I. Leia atentamente as questões seguintes e para cada uma selecione a única opção completamente correta:

II.

1. No MS SQL para minimizar a perda de dados devido a corrupções nos ficheiros:

a. Os ficheiros mdf, ndf e ldf devem ser colocados na mesma diretoria (sem RAID)

b. Os ficheiros mdf, ndf e ldf devem ser colocados na mesma diretoria sob RAID0

c. Os ficheiros mdf, ndf e ldf devem ser colocados em discos diferentes

d. Respostas b e c

Complemento de Resposta:

Para existir uma minimização de perda de dados devido a corrupções nos ficheiros, devemos sempre colocar os ficheiros primários (mdf), secundários (ndf) e de logs (ldf) em discos diferentes. Os backups destes ficheiros corrompidos também fundamenta a resposta a esta pergunta. Pois todos os backups deveram estar em discos diferentes.

2. Considerando realizado um backup integral, uma operação de backup diferencial, pode ser realizada após:

a. Um outro backup diferencial

b. Um backup de transacções

c. Qualquer das duas operações anteriores

d. Nenhuma das duas operações

Complemento de resposta:

Um backup diferencial retrata um backup das últimas alterações efetuadas desde o último backup integral, logo, se tiver entre backup integrais, não importa se foi feito um outro backup diferencial ou backup de transações antes.

3. Numa consulta com condição de seleção (idade BETWEEN 18 AND 25), o campo idade, seria melhor candidato a uma indexação tipo:

a. Hash

b. B+ Tree

c. Sequencial

d. É indiferente

Complemento de resposta:

Como esta consulta com condição de idades pode ser retratada como uma consulta em sequência, tanto B+Tree como sequencial seriam resposta corretas. Mas considera-se B+tree como uma alternativa a índices sequenciais, sendo mais comumente utilizado, devido às suas vantagens.

4. O nível de isolamento snapshot isolation, evita as seguintes ocorrências:

a. Somente Non-repeatable read e phantom read

b. Non-repeatable read, phantom read e Dirty read

c. Somente Phantom read

d. Nenhuma das anteriores

Complemento de resposta:

Tanto o nível de isolamento **snapshot e o serializable** evitam todas as ocorrências existentes (Non-Repeatable read, phantom read e dirty read).

Efeitos secundários admitidos por nível de isolamento:

	Dirty read	***Non-Repeatble read***	***Phantom Read***
Read Uncommited	Sim	Sim	Sim
Read Commited	Não	Sim	Sim
Repeatable Read	Não	Não	Sim
Serializable	Não	Não	Não
Snapshot	Não	Não	Não

5. Sejam os seguintes acontecimentos: backup integral_1 + backup Tlogs + backup diferencial + backup integral_2 + crash da BD. A recuperação da BD, com vista a minimizar a perda de dados, deverá ter a seguinte sequência:

a. Repor Backup Integral_1 + repor Backup TLogs + repor Backup Diferencial + Repor Backup Integral_2

b. Repor Backup TLogs + repor Backup Diferencial + repor Backup Integral_2

c. Backup Log + repor Backup Integral_2 + repor Backup Log

d. Backup Log + repor Backup Diferencial + repor Backup Integral_2 + repor Backup Logs

Complemento de Resposta:

Os passos definidos para um restore bem sucedido são:

I. Fazer o backup do tail do transaction log;
II. Recuperar a partir do backup completo;
III. Recuperar os logs das transacções;
IV. Recuperar o tail dos logs das transações

6. No contexto de transações concorrentes, qual o nível de isolamento mínimo que evita a ocorrência de "dirty read"

a. Read commited

b. Read uncommited

c. Repetable read

d. Serializable

Complemento de Resposta:

Read Commited:

- Os operadores de leitura e de escrita são diferentes;
- Não existe bloqueio sobre updates;
- **Evita problemas de Dirty Read.**

7. Relativamente ao modo de recuperação de uma BD

a. Operando em bulk logged tem ficheiros de registos de logs maiores

b. Operando em full logged tem ficheiros de registos de logs maiores

c. Operando em bulk logged tem menor potencial de perda de dados em caso de corrupções no ficheiro de dados

d. Nenhuma das anteriores

Complemento de resposta:

Proteção mais elevada contra a perda de informação **todas as alterações são escritas no ficheiro de logs de transações**. **É o mais dispendioso em espaço necessário** para o ficheiro de logs, e desempenho pois **cada transação é guardada no ficheiro de log**.

8. No âmbito da encriptação de dados na BD

a. A utilização de chaves assimétricas potencia alguma degradação da performance face à utilização de chaves simétricas

b. A utilização de chaves assimétricas potencia maior segurança face à utilização de chaves simétricas

c. Respostas a. e b.

d. Nenhuma das anteriores

Complemento de resposta:

Chaves Assimétricas:

- Mais poderoso ;
- Mais complexo;
- Menos preformante; **(alguma degradação de performance – a.)**
- Tipicamente utiliza-se modo assimétrico para encriptar chaves do modo simétrico, estas últimas usadas então na encriptação dos dados. (**maior segurança – b.**)

II. Classifique como Verdadeira (V) ou Falsa (F) cada uma das afirmações seguintes:

9. Um dos cenários que pode motivar o recurso à replicação será a necessidade de realizar um balanceamento do armazenamento e processamento dos dados.

Resposta: *V* - Utilizamos a replicação se:

Numa arquitetura completamente centralizada, se:

- Muitos acessos (concorrentes)
- Muitos dados persistidos
- Muitos dados trocados

Podemos ter problemas de:

- Performance
- Disponibilidade
- Manutenção

10. Um dos cenários que pode motivar o recurso à replicação será a necessidade de impor redundância ao sistema.

Resposta: *V* – Respondido na 9.

11. No contexto da replicação em MS SQL Server uma tabela pode ser replicada parcialmente filtrando que registos e que colunas participam na publicação

Resposta: *V*

12. No contexto da replicação em MS SQL Server uma view só pode pertencer a uma publicação, não podendo estar presente noutra(s) publicação

Resposta: *F* - As réplicas (dados replicados (tabelas,views,etc.))podem ser referentes a 1 ou mais publicações.

13. No contexto da replicação em MS SQL Server a modalidade snapshot é que apresenta menor desfasamento de dados.

Resposta: *F* - Quando utilizar snapshot:

- Com alterações substanciais, mas pouco frequentes dos dados;
- Os Subscribers necessitam de acesso a dados apenas em leitura;
- **Existe um grande desfasamento** nos períodos de atualização dos dados.

14. Num sistema de réplicas distribuídas de Base de Dados MongoDB, na modalidade de Replica Set o servidor Primário destina-se às operações de leitura de dados e o secundário às operações de escrita

Resposta: *F* – Na modalidade de Replica Set o servidor primário destina-se às operações de escrita e o secundário às operações de leituras de dados.

Replica set:

- Grupo de bases de dados que contêm os mesmos dados :
- Redundância e alta disponibilidade dos dados
- Separação entre servidor Primário e N Secundários
- Primários para escrita
- Secundários para leitura
- Solução para casos de falha em servidores de produção

15. Num sistema de réplicas distribuídas de Base de Dados MongoDB, na modalidade de Replica Set pode haver vários servidores Primários e vários servidores Secundários

Resposta: *F* **–** Apenas pode **haver um servidor Primário** e vários servidores Secundários.

16. Para aplicações transacionais as bases de dados NoSQL mais adequadas são as graph DB.

Resposta: *F* – Para aplicações transacionais as bases de dados NOSQL mais adequadas são as document.

17. Considere o código da Figura abaixo. Para os comandos de 1 a 4 indique Sucesso (S) ou Insucesso (I), de acordo com as permissões. Considere que o script é executado sequencialmente.

1- Insucesso

2- Sucesso

3- Sucesso

4- Insucesso

DENY != REVOKE

```
CREATE ROLE Schema1QueryRole;
ALTER ROLE Schema1QueryRole ADD MEMBER usr;

Execute as user='usr';
--1
Select * from schema1.table1;

revert;
GRANT select ON schema::schema1 TO Schema1QueryRole;

Execute as user='usr';
--2
Select * from schema1.table1;

revert;
GRANT select ON schema1.table1 TO usr;
REVOKE select ON schema::schema1 TO Schema1QueryRole;

Execute as user='usr';
--3
Select * from schema1.table1;

revert;
DENY select ON schema::schema1 TO Schema1QueryRole;

Execute as user='usr';
--4
Select * from schema1.table1;
```

18. Considerando a collection "Orders" no MongoDB na Figura A, em baixo, indique qual o resultado da execução do comando da Figura B.

Orders	
{ cust_id: "A123", amount: 500, status: "A" } { cust_id: "A123", amount: 250, status: "A" } { cust_id: "B212", amount: 200, status: "A" } { cust_id: "A123", amount:300, status: "D" }	db.orders.aggregate([$match stage → { $match: { status: "A" } }, $group stage → { $group: { _id: "$cust_id",total: { $sum: "$amount" } } }])
A	B

É efetuada uma agregação dos valores, filtrados por status (apenas ("A")), sendo que este agrupamento consiste numa soma ($sum). Também se encontra agrupado por id ($group).

Resposta:

ID : A123

Total: 750

ID: B212

Total: 200

19. Ordene indicando a sequência através das respetivas letras identificadoras, os enxertos de código da tabela em baixo de modo a que se possa executar os comandos:

SELECT Encrypt(myColumn) FROM myTable

SELECT Decrypt(myColumn) FROM myTable

A	CREATE SYMMETRIC KEY MySymmetricKeyName WITH ALGORITHM = AES_256, KEY_SOURCE = 'a very secure strong password or phrase' ENCRYPTION BY CERTIFICATE ACertificate;
B	CREATE CERTIFICATE ACertificate ENCRYPTION BY PASSWORD = 'certfpass' WITH SUBJECT = 'protect data' EXPIRY_DATE = '20210113'
C	EXEC OpenKeys
D	CREATE FUNCTION Encrypt (@ValueToEncrypt varchar(max)) RETURNS varbinary(256) AS BEGIN DECLARE @Result varbinary(256) SET @Result = EncryptByKey(Key_GUID('MySymmetricKeyName'), @ValueToEncrypt) RETURN @Result END
E	CREATE PROCEDURE OpenKeys AS BEGIN SET NOCOUNT ON; BEGIN TRY OPEN SYMMETRIC KEY MySymmetricKeyName DECRYPTION BY CERTIFICATE ACertificate END TRY BEGIN CATCH *-- Handle non-existent key here* END CATCH END
F	CREATE FUNCTION Decrypt (@ValueToDecrypt varbinary(256)) RETURNS varchar(max) AS BEGIN DECLARE @Result varchar(max) SET @Result = DecryptByKey(@ValueToDecrypt) RETURN @Result END
G	Create a Database Master Key CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'myStrongPassword'

Resposta:

G -> B -> E/D/F -> C