# **Exercícios de Fixação**

# **(1) Introdução – Conceitos Básicos**

**1)** Marque a alternativa **INCORRETA** sobre o SSMS (SQL Server Management Studio):

a) Os modos de autenticação podem ser via Windows ou SQL Server.

b) Não é possível conectar em instâncias do SQL Azure via SSMS.

c) Podemos executar comandos T-SQL.

d) É uma ferramenta GRATUITA da Microsoft.

e) Pode ser utilizada por DBAs, desenvolvedores e analistas de BI.

**2)** Marque a opção **INCORRETA** sobre o Configuration Manager e o SSMS:

a) No Configuration Manager podemos visualizar as instâncias que estão instaladas no servidor.

b) No Configuration Manager nós encontramos os serviços do SQL Server e podemos fazer um Start, Stop ou Restart.

c) No Configuration Manager podemos escrever os nossos códigos T-SQL.

d) No SSMS (SQL Server Management Studio) podemos escrever os nossos códigos T-SQL.

e) No SSMS (SQL Server Management Studio) podemos conectar via Windows Authentication ou SQL Server Authentication.

**3)** Marque a opção **INCORRETA** sobre as Versões do SQL Server:

a) Elas evoluem com o passar do tempo incluindo novas funcionalidades.

b) Podemos fazer o UPGRADE de uma versão mais antiga para uma mais atual normalmente.

c) Podemos fazer o DOWNGRADE de uma versão mais atual para uma mais antiga normalmente.

d) Alguns comandos T-SQL podem ser removidos (deprecated) de uma versão para a outra.

e) Alguns comandos T-SQL podem ser adicionados de uma versão para a outra.

**4)** Marque a opção **INCORRETA** sobre as Versões / Edições do SQL Server:

a) O Developer Edition é a mais recomendada para utilização em ambientes de teste / estudo, pois ela é gratuita. Contudo, não deve ser utilizada em um ambiente de produção (problemas de licenciamento).

b) O Enterprise Edition, apesar de ser a edição com mais features disponíveis ainda possui algumas limitações de hardware (memória e CPU).

c) As edições do SQL Server são: Enterprise, Standard, Web, Express e Developer.

d) O Standard Edition possui mais limitações do que o Enterprise Edition.

e) O comando para validar a versão / edição do SQL Server é o SELECT @@VERSION

**5)** Marque a opção **INCORRETA** sobre o Service Pack (SP) / Cumulative Update (CU):

a) É uma boa prática atualizar o SP / CU do banco de dados.

b) O SP / CU pode corrigir algumas features que não estão funcionando conforme o previsto.

c) O SP / CU pode incrementar a performance de alguma feature do SQL Server.

d) O SP / CU pode corrigir falhas de segurança no SQL Server.

e) Não é obrigatório atualizar o SP / CU, pois as melhorias são mínimas.

**6)** Marque a opção **CORRETA**:

(1) Uma instância do SQL Server pode conter vários servidores.

(2) Um servidor pode conter várias instâncias do SQL Server.

(3) Quando um servidor possui mais de uma instância, os objetos (banco de dados, usuários, etc.) de cada instância são compartilhados entre elas.

(4) Podemos ter mais de uma instância “default” em um servidor.

(5) Os recursos do servidor (CPU, memória e disco) são compartilhados por todas as suas instâncias.

a) F, V, F, F, V

b) F, V, F, F, F

c) F, V, V, F, V

d) V, F, V, F, V

e) V, V, F, V, V

**7)** Marque a opção **INCORRETA** sobre os Tipos de Comandos no SQL Server:

a) Os comandos CREATE, ALTER e UPDATE pertencem ao DDL (Data Definition Language).

b) Os comandos BEGIN TRAN, COMMIT e ROLLBACK pertencem ao TCL (Transaction Control Language).

c) Os comandos GRANT, DENY e REVOKE pertencem ao DCL (Data Control Language).

d) Os comandos SELECT, INSERT e DELETE pertencem ao DML (Data Manipulation Language).

**8)** Marque a opção **CORRETA** sobre os tipos de dados inteiros:

(1) Os tipos de dados BIT, TINYINT, SMALLINT, INT e BIGINT são utilizados para representar os números inteiros.

(2) O tipo de dados BIT é o menor e mais simples, pois pode representar apenas os valores 0, 1 ou NULL.

(3) O tipo de dados INT consegue representar todos os valores inteiros.

(4) Não precisamos nos preocupar com o tamanho dos tipos de dados, pois a diferença é de poucos bytes apenas.

(5) O tipo de dados INT utiliza mais bytes que os tipos SMALLINT e TINYINT, pois ele consegue representar mais valores.

a) F, V, F, F, F

b) F, V, V, V, F

c) V, V, F, V, V

d) V, V, F, F, V

e) V, F, F, V, V

**9)** Marque a opção **INCORRETA** sobre os tipos de dados numéricos e float:

a) O tipo de dados FLOAT utiliza números aproximados em sua representação.

b) O tipo de dados NUMERIC utiliza números exatos em sua representação.

c) Ao escolher um tipo de dados no SQL Server, os tipos DECIMAL e FLOAT não possuem muitas diferenças.

d) Ao escolher um tipo de dados no SQL Server, os tipos DECIMAL e NUMERIC não possuem muitas diferenças.

e) Podemos utilizar a declaração de uma variável com o tipo NUMERIC (P, S), onde P representa a quantidade total de dígitos e S a quantidade de números decimais (depois do ponto).

**10)** Marque a opção **CORRETA** sobre os tipos de dados de data e caracteres:

(1) Podemos utilizar o tipo de dados DATE para representar apenas a data e o tipo DATETIME para representar a data e a hora.

(2) Os tipos de dados VARCHAR e NVARCHAR possuem o mesmo tamanho em bytes por caractere.

(3) O tamanho (em bytes) do tipo de dados VARCHAR será apenas a quantidade de caracteres que realmente forem utilizados. No caso abaixo, irá utilizar apenas 4 bytes.

DECLARE @nome VARCHAR(100)

SET @nome = 'Luiz'

(4) Os tipos de dados que utilizam a tabela UNICODE (NCHAR e NVARCHAR) utilizam 1 byte por caractere. Já os tipos de dados da tabela ASCII (CHAR e VARCHAR) utilizam 2 bytes por caractere.

(5) A função GETDATE() retorna a data e hora atual.

a) F, F, V, F, F

b) F, F, V, V, V

c) V, F, V, F, V

d) V, V, V, F, F

e) V, F, F, F, V

**11)** Marque a opção **CORRETA**:

(1) SELECT 5 \* (2 + 3) -- RESULTADO = 25

(2) SELECT 5 + 2 \* 3 -- RESULTADO = 21

(3) SELECT 5 + (5 + (2 \* 2)) + 2 \* 10 -- RESULTADO = 160

(4) SELECT 5 / 2 -- RESULTADO = 2.5

(5) SELECT 10 \* 5 / 2 -- RESULTADO = 25

(6) SELECT 10 / 5 \* 2 -- RESULTADO = 1

a) F, V, F, V, F, F

b) V, V, F, V, V, V

c) V, F, V, V, V, V

d) V, F, F, V, V, V

e) V, F, F, F, V, F

**12)** Marque a opção **CORRETA** sobre Tabelas Temporárias:

(1) Uma tabela Temporária Local é visível apenas para a sessão onde foi criada. Já a tabela Temporária Global é visível para qualquer sessão no banco de dados, mas apenas com permissão de leitura.

(2) Não precisamos nos preocupar com a database “tempdb”, pois as tabelas temporárias são pequenas e com um curto período de vida.

(3) As tabelas temporárias são muito úteis quando precisamos armazenar valores temporariamente. Ex: dentro de um loop, para realizar algum cálculo, etc.

(4) Não existem muitas diferenças entre uma tabela Temporária Local e Global, pois ambas ficam armazenadas na database “tempdb”.

(5) As tabelas temporárias não são excluídas automaticamente pelo SQL Server. Dessa forma, sempre precisamos excluir explicitamente ao final do script. Ex: DROP TABLE #TEMP\_Cliente

a) F, F, V, F, F

b) F, V, V, F, F

c) V, F, V, F, V

d) V, F, V, V, F

e) V, V, F, V, V

**13)** Imagine uma tabela chamada “Cliente” com uma coluna “Nm\_Cliente” do tipo VARCHAR(100). Essa tabela utiliza a COLLATION “Latin1\_General\_CI\_AS”. Ela possui dois registros com os nomes “Fabrício Lima” e “Luiz Lima”.



Dado os SELECTs 1 e 2 abaixo, marque a resposta **CORRETA**:

-- SELECT 1

SELECT \*

FROM Cliente

WHERE Nm\_Cliente = 'Fabricio Lima'

-- SELECT 2

SELECT \*

FROM Cliente

WHERE Nm\_Cliente = 'luiz lima'

a) O SELECT 1 irá retornar registros, pois a COLLATION é AS (Accent Sensitive).

b) O SELECT 1 não irá retornar registros, pois a COLLATION é AS (Accent Sensitive).

c) O SELECT 2 não irá retornar registros, pois a COLLATION é AS (Accent Sensitive).

d) O SELECT 2 não irá retornar registros, pois a COLLATION é CI (Case Insensitive).

**14)** Dado o script abaixo, qual será o resultado?

DECLARE @People TABLE (Name VARCHAR(50));

BEGIN TRAN

INSERT INTO @People VALUES ('Bill Gates');

INSERT INTO @People VALUES ('Melinda Gates');

INSERT INTO @People VALUES ('Satya Nadella');

ROLLBACK

-- RESULTADO

SELECT \* FROM @People;

a) Irá retornar as 3 linhas.

b) Nenhuma linha será retornada.

c) Vai retornar somente a primeira linha.

d) Não sei, vou perguntar pro Dirceu Resende aqui.

**15)** Dado o script abaixo, qual será o resultado?

DROP TABLE IF EXISTS #People

CREATE TABLE #People (

Name VARCHAR(50)

);

BEGIN TRAN

INSERT INTO #People VALUES ('Bill Gates');

INSERT INTO #People VALUES ('Melinda Gates');

INSERT INTO #People VALUES ('Satya Nadella');

ROLLBACK

-- RESULTADO

SELECT \* FROM #People;

a) Irá retornar as 3 linhas.

b) Nenhuma linha será retornada.

c) Vai retornar somente a primeira linha.

d) Não sei, vou perguntar pro Fabiano Amorim aqui.