0	0	0	0	0
C	1/2	0	0	1/2	0	1
D	0	1	0	0	0	1
E	0	1	0	0	0	1

iteração	A	B	C	D	E
0	1	1	1	1	1
1	0.75	1.67	0.67	0.92	0.50
2	0.67	1.33	0.63	0.79	0.50
3	0.66	1.26	0.61	0.77	0.50
4	0.65	1.24	0.61	0.76	0.50
5	0.65	1.24	0.61	0.76	0.50

Conclusão: B > D > A > C > E

Critérios de correção:
- i) 1 valor, tabela T(i,j) e fórmulas
- ii) 1 valor, resultado depois de algumas iterações
- erros, omissões, redundâncias ou apresentação desadequada: -20% a -100%

FIM