

AULA 02: SQL

SUMÁRIO	PÁGINA
1. Apresentação	1
2. Questões Comentadas	1
3. Lista das questões apresentadas	28
4. Gabarito	42

1. APRESENTAÇÃO

Ola amigos(as),

Sem mais delongas, vamos a nossa aula de SQL. Qualquer dúvida estou à disposição.

2. QUESTÕES COMENTADAS

1. (ESAF/Analista Sistemas/CVM 2010) Assinale a opção correta.

a) Programação de domínio é a associação de um domínio de valores possíveis a determinados atributos.

b) Uma Linguagem de Manipulação de Dados (DML) principal requer que o usuário especifique que dados prioritários são necessários.

c) Restrição dimensional é a associação dimensional a certos tipos de atributos.

d) Uma Linguagem de Manipulação de Dados (DML) procedural requer que o usuário especifique que dados são necessários e como obtê-los.

e) Uma Linguagem de Recuperação de Dados (DRL) requer que o usuário tenha conhecimento de programação em linguagem de máquina.

a) O domínio de um atributo determina o conjunto de valores que esse atributo pode ter. Assim, um atributo idade tem como domínio o conjunto dos inteiros, o atributo salário tem como domínio um valor real positivo, e assim sucessivamente. Existem diversas maneiras de se garantir que o valor de um atributo obedeça a sua restrição de domínio, algumas implementadas pelo próprio SGBD. Contudo, não existe programação de domínio, é invenção da banca. Item incorreto.

b) Para começar, não existe esse conceito de DML principal. DML (Data Manipulation Language) é simplesmente a linguagem de manipulação de dados e nada mais. A DML não requer que se especifique dados

prioritários, na verdade não existe uma hierarquia explícita entre os dados manipulados por DML.

c) Restrição dimensional não diz respeito a banco de dados, pois nos SGBDs tradicionais não existem dimensões dos dados. Item incorreto.

d) Linguagem procedural é o tipo mais comum de linguagem, onde se especifica os passos que um programa deve seguir para alcançar um resultado desejado. É diferente de uma linguagem funcional (geralmente usadas em IA) como o LISP, que trata a computação como uma avaliação de funções matemáticas e que evita estados ou dados mutáveis. A DML é uma linguagem predominantemente procedural onde realmente se especificam os dados (operação de projeção) e como obtê-los (operação de seleção). Item correto.

e) DRL (Data Recuperation Language), linguagem de recuperação de dados, é uma classificação criada para a parte da DML que cuida exclusivamente da recuperação de dados. Seria, na prática, apenas a instrução Select no SQL. Seu uso não requer conhecimento em programação de linguagem de máquina, requer apenas conhecimento da própria linguagem e da estrutura do BD.

Gabarito: Letra d

2. (ESAF/Analista Infraestrutura/CVM 2010) Na linguagem SQL

a) char(n): uma string de caracteres de tamanho fixo n, especificado pelo usuário.

b) floatchar(n): uma string de caracteres de tamanho variável máximo n, especificado pelo usuário.

c) var(n): um número de ponto variável, com precisão de até n dígitos.

d) close(n): uma string de aproximação de caracteres numéricos de tamanho fixo n, definido pela linguagem.

e) doublefloat(n): um número de ponto flutuante duplo, com precisão modificada.

Amigos, temos que ter uma coisa em mente, quando se trata de SQL. Existe um padrão chamado ANSI SQL, que digamos assim, é o SQL padronizado. Mas cada SGBD implementa de forma própria o SQL, principalmente quando se trata de tipos de dados e de funções. Portanto, essa questão, muito mal formulada por sinal, traz uma série de tipos de dados e pergunta qual é aceito em SQL. O único que é implementado em alguns SGBDs é o tipo char(n), representa uma string de tamanho n. Contudo, vale ressaltar que em ANSI SQL não existe esse tipo char(n). Os demais tipos não existem mesmo.

Gabarito: Letra a

3. (ESAF/Analista Infraestrutura/CVM 2010) Assinale a opção correta.

- a) A linguagem de definição de dados permite expressar as consultas e atualizações do banco de dados.**
- b) A linguagem de manipulação de dados permite a especificação do esquema do banco de dados.**
- c) A linguagem de manutenção de dados permite expressar as consultas e atualizações do banco de dados.**
- d) A linguagem de definição de dados permite a especificação do esquema do banco de dados.**
- e) A linguagem de manipulação de consultas permite a atualizações do banco de dados.**

- a) DDL é utilizada para definir o Banco de Dados. As consultas e atualizações são efetuadas pela DML.
- b) Inverteu novamente os conceitos. DDL é que especifica o esquema do BD.
- c) Linguagem de manutenção de dados não existe, o que existe é linguagem de manipulação de dados.
- d) Correto, conforme já comentei, a DDL especifica o esquema do Banco de Dados. Exemplo de comandos de DDL são Create Table e Drop Table.
- e) Linguagem de manipulação de consultas não é um termo utilizado em BD. O que ainda pode ser aceito é Linguagem de Recuperação de Dados (DRL) comentada na questão 1. E ela não permite atualizações no Banco de Dados.

Gabarito: Letra d

4. (ESAF/APOTI/MPOG 2010) Em uma SQL

- a) a Linguagem de Manipulação de Relacionamentos compreende os comandos para construir tabelas em um banco de dados.**
- b) a Linguagem de Definição de Dados fornece tabelas para criação e modificação de comandos.**
- c) os comandos básicos da Linguagem de Definição de Dados são Select, Insert, Update e Delete.**
- d) a Linguagem de Manipulação de Dados compreende os comandos para inserir, remover e modificar informações em um banco de dados.**
- e) os comandos básicos da Linguagem de Definição de Dados são Sort, Insert, Undo e Store.**

- a) A ESAF é bem criativa, não é? Agora colocou ai linguagem de manipulação de relacionamentos. A construção de tabelas no Banco de Dados é feita dela DDL. Item incorreto.

- b) Inverteu novamente os conceitos, a DDL não fornece tabelas para a construção de comandos, fornece comandos para a construção de tabelas.
- c) Os comandos Select, Insert, Update e Delete são os comandos básicos de DML, e não de DDL. Afinal de contas, eles servem para manipular os dados, e não para criar a estrutura do BD. Item Incorreto.
- d) Exatamente como já comentei. Item Correto.
- e) Entre os principais comandos da DDL estão Create Table, Create View, Create Index, Alter Table, Alter Index, Drop View etc etc. Item incorreto.

Gabarito: Letra d

5. (ESAF/Analista Desen Sistemas/ANA 2009) Em SQL, a cláusula check aplicada a uma declaração de domínio

- a) permite especificar um predicado que deve ser satisfeito por qualquer valor atribuído a uma variável de determinado domínio.**
- b) especifica um predicado que deve ser satisfeito por uma tupla em uma relação.**
- c) proíbe a inserção de um valor nulo para as variáveis do domínio.**
- d) verifica se os atributos considerados formam uma chave candidata.**
- e) não tem efeito, pois não se aplica esta cláusula a declarações de domínio.**

Conforme comentei a pouco, os próprios SGBDS implementam mecanismos para a garantir a integridade de domínio dos atributos. Uma maneira de se fazer essa conformação é através da cláusula check. Passo a seguir 2 exemplos, um na sintaxe do MySql e outro na sintaxe do SQL Server.

MySQL:

```
CREATE TABLE Persons  
(  
  P_Id int NOT NULL,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255),  
  Address varchar(255),  
  City varchar(255),  
  CHECK (P_Id>0)  
)
```

SQL SERVER:

```
CREATE TABLE Persons  
(  
  P_Id int NOT NULL CHECK (P_Id>0),  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255),
```

Address varchar(255),
City varchar(255)
)

Nos dois casos a cláusula check faz com que o campo P_Id só aceite valores maiores que zero. Ou seja, foi especificado um predicado que deve ser satisfeito para o atributo em questão.

Gabarito: Letra a

6. (ESAF/PSS TI 2008) Na linguagem SQL, o uso da cláusula WHERE, quando apropriado,

- a) agrupa o resultado em subconjuntos que possuem valores correspondentes em uma ou mais colunas. Em cada grupo não há duas linhas com o mesmo valor na(s) coluna(s) de agrupamento.**
- b) permite selecionar linhas baseadas em uma expressão booleana. Somente as linhas para as quais a expressão é avaliada como TRUE são retornadas no resultado.**
- c) especifica se o ResultSet de uma instrução SELECT simples, que atende aos requisitos para um cursor, é atualizável ou não.**
- d) permite selecionar colunas baseado em uma expressão numérica. Somente as colunas para as quais a expressão é avaliada como TRUE são retornadas no resultado. O uso desta cláusula obriga que os valores nulos sejam considerados diferentes nos agrupamentos.**
- e) cria uma ou mais linhas e as armazena na tabela especificada. O número de valores especificados na instrução WHERE deve ser idêntico ao número de colunas especificadas ou implícitas.**

- a) Quem faz agrupamento é o group by. Item incorreto.
- b) Exatamente, o where é seguido por uma expressão lógica (booleana) que é o critério de seleção das tuplas em uma tabela. É a operação de seleção em álgebra relacional.
- c) O ResultSet é tudo aquilo que é retornado de uma consulta. Um cursor é uma referência a determinada consulta. Por fim, o where não determina se os dados retornados são ou não atualizáveis.
- d) Quem seleciona as consultas é o Select (operação de projeção), e para essa seleção não é necessária a especificação de uma expressão lógica. Item totalmente incorreto.
- e) Where não cria linhas, apenas determina o critério de seleção das mesmas.

Gabarito: Letra b

7. (ESAF/Auditor Informática/Pref Natal 2008) Quanto à estrutura, propriedades e sintaxe da linguagem SQL, é correto afirmar que

- a) o CREATE e o DROP são comandos básicos da DDL (Data Definition Language - Linguagem de Definição de Dados).
- b) o SELECT e o INSERT são comandos básicos da DDL (Data Definition Language - Linguagem de Definição de Dados).
- c) o CREATE e o DROP são comandos básicos da DCL (Data Control Language - Linguagem de Controle de Dados).
- d) o GRANT e o REVOKE são comandos básicos da DDL (Data Manipulation Language - Linguagem de Manipulação de Dados).
- e) o SELECT e o INSERT são comandos básicos da DCL (Data Control Language - Linguagem de Controle de Dados).

- a) Conforme já estudamos, está correto.
- b) Select e Insert são da DML. Item incorreto.
- c) Conceito novo, DCL (Data Control Language) controla os aspectos de autorização dos dados e níveis de acesso aos objetos de um Banco de Dados. No fundo, é uma especialização da DDL. Seus principais comandos são Grant e Rvoke
- d) Como já sabemos, os comandos básicos da DDL são Create e Drop. Item incorreto.
- e) Select e Insert são comandos da DML. Item incorreto.

Gabarito: Letra a

8. (ESAF/Auditor Informática/Pref Natal 2008) Com relação às características da linguagem SQL e dos SGBD, é correto afirmar que

- a) os "Triggers" são utilizados para auxiliar a manutenção da consistência dos dados. Também podem ser utilizados para propagar alterações ocorridas em um determinado dado de uma tabela para outra tabela.
- b) considerando o SQL_ANSI, o código a seguir é um exemplo correto para a criação de um Trigger relacionado a ações na tabela "cadastro": `CREATE TRIGGER trigger_cpf AS Codigo a ser executado FOR UPDATE ON cpf.cadastro`
- c) os "Triggers" são relações que não armazenam dados, mas são definidos dinamicamente por uma consulta a uma tabela previamente analisada e otimizada.
- d) os "Triggers" são relações que armazenam dados, definidos dinamicamente por uma regra de inserção a uma tabela.
- e) os "Triggers" são um arquivo auxiliar associado a uma ou mais tabelas. Sua função é acelerar o tempo de acesso aos dados de uma tabela.

- a) Já descrevi em aulas passadas que as triggers são trechos de código disparados quando algum evento ocorre em determinada tabela, como por exemplo a inserção, exclusão ou alteração de um registro. O fato de

serem executadas nesses eventos, as vezes antes do encerramento do evento, as vezes depois, faz com que as triggers possam ser usadas para a manutenção da consistência dos dados, ou seja, pode-se usar as triggers para a validação dos mesmos. Outro uso seu é a replicação automática de uma informação em outro lugar do Banco de Dados, ou da aplicação como um todo. Imaginem que existe a necessidade levantada no projeto de toda vez que uma Nota Fiscal for inserida, a mesma seja gravada além da tabela principal (onde ela normalmente está) em outra(s) tabela(s). Ou que toda vez que um registro é alterado ou deletado, um logo seja gerado. As triggers podem auxiliar nessa tarefa. Item correto.

b) Não existem comandos para a definição de triggers em ANSI SQL. Cada SGBD cria a sua própria implementação.

c) e d) Triggers não são relações, são trechos de código, sub-rotinas por assim dizer.

e) Triggers também não são arquivos.

Gabarito: Letra a

9. (ESAF/AFC/CGU 2008) Em um banco de dados que utiliza a linguagem SQL para definição, manipulação e controle de dados, é correto afirmar que os comandos

a) CREATE, DROP e INSERT fazem parte da DML (Linguagem de Manipulação de Dados).

b) GRANT e REVOKE fazem parte da DCL (Linguagem de Controle de Dados).

c) INSERT, UPDATE e SELECT fazem parte da DDL (Linguagem de Definição de Dados).

d) ALTER, DELETE e REVOKE fazem parte da DML (Linguagem de Manipulação de Dados).

e) CREATE, GRANT e DROP fazem parte da DCL (Linguagem de Controle de Dados).

a) Create e Drop fazem parte da DDL, e Insert da DML. Item incorreto.

b) Item correto. Temos na DCL o controle de acesso dos usuários aos objetos. Isso é feito pelos comandos Grant (que dá uma permissão) e Rovoque (revoga uma permissão).

c) Fazem parte da DML.

d) Entre os citados, só Delete faz parte da DML.

e) Entre os citados, só Grant faz parte da DCL. Create e Drop fazem parte da DDL.

Gabarito: Letra b

10. (ESAF/AFC/CGU 2008) Uma subquery (subconsulta) é um comando SELECT que foi "embutido" em outro comando

SELECT, UPDATE, DELETE ou dentro de outra subquery. A finalidade da subquery é retornar um conjunto de linhas para a query (consulta) ou comando principal. Com relação às subqueries, é correto afirmar que

- a) uma subquery não precisa estar incluída entre parêntesis.**
- b) uma subquery sempre deve estar do lado esquerdo do operador de comparação.**
- c) uma subquery pode conter a cláusula ORDER BY.**
- d) o operador IN não pode ser utilizado em uma subquery que retorne múltiplas linhas.**
- e) o operador igual "=" não pode ser utilizado em uma consulta que contenha uma subquery que retorne múltiplas linhas.**

a) As subqueries devem estar entre parêntesis por padrão, apesar de alguns SGBDs poderem aceitar uma subquery sem os parêntesis. Algumas perguntas quando se trata de SQL devem ser vistas com cuidado, pois quando a banca não especifica se estamos tratando ou não de ANSI SQL, fica complicado julgar, pois os SGBDs implementam certos recursos de formas diferentes. Item incorreto.

b) Um subquery normalmente está do lado direito do operador de comparação. Em alguns casos pode até estar do lado esquerdo, ou mesmo não ser usada em uma comparação (uma subquery após o from). Mas a questão está errada por afirmar que uma subquery sempre deve estar do lado esquerdo de um operador de comparação.

c) Item correto, uma subquery pode conter um order by.

d) Pode e normalmente é usado. O operador IN testa se determinado valor está em um conjunto, que pode ser retornado por uma subquery.

e) Bem, a banca considerou esse item incorreto, mas na minha opinião está correto, pois nas implementações que eu conheço, o operador igual não é um operador de conjunto. Assim, em uma subquery que retorne um ResultSet, caberia o uso do In e do Exists, mas o igual é mais complicado aceitar. Tudo bem, como sempre digo para vocês, não adianta brigar com a banca. Na pior das hipóteses, partindo para outra interpretação do enunciado, posso dizer que o operador igual pode ser usado em uma query que contenha uma subquery, contanto que não seja usado diretamente em uma operação com a subquery. Pode ser que seja isso que a banca considerou.

Gabarito: Letra c

11. (ESAF/AFC/CGU 2008) Considere as seguintes consultas envolvendo dados de duas tabelas (T1 e T2) que possuem o atributo CODIGO em comum e assinale a opção correta.

Consulta 1

SELECT * FROM T1 WHERE CODIGO IN (SELECT CODIGO FROM T2)

Consulta 2

SELECT * FROM T1 WHERE NOT EXISTS (SELECT CODIGO FROM T2 WHERE T1.CODIGO = T2.CODIGO)

Consulta 3

SELECT T1.* , T2.* FROM T1 INNER JOIN T2 ON T1.CODIGO = T2.CODIGO

Consulta 4

SELECT * FROM T1 WHERE CODIGO NOT IN (SELECT CODIGO FROM T2)

Consulta 5

SELECT T1.* FROM T1 LEFT OUTER JOIN T2 ON T1.CODIGO = T2.CODIGO WHERE T2.CODIGO IS NULL

Consulta 6

SELECT T1.* , T2.* FROM T1 , T2 WHERE T1.CODIGO = T2.CODIGO

- a) O resultado da consulta 3 é igual ao resultado da consulta 4.
- b) O resultado da consulta 2 é igual ao resultado da consulta 6.
- c) O resultado da consulta 2 é igual ao resultado da consulta 5.
- d) O resultado da consulta 5 é igual ao resultado da consulta 6.
- e) O resultado da consulta 1 é igual ao resultado da consulta 5

Vamos analisar uma consulta por vez:

Consulta 1: Retorna as linhas de T1 em que o campo Codigo também existe em T2.

Consulta 2: Retorna linhas em que Codigo existe em T1 mas não existe em T2.

Consulta 3: Retorna linhas em que o Codigo é igual em T1 e T2.

Consulta 4: Retorna linhas de T1 em que o campo Codigo existe em T2.

Consulta 5: Retorna linhas em que Codigo existe em T1 mas não existe em T2.

Consulta 6: Faz o produto cartesiano das tabelas e retorna as linhas em que os códigos são iguais.

Portanto, as consultas 2 e 5 são equivalentes.

Gabarito: Letra c

12. (ESAF/Analista de TI/SEFAZ-CE 2007) Na linguagem SQL, os procedimentos gravados para serem executados implicitamente quando ocorrer determinada ação do usuário, como, por exemplo, a modificação de uma tabela, são denominados

- a) Selects.
- b) Inserts.
- c) Views.
- d) Queries.
- e) Triggers.

Questão recorrente da ESAF, e vejam como foi cobrada em concursos importantes. Como já vimos e revimos, a questão trata dos triggers. Vale ressaltar que alguns SGBDs têm em sua implementação triggers que são relacionadas com eventos do SGBD, e não das tabelas especificamente. Normalmente são denominados triggers de sistema.

Gabarito: Letra e

13. (ESAF/Analista de TI/SEFAZ-CE 2007) No SQL, considerando o uso das cláusulas SELECT, GROUP BY e HAVING, indique a opção que apresenta uma sintaxe correta.

- a) **SELECT Ano, COUNT(*) AS 'Total' FROM Matricula WHERE Cod = 171 GROUP BY Ano HAVING count(*) > 5**
- b) **SELECT Ano, COUNT(*) AS 'Total' FROM Matricula HAVING count(*) > 5 WHERE Cod = 171 GROUP BY Ano**
- c) **SELECT Ano, COUNT(*) AS 'Total' WHERE Cod = 171 FROM Matricula GROUP BY Ano HAVING count(*) > 5**
- d) **SELECT Ano, COUNT(*) AS 'Total' WHERE Cod = 171 GROUP BY Ano HAVING count(*) > 5 FROM Matricula**
- e) **SELECT Ano WHERE Cod = 171 GROUP BY Ano HAVING count(*) > 5 COUNT(*) AS 'Total' FROM Matricula**

Para resolvermos essa questão, devemos conhecer a sintaxe do Select com o uso do Group By e Having. Funciona assim:

Select <campos/expressões> **from** <tabelas/subquery> **where** <condição> **group by** <campos de agrupamento> **Having** <condição>
Agora, podemos facilmente identificar que a letra (a) está com a sintaxe correta, e as demais estão com cláusulas na ordem incorreta.

Gabarito: Letra a

14. (ESAF/Analista de Sistemas/ENAP 2006) Analise as seguintes afirmações relacionadas a Gerenciamento de Banco de Dados.

- I. A *Data Definition Language* (DDL) é composta, entre outros, pelo comando *Create*, que é utilizado para criar bancos de dados, tabelas e relações existentes entre as tabelas.**
- II. Os comandos da *Data Manipulation Language* (DML) são destinados a consultas, inserções, exclusões e alterações em um ou mais registros de uma ou mais tabelas de maneira simultânea.**
- III. Os comandos *Truncate* e *Delete* são exemplos de comandos da *Data Definition Language* (DDL).**
- IV. Os comandos *Alter*, *Drop* e *Rename* são exemplos de comandos da subclasse de comandos da *Data Manipulation Language* (DML) chamada DCL (*Data Control Language*) .**

Indique a opção que contenha todas as afirmações verdadeiras.

- a) II e III**
- b) I e II**
- c) III e IV**
- d) I e III**
- e) II e IV**

I. O comando Create faz parte da DDL, e pode ser usado para criar tabelas, bancos de dados e relações entre tabelas (chaves estrangeiras). Item correto.

II. Como já vimos, os comandos da DML são destinados para consultas, inserções, exclusões e alterações (Select, Insert, Delete e Update). Item correto.

III. O Truncate limpa a tabela, deixando-a sem registros. Normalmente ele é implementado para que obedeça às constraints, ou seja, ele deve manter a integridade referencial. Outra característica do Truncate é não gerar log (undo), não podendo a operação ser desfeita. Tanto Truncate como Delete são da DML. Item incorreto.

IV. Data Control Language não incorpora Alter, Drop e Rename, que são da DDL. Item incorreto.

Gabarito: Letra b

15. (ESAF/Analista de Sistemas/ENAP 2006) Quanto ao uso de diagramas da linguagem SQL é correto afirmar que os comandos *Grant* e *Revoke*

- a) são usados para alterar estruturas de tabelas e de seus atributos.**
- b) possibilitam incluir o nome dos atributos e das tabelas.**
- c) não permitem retrocessos e por isso são classificados como *Data Definition Language* (DDL).**
- d) não permitem retrocessos e por isso são classificados como *Data Manipulation Language* (DML).**
- e) são responsáveis por dar ou retirar permissões para os usuários.**

a) Para a alteração das estruturas de tabelas é usado o comando Alter.

b) Não servem para incluir nomes de atributos. São usados para atribuir permissões.

c) Na verdade a Data Control Language (DCL) é um subtipo de DDL. Isso partindo da classificação original achada em muitos livros, que definem apenas DDL e DML. Dentro desse contexto lato sensu, o Grant e Revoke são DDL, pois definem o acesso dos usuários aos objetos do Banco de Dados. Mas a questão está errada quando afirma que não permitem retrocesso e por isso são classificadas como DDL. Alias, não se pode dizer

que eles não permitem retrocessos, afinal, o que um comando faz o outro pode desfazer.

d) Não fazem parte da DML.

e) Item correto. Vale ressaltar que a forma de implementação do Grant e Revoke variam em cada SGBD. No MySQL, por exemplo, os privilégios podem ser concedidos em quatro níveis: (1) Global, aplicando-se a todos os Banco de Dados de um determinado servidor; (2) Banco de Dados, aplicando a todas as tabelas em um determinado Banco de Dados; (3) Tabela, aplicando-se a todas as colunas de determinada tabela; (4) Coluna, aplicando-se a uma coluna de determinada tabela.

Gabarito: Letra e

16. (ESAF/Analista de Sistemas/ENAP 2006) A SQL possui recursos para apagar tabelas e bancos de dados a partir do comando

a) CREATE TABLE NULL.

b) TRUNCATE PRIMARY KEY.

c) DROP.

d) INSERT INTO TABLE NULL.

e) DELETE TABLE NULL.

Para vocês verem que até a ESAF já cobrou questões bem simples. O Comando para apagar tabelas e bancos de dados é o Drop.

Gabarito: Letra c

17. (ESAF/Analista Técnico/SUSEP 2006) Analise as seguintes afirmações relacionadas a conceitos básicos sobre Banco de Dados.

I. O comando SQL responsável por fechar uma transação confirmando as operações feitas é o INSERT.

II. O comando SQL responsável por fechar uma transação e desfazer todas as operações é o COMMIT.

III. Quando uma transação ainda está aberta para um usuário, enquanto não é executado um comando COMMIT, o próprio usuário pode ver as suas alterações, mas outros usuários não podem vê-las.

IV. Uma transação assegura um espaço de trabalho que contém várias alterações, inclusões e exclusões de dados em uma ou mais tabelas, com a possibilidade de confirmação ou cancelamento das operações sem comprometimento dos dados.

Indique a opção que contenha todas as afirmações verdadeiras.

a) I e II

b) II e III

c) III e IV

- d) I e III**
- e) II e IV**

I. O comando que fecha a transação e confirma os valores é o Commit. Item incorreto.

II. O Commit não desfaz as operações. Quem desfaz é o RollBack. Item incorreto.

III. Exato, as alterações não são propagadas até que o Commit feche a transação, mas se o usuário que fez uma atualização, por exemplo, antes do Commit executar um Select, vai ver os dados já alterados. Os demais usuários não vêem a alteração.

IV. É uma boa definição para transação, correta em todos os sentidos.

Gabarito: Letra c

18. (ESAF/Analista Informática/IRB 2006) Considere um banco de dados com uma tabela com o nome "Carro", onde estão cadastrados os mais de 1000 veículos de uma empresa. Esta tabela contém um campo de nome "Cor", que pode ser nulo, contendo a informação da cor de cada veículo cadastrado. Ao se executar a instrução SQL ALTER TABLE Carro DROP COLUMN Cor

a) a tabela Carro será totalmente apagada devido à execução do comando DROP.

b) todos os registros preenchidos com o valor Null serão alterados.

c) todos os registros preenchidos com o valor diferente de Null serão alterados.

d) todas as informações das cores dos veículos contidas neste campo serão perdidas.

e) nada irá acontecer com os dados da tabela Carro.

a) Não, o que está sendo apagado (dropado) é a coluna, e não a tabela como um todo.

b) Quando se apaga uma coluna o não importa o conteúdo de cada registro, pois todos serão alterados, no sentido de não terem mais aquela coluna. Item incorreto.

c) Mesmo raciocínio da letra (b), pouco importa se a coluna apagada tinha valor null ou qualquer outro.

d) Exatamente, as informações dessa coluna são apagadas, e portanto perdidas.

e) O Alter Table é um comando da DDL, portanto altera a estrutura da tabela, e alterando essa estrutura há um reflexo nos dados da tabela. Nesse caso, os dados de determinada coluna deixam de existir.

Gabarito: Letra d

19. (ESAF/Analista Informática/IRB 2006) Analise as seguintes afirmações relacionadas a banco de dados e ao uso da linguagem SQL.

I. A cláusula DISTINCT é usada com o SELECT nos casos em que se deseja eliminar as tuplas repetidas deixando apenas uma tupla de cada valor no resultado da consulta.

II. As funções UPPER, LOWER e REPLACE são exemplos de funções para manipulação de chaves primárias.

III. Considere que em um banco de dados uma tabela chamada "Carro" contenha os campos CodCarro, Descricao e Cor.

Condiderando, ainda, que os campos Descricao e Cor podem receber o valor Null, a execução do comando SQL SELECT CodCarro, Descricao, NVL(Cor,'Branca') FROM Carro WHERE Cor is NULL caso retorne registros, todos os valores do campo Cor destes registros estarão preenchidos com o valor Branca.

IV. A função COUNT faz a soma dos valores das linhas retornadas em uma consulta.

Indique a opção que contenha todas as afirmações verdadeiras.

- a) I e II**
- b) II e III**
- c) III e IV**
- d) I e III**
- e) II e IV**

I. Exatamente. O Distinct é usado para eliminar repetições em um ResultSet, retornando apenas uma vez cada valor. Por exemplo, imaginem uma tabela que contenha o ISBN de todos os livros que todos os alunos do EstrategiaConcursos estão utilizando para sua preparação para a CGU. Muito provavelmente alguns livros são usados por mais de um aluno, então teremos ISBNs repetidos nessa tabela. Se eu quiser saber quais são todos os livros utilizados por algum aluno da preparação para a CGU, eu posso usar o comando a seguir para obter somente os ISBNs dos livros, sem repetição:

Select Distinct(ISBN) from LivrosDosAlunos (abstraindo-se os metadados).

II. Não, essas funções manipulam strings. O Upper transforma para maiúsculo, o Lower para minúsculo e o Replace faz a substituição de caracteres. Item incorreto.

III. Primeiramente analisando a cláusula where, verificamos que apenas tuplas em que o atributo cor é null são retornadas. Pois bem, a projeção pede CodCarro, Descricao e NVL (Cor, 'Branca'). A função NVL faz o seguinte: se o atributo Cor for igual a null, o seu retorno é o segundo parâmetro, nesse caso, a string 'Branca'. Se o atributo for diferente de null, o seu valor é retornado. Assim, a afirmativa esta correta, pois caso

sejam retornadas tuplas, todos os campos do atributo Cor terão o valor Branca.

IV. O Count não faz a soma de valores, simplesmente retorna o número de linhas do Select. Para a soma de valores, deve-se utilizar o Sum.

Gabarito: Letra d

20. (ESAF/Analista Informática/IRB 2006) Analise as seguintes afirmações relacionadas a banco de dados e ao uso da linguagem SQL.

I. A cláusula ORDER BY permite criar consultas com as linhas em uma ordem específica, modificando a ordem de apresentação do resultado da pesquisa. As linhas podem ser ordenadas de forma ascendente ou descendente de acordo com as colunas declaradas na cláusula ORDER BY.

II. Os dois comandos SELECT apresentados a seguir são equivalentes.

SELECT *

FROM Carro

**WHERE CodCarro <> ANY (SELECT CodCarro
FROM Vendido).**

SELECT *

**FROM Carro WHERE CodCarro NOT IN (SELECT
CodCarro FROM Vendido).**

III. O comando SQL abaixo retornará a relação dos carros cujo código CodCarro seja igual a 51.

SELECT Ano, COUNT(*) AS Total de Carros

FROM Veiculos

GROUP BY Ano

HAVING CodCarro = 51

IV. A cláusula GROUP BY pode ser colocada antes ou depois da cláusula HAVING. Se ela for colocada antes, os grupos são formados e as funções de grupos são calculadas antes de resolver a cláusula HAVING. Se for colocada depois, a cláusula HAVING filtra a seleção antes da formação dos grupos e do cálculo das funções.

Indique a opção que contenha todas as afirmações verdadeiras.

a) I e II

b) II e III

c) III e IV

d) I e III

e) II e IV

I. Exato, o order by pode ser usado para ordenar um ResultSet. Além disso, a ordenação pode ser por mais de uma coluna, e cada coluna

especificada pode ter as cláusulas ASC e DESC, que ordenam em ordens ascendente ou decendente, respectivamente,

II. Nos dois comandos Select há uma subquery que retorna os valores de CodCarro da tabela Vendido. E ambos mostram as tuplas da tabela Carro, contando que seu código não esteja na tabela Vendido. Assim <> (diferente) ANY e NOT IN exercem nesse caso a mesma função.

III. Na verdade este Select deveria estar escrito da seguinte forma:

```
SELECT Ano, COUNT(*) AS Total de Carros
```

```
FROM Veiculos
```

```
Where CodCarro = 51
```

```
GROUP BY Ano
```

Lembrem-se que o Having é usado para tratar das funções de agregação (nesse caso, o Count)

IV. O Group by é colocada antes do Having. Item incorreto.

Gabarito: Letra a

21. (ESAF/AFC/CGU 2006) Um procedimento armazenado (stored procedure) é uma coleção de comandos em SQL que

a) provoca um aumento no tráfego na rede e reduz a performance do sistema, mas continua sendo largamente utilizado para criar mecanismos de segurança em bancos de dados relacionais.

b) encapsula tarefas repetitivas, aceita parâmetros de entrada e pode retornar um valor de status para indicar sucesso ou falha na execução.

c) estão armazenados no banco de dados e que são executadas diretamente na máquina do usuário.

d) estão armazenados na máquina do usuário e que são executadas diretamente no servidor do banco de dados.

e) são utilizados unicamente para autenticar um usuário, dando a ele direitos de acesso a escrita/alteração em tabelas do banco de dados.

a) Não se pode dizer que as stored procedures aumentam o tráfego na rede e reduzem a performance do sistema. Na verdade, o que acontece normalmente é exatamente o contrário. Por estarem armazenadas no próprio Banco de Dados, os códigos executados pelas Stored Procedure evitam que esse dados brutos percorram a rede. A performance pode até ser melhorada, se esses dados forem processados na próprio servidor do SGBD ao invés de serem transportados a um servidor de aplicação para processamento. O grande problema de se fazer isso em larga escala é que as regras de negócio do seu sistema ficariam presas ao Banco de Dados, indo se encontro ao projeto em camadas, e fazendo sua aplicação mais dependente daquela plataforma. Por exemplo, imaginem se toda a regra de negócio de sua aplicação estiver armazenada em Stored

Procedures de uma SGBD SQL Server, por exemplo, e sua empresa decide que agora vai utilizar somente o MySQL. O que acontece? As procedures deverão ser reescritas para rodar no outro banco de dados, já que muitas vezes as implementações de SQL são bem diferentes. Então, esse item está incorreto.

b) Exatamente, as Stored Procedures podem encapsular tarefas repetitivas. Elas podem aceitar parâmetros de entradas e devolver o status de sua execução.

c) Estão armazenadas no Banco de Dados e são executadas no próprio servidor do SGBD, e não na máquina do usuário.

d) Como já vimos, item incorreto.

e) Podem ser usadas para autenticar usuários, mas não somente para isso. Podem executar em princípio qualquer comando SQL, e executar uma infinidade de funções.

Gabarito: Letra b

22. (ESAF/AFC/CGU 2006) Analise as seguintes afirmações relacionadas a conceitos básicos de banco de dados e linguagem SQL.

I. Na linguagem SQL um INNER JOIN retorna todas as tuplas comuns às duas tabelas.

II. Em uma Junção entre duas tabelas a cláusula USING só poderá ser usada quando o nome do atributo for igual nas duas tabelas.

III. Na linguagem SQL um RIGHT OUTER JOIN retorna todas as tuplas que não são comuns às duas tabelas.

IV. Uma Junção é usada para compor informações complexas a partir de tabelas sem nenhum tipo de relacionamento.

Indique a opção que contenha todas as afirmações verdadeiras.

a) I e III

b) II e III

c) III e IV

d) I e II

e) II e IV

I. Isso mesmo, o Inner Join vai retornar as tuplas que combinam nas duas tabelas.

II. Exatamente, o Using é uma forma de escrever um comando mais enxuto. Vou me valer do seguinte exemplo extraído do site da IBM:

```
SELECT EMPNO, ACSTDATE
FROM CORPDATA.PROJACT INNER JOIN CORPDATA.EMPPROJACT
USING (PROJNO, ACTNO)
WHERE ACSDATE > '1982-12-31';
```

É equivamente a:

```
SELECT EMPNO, ACSTDATE
FROM CORPDATA.PROJACT INNER JOIN CORPDATA.EMPPROJACT
ON CORPDATA.PROJACT.PROJNO =
CORPDATA.EMPPROJACT.PROJNO AND
CORPDATA.PROJACT.ACTNO = CORPDATA.EMPPROJACT.ACTNO
WHERE ACSTDATE > '1982-12-31';
```

Claro, se não existirem os campos projno e actno nas duas tabelas, não tem como utilizar o Using.

III. RIGHT OUTER JOIN é uma Operação JOIN que permite especificar a cláusula de junção. Preserva as linhas sem correspondência da segunda tabela (direita), juntando-as com uma linha nula na forma da primeira tabela (esquerda). (A LEFT OUTER JOIN B) é equivalente a (B RIGHT OUTER JOIN A), com as colunas em uma ordem diferente. Portanto, afirmativa incorreta.

IV. Se não houver relacionamento entre as tabelas, vai fazer junção de que? Item incorreto.

Gabarito: Letra d

23. (ESAF/TRF/SRF 2006) Analise as seguintes afirmações relacionadas a Bancos de Dados e à linguagem SQL:

I. A cláusula *GROUP BY* do comando *SELECT* é utilizada para dividir colunas em conjuntos maiores de dados relacionados entre si.

II. Uma *view* é uma forma predeterminada de visualizar dados de uma ou mais tabelas como se fosse apenas uma tabela.

III. Quando o comando *DROP TABLE* é executado, as estruturas da tabela e os dados são excluídos. Porém, quando o *DROP VIEW* é executado, nenhum dado referenciado pela *view* é afetado.

IV. O *trigger* é um tipo de *view* criado quando um evento em particular ocorre. Indique a opção que contenha todas as afirmações verdadeiras.

- a) I e II
- b) III e IV
- c) II e III
- d) I e III
- e) II e IV

I. O Group By é utilizado para agrupamentos de dados em determinado critério, que pode ser uma campo, um conjunto deles etc. Ele não é usado para dividir colunas em grupos maiores.

II. Exatamente, através de uma View podemos visualizar dados que estão em uma ou diversas tabelas. A View é uma unidade lógica, não gera um objeto físico (apesar de alguns Bancos de Dados gerarem o que se chama

Materialized View). Ela é criada especificando um comando Select com acesso a uma ou mais tabelas.

III. Isso mesmo, Drop Table apaga a tabela, com a estrutura e todos os seus dados. Como citei, uma view é apenas uma estrutura lógica, assim o Drop View em nada afeta a(s) tabela(s) que faz referência.

IV. A Trigger, como já vimos, não é um tipo de view, mas sim uma espécie de procedure.

Gabarito: Letra c

24. (ESAF/TRF/SRF 2006) Em um Banco de Dados Relacional, utilizando-se a linguagem SQL para a manipulação de dados, é correto afirmar que

a) para se efetuar a JUNÇÃO entre tabelas é essencial que não exista relacionamento entre elas.

b) a DML (*Data Manipulation Language*) é responsável pela manipulação de estruturas de dados, sendo composta por comandos destinados à criação, alteração e exclusão de bancos de dados e tabelas. Os comandos *Create*, *Alter*, *Drop* e *Rename* são exemplos de comandos da classe DML.

c) na JUNÇÃO entre tabelas, a chave estrangeira da entidade "1" no relacionamento "1 : N" deve ser exportada como chave primária da entidade "N".

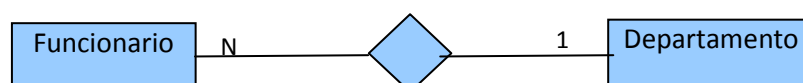
d) uma transação é uma unidade lógica que contém um conjunto de operações que formam uma unidade de processamento. O comando SQL responsável por fechar uma transação, confirmando as operações feitas é o *COMMIT*. Para desfazer todas as operações, o comando é *ROLLBACK*.

e) na JUNÇÃO entre tabelas, a cláusula *ON* só poderá ser usada quando o nome do atributo for igual nas duas tabelas. Quando os nomes dos atributos tiverem valores diferentes, a cláusula a ser usada será a *USING*.

a) A afirmação é falsa. É exatamente o contrário, pois não faz sentido fazer junções de tabelas se elas não têm relacionamento, ou seja, itens em comum.

b) Os comandos citados e as operações descritas (criação, alteração e exclusão de bancos de dados e tabelas) são da DDL e não da DML.

c) Vamos exemplificar para ficar mais fácil. Imaginem um relacionamento entre a Tabela Funcionario e a Tabela Departamento. Um funcionário é lotado em um departamento, e um departamento tem N funcionários. Assim, a Tabela Departamento é a entidade 1, e a Tabela Funcionário é a entidade N, correto?



Assim, o correto é afirmar que a chave primária da entidade 1 (Tabela Departamento) deverá ser exportada como chave estrangeira para a entidade N (Tabela Funcionario). Item incorreto.

d) Conforme já comentamos, o conceito de transação está correto. Normalmente se utilizam transações quando temos uma série de procedimentos para serem finalizados, e estes devem ocorrer na sua totalidade. Ou seja, se pelo menos um dos procedimentos der errado, os demais devem retornar ao estado anterior. Imaginem se não existissem as transações, vocês teriam coragem de sacar dinheiro no caixa eletrônico?? Nesse caso com certeza são utilizadas transações, pois caso o estado final não se confirme (o dinheiro sair do caixa), o valor não será debitado da conta.

e) É o contrário, quando os nomes forem diferentes temos que utilizar o ON, e quando forem iguais podemos utilizar o USING.

Gabarito: Letra d

25. (ESAF/AFRF/SRF 2005) SQL é uma linguagem de consulta, criada para extrair, organizar e atualizar informações em bancos de dados relacionais. Com relação aos conceitos básicos de SQL, é correto afirmar que a instrução COMMIT

- a) realiza todas as alterações feitas até o ponto em questão.**
- b) altera a definição de uma tabela ou de outro objeto de banco de dados.**
- c) elimina uma tabela ou outro objeto de um banco de dados.**
- d) desfaz todas as alterações feitas até o ponto em questão.**
- e) define um cursor, que pode ser utilizado para processar linhas acessadas por uma consulta. A instrução COMMIT inclui uma instrução SELECT, que é utilizada para processar a consulta.**

A instrução Commit confirma e fecha determinada transação, realizando todas as operações feitas até determinado ponto, como está descrito na alternativa (a)

Gabarito: Letra a

26. (ESAF/AFRF/SRF 2005) Com relação ao uso da SQL na manipulação de dados, caso se queira eliminar linhas repetidas do conjunto resultado, deve-se utilizar a palavra-chave DISTINCT, da seguinte forma:

- a) SELECT {colunas} FROM {tabelas} DISTINCT.**
- b) DISTINCT SELECT {colunas} FROM {tabelas}.**
- c) SELECT FROM {tabelas} DISTINCT {colunas}.**

- d) SELECT DISTINCT {colunas} FROM {tabelas}.**
e) FROM {tabelas} SELECT DISTINCT {colunas}.

Para resolver essa questão, da Receita Federal, basta saber a sintaxe do operador DISTINCT. Bem fácil essa, não?

Gabarito: Letra d

27. (FCC/Analista de Sistemas/COPERGAS 2011) São respectivos

a DML, DDL, DCL e DTL:

- a) Update, Drop, Grant e Commit.**
b) Select, Revoke, Delete e Rollback.
c) Delete, Alter, Insert e Drop.
d) Insert, Commit, Revoke e Select.
e) Alter, Update, Revoke e Rollback.

- a) Update: DML; Drop: DDL; Grant DCL; Commit: DTL
b) Select: DML; Revoke: DCL; Delete: DML; Rollback: DTL.
c) Delete: DML; Alter: DDL; Insert: DML; Drop: DDL.
d) Insert: DML; Commit: DTL; Revoke: DCL; Select: DML.
e) Alter: DDL; Update: DML; Revoke: DCL; Rollback: DTL.
Assim, pela sequência pedida, temos a letra (a) como resposta

Gabarito: Letra a

28. (FCC/ACE-TI/TCE-SE 2011) Durante a criação de uma tabela - Create Table, em SQL, deseja-se especificar que uma coluna só possa incluir, por exemplo, valores maiores que zero. Uma constraint utilizada para isso é

- a) Verify.**
b) Check.
c) Max.
d) Avg.
e) Having.

O check é utilizado com o objetivo de colocar uma restrição (constraint) para ser checada em determinado campo. Olhem o exemplo abaixo:

```
CREATE TABLE suppliers  
(  
    supplier_id numeric(4),  
    supplier_name varchar2(50),  
    CONSTRAINT check_supplier_id  
    CHECK (supplier_id BETWEEN 100 and 9999)  
);
```

O check determina que o campo `supplier_id` tenha valores entre 100 e 9999.

Gabarito: Letra b

29. (FCC/ACE-TI/TCE-SE 2011) A linguagem de manipulação de dados (DML)

- a) sempre se utiliza de outra linguagem de programação hospedeira para que possa ser executada.**
- b) utiliza instruções INSERT, DROP, SELECT, UPDATE, entre outras.**
- c) pode ser executada mesmo que suas sentenças não estejam escritas em um código de linguagem de programação.**
- d) se utiliza de instruções que permitem a inclusão e alteração de dados, assim como, atribuição e negação de permissões.**
- e) é uma linguagem exclusivamente procedural, para especificar como os dados devem ser obtidos do banco.**

- a) Não, vamos ver mais a frente em uma questão mais específica que existe o SQL embutido e interativo, e a afirmativa diz que o comando SQL deve ser embutido em uma Linguagem de Programação. Quando abrimos o console de um SGBD, utilizamos SQL diretamente (interativamente), sem precisar embutir em uma Linguagem de Programação.
- b) Errado, pois Drop não é DML, é DDL.
- c) Isso, o SQL pode ser usado de forma interativa, diretamente no SGBD.
- d) Atribuição de negação de permissões é DCL, e não DML.
- e) É uma linguagem procedural, como já vimos, mas não pode ser considerada exclusivamente procedural, pois a DML é diferente de uma Linguagem de Programação, tendo muitas vezes a característica de uma linguagem imperativa. Por fim, não especifica apenas como os dados são obtidos, podendo ter outras utilidades (como comando para a exclusão de dados).

Gabarito: Letra c

30. (FCC/Analista de BD/INFRAERO 2011) Nas linguagens DML, DDL, DCL e DTL são usadas, respectivamente, as instruções

- a) Select, Drop, Start Transaction e Create Synonym.**
- b) Delete, Create, Commit e Revoke.**
- c) Create Synonym, Rollback, Select e Commit.**
- d) Select, Delete, Grant e Commit.**
- e) Update, Drop, Revoke e Rollback.**

Vamos lá, analisando o primeiro comando, quem se encaixa em DML: Select. Delete e Update. Assim, eliminamos a letra (c).

Vendo o segundo comando, quem é DDL: Drop e Create, a letra (d) está fora.

Agora vamos procurar DCL no terceiro comando das opções que sobraram: Somente o Revoke na letra (e), que é nossa resposta. Vendo o último comando, temos RollBack como DTL, confirmando assim a nossa resposta.

Gabarito: Letra e

31. (FCC/Analista de BD/INFRAERO 2011) ALTER PASSWORD e INSERT são, respectivamente, pertinentes à

- a) DCL e DDL.
- b) DCL e DML.
- c) DTL e DCL.
- d) DTL e DML.
- e) DDL e DTL.

Alter Password, utilizado para alterar uma senha, é um comando de DCL. Já Insert é DML como já sabemos.

Gabarito: Letra b

32. (FCC/Analista de Arquitetura de Software/INFRAERO 2011) Considere:

- I. Retorna linhas quando houver pelo menos uma correspondência entre duas tabelas.**
- II. Operador usado para combinar o resultado do conjunto de duas ou mais instruções SELECT.**
- III. Operador usado em uma cláusula WHERE para pesquisar um padrão específico em uma coluna.**

I, II e III correspondem em SQL, respectivamente, a

- a) SELECT, UNIQUE e BETWEEN.
- b) INNER JOIN, JOIN e DISTINCT.
- c) LEFT JOIN, UNIQUE e LIKE.
- d) SELECT, JOIN e BETWEEN.
- e) INNER JOIN, UNION e LIKE.

I. Retorna linhas quando houver pelo menos uma correspondência entre duas tabelas: O operador que faz isso é o Inner Join

II. Operador usado para combinar o resultado do conjunto de duas ou mais instruções SELECT: O operador Union faz a união de dois conjuntos, neste caso, dois ResultSets.

III. Operador usado em uma cláusula WHERE para pesquisar um padrão específico em uma coluna: Quem faz a pesquisa por um determinado

padrão é o like, como por exemplo: `Select * from Tabela where descrição LIKE 'ABC%'`

Gabarito: Letra e

33. (ISAE/Administrador BD/ALE-AM 2011) O Administrador de Bancos de Dados da Assembleia Legislativa do Estado do Amazonas deseja excluir um banco de dados denominado AUDITORIA_CONTROLE. Para isso, ele deve executar o seguinte comando SQL:

- a) **EXCLUDE database AUDITORIA_CONTROLE**
- b) **REMOVE database AUDITORIA_CONTROLE**
- c) **DELETE database AUDITORIA_CONTROLE**
- d) **ERASE database AUDITORIA_CONTROLE**
- e) **DROP database AUDITORIA_CONTROLE**

Indo direto ao que interessa, o comando Drop Database BancoDeDados, exclui determinado Banco de Dados.

Gabarito: Letra e

34. (ISAE/Administrador BD/ALE-AM 2011) Uma variação importante do comando SELECT é dada por uma cláusula SQL, que remove as entradas duplicadas do conjunto de resultados. Esta cláusula é mostrada na seguinte opção:

- a) **SELECT UNIQUE GERENTE IN CADASTRO_GERAL_ALEAM**
- b) **SELECT UNIQUE GERENTE ON CADASTRO_GERAL_ALEAM**
- c) **SELECT DISTINCT GERENTE IN CADASTRO_GERAL_ALEAM**
- d) **SELECT DISTINCT GERENTE ON CADASTRO_GERAL_ALEAM**
- e) **SELECT DISTINCT GERENTE FROM CADASTRO_GERAL_ALEAM**

O distinct é o operador que traz somente a informação distinta, retirando as repetições. Vejam que nas alternativas (a) até (d), nem a estrutura do Select está correta (falta o from)

Gabarito: Letra d

35. (CEPS-UFPA/Analista de TI/UFPA 2011). Em relação à linguagem SQL, é correto afirmar:

- a) **Pode ser usada através de dois modos: SQL Interativa e SQL Embutida.**
- b) **Divide-se em linguagem de definição de dados e linguagem de controle de dados.**
- c) **É composta somente pela Linguagem de Transação de Dados e Linguagem de Consulta de Dados.**

- d) A manipulação de dados é feita utilizando-se uma biblioteca à parte que é um adendo dessa linguagem.**
e) A linguagem de controle de dados tem como objetivo definir, alterar e eliminar as tabelas usadas para armazenar os dados.

A linguagem SQL interativa é a que conhecemos quando acessamos um console de um SGBD e digitamos diretamente nossos comandos SQL. Mas o SQL precisa muitas vezes ser embutido em uma linguagem de programação (linguagem host) para que seja utilizada em Sistemas de Informação. Os pedidos em SQL embutida são substituídos por declarações em linguagem host (C++, por exemplo), que chamam procedimentos disponíveis em tempo de execução para permitir o acesso ao Banco de Dados. Assim, a alternativa (a) está correta.

Gabarito: Letra a

36. (CEPS-UFPB/Analista de TI/UFPB 2011) O comando SQL utilizado para apagar os dados de uma tabela, fazendo com que esta retorne ao estado de origem, é

- a) ALTER TABLE**
- b) DROP TABLE**
- c) RESTART TABLE**
- d) TRUNCATE TABLE**
- e) DELETE ONLY DATA**

O comando Truncate Table NomeDaTabela é utilizado para limpar a tabela, apagando todos os seus registros. Seu uso é diferente de Delete * from NomeDaTabela, pois o truncate não gera Log, nem aciona triggers. Para limpar a tabela, o Sistema antes verifica se não há alguma restrição de integridade que impeça a deleção das tuplas.

Gabarito: Letra d

37. (CEPS-UFPB/Analista de TI/UFPB 2011) O comando SQL Select é utilizado para

- a) atualizar um valor de um atributo.**
- b) excluir uma linha de uma tabela.**
- c) calcular um valor de uma expressão aritmética.**
- d) incluir uma linha em uma tabela.**
- e) remover um atributo de uma tabela.**

- a) Não, quem faz isso é o Update.
- b) Quem exclui é o Delete

c) Sim, o comando Select pode ser usado para calcula o valor de uma expressão aritmética, sem consultar tabela nenhuma. Por exemplo, em MySql podemos usar:

```
Select 130 * 100 from dual
```

O dual é o que se chama de tabela dummy, utilizada quando nenhuma tabela do seu esquema for referenciada. No MySQL, pode ser usada a partir da versão 4.1.0. No SQL Server, utilizaríamos apenas:

```
Select 130 * 100
```

d) Quem inclui é o Insert

e) Para remover um atributo de uma tabela, ou usa Alter ou dropa a tabela e a recria

Gabarito: Letra c

38. (FCC/Analista Judiciário - TI/TRT19 2011) Considere as linguagens inseridas no contexto SQL: DML, DDL, DTL, DCL e DQL. Desta forma, Grant, Commit, Update, Delete e Alter, correspondem, respectivamente, a

- a) DCL, DCL, DDL, DDL e DDL.
- b) DTL, DCL, DML, DDL e DQL.
- c) DCL, DTL, DML, DML e DDL.
- d) DML, DTL, DCL, DQL e DML.
- e) DQL, DDL, DML, DML e DDL.

Grant: DCL

Commit: DTL

Update: DML

Delete: DML

Alter: DDL

Gabarito: Letra c

38. (FCC/Técnico Judiciário - TI/TRT19 2011) Uma tabela A contém apenas o código (cod_pac) de todos os pacientes internados em um hospital. Uma tabela B, com a mesma estrutura, contém o código (cod_pac) de todos os pacientes deste hospital que estão internados há mais de 15 dias. Para listar o código de todos os pacientes deste hospital internados por um período menor ou igual a 15 dias, deve-se utilizar a seguinte instrução SQL:

- a) **SELECT cod_pac FROM A DIFF SELECT cod_pac FROM B**
- b) **SELECT cod_pac FROM (A UNION B)**
- c) **SELECT cod_pac FROM A where cod_pac not in (SELECT cod_pac FROM B)**

- d) SELECT cod_pac FROM A where cod_pac in (A MINUS B)**
e) SELECT cod_pac FROM (A DIFF B)

Temos aí um Select com uma subquery. A questão pede os códigos de pacientes internados em período menor ou igual a 15 dias. Ora, os pacientes internados em período menor ou igual a 15 dias são aqueles armazenados na tabela A (que tem todos os pacientes internados, independente do prazo) e que não estão na tabela B (que tem pacientes internados a mais de 15 dias).

Para encontrar os pacientes que serão excluídos, já que estão internados a mais de 15 dias, utilizamos:

Select cod_pac from B (esta é a nossa subquery)

Agora, retornando os pacientes com 15 dias ou menos de internação, temos:

Select cod_pac from A where cod_pac not in (Select cod_pac from B).

A subquery (em azul) é executada primeiro.

Gabarito: Letra c

40. (FCC/Analista Judiciário - TI/TRT23 2011) As operações da álgebra relacional Seleção, Projeção e Produto Cartesiano são implementadas na linguagem SQL, respectivamente, pelas cláusulas

- a) Select, From e Where.**
- b) Select, Where e From.**
- c) Where, Select e From.**
- d) Where, From e Select.**
- e) Select, Select e Join.**

A operação de Seleção é implementada pelo Where.

A operação de Projeção é implementada pelo Select.

A operação de Produto Cartesiano é implementada quando se informa mais de uma tabela após o From.

Gabarito: Letra c

41. (FCC/Analista Judiciário - TI/TRT14 2011) A função SQL usada para extrair caracteres de um campo de texto é

- a) AVG().**
- b) MID().**
- c) FORMAT().**
- d) CASE().**
- e) LEN().**

A função MID é utilizada para extrair caracteres de um campo texto. Sua sintaxe é **MID(NomeDaColuna,inicio,[tamanho])**, sendo:

inicio: a posição inicial a ser retornada dentro do campo texto.

tamanho: O número de caracteres a serem retornados. É opcional, e se não for informado são retornados todos os caracteres a partir de inicio.

Gabarito: Letra b

42. (FCC/Analista Judiciário - TI/TRT24 2011) Um wildcard (curinga) SQL deve ser usado com um operador LIKE. Assim,

a) _ (símbolo underline) é um substituto para zero ou mais caracteres.

b) % (símbolo percentual) é um substituto para zero ou mais caracteres.

c) % (símbolo percentual) é um substituto para exatamente um caracter.

d) [!lista] indica qualquer caracter simples da lista.

e) [lista] indica nenhum caracter simples da lista.

O operador LIKE é utilizado para comparar strings com um determinado padrão. Assim, utilizamos o curinga % para substituir no padrão zero ou mais caracteres. Por exemplo, se quisermos procurar em determinada tabela todas as pessoas que tem MARIA no nome, utilizaríamos algo do tipo:

```
Select * from Tabela where nome like '%MARIA%'
```

Assim, todas as tuplas em que o campo Nome de Tabela tiver a sequência MARIA serão retornadas, como por exemplo: JOSÉ MARIA, MARIANA, ANAMARIA, e assim sucessivamente.

Observem que poderíamos usar somente um operador %, como por exemplo:

```
Select * from Tabela where nome like 'MARIA%' : retornaria todos os nomes que começam por MARIA.
```

```
Select * from Tabela where nome like '%MARIA': retornaria todos os nomes que terminam por MARIA.
```

Gabarito: Letra b

Ficamos por aqui, grande abraço e força na reta final!!!

3. LISTA DE QUESTÕES APRESENTADAS

1. (ESAF/Analista Sistemas/CVM 2010) Assinale a opção correta.

a) Programação de domínio é a associação de um domínio de valores possíveis a determinados atributos.

- b) Uma Linguagem de Manipulação de Dados (DML) principal requer que o usuário especifique que dados prioritários são necessários.**
- c) Restrição dimensional é a associação dimensional a certos tipos de atributos.**
- d) Uma Linguagem de Manipulação de Dados (DML) procedural requer que o usuário especifique que dados são necessários e como obtê-los.**
- e) Uma Linguagem de Recuperação de Dados (DRL) requer que o usuário tenha conhecimento de programação em linguagem de máquina.**

2. (ESAF/Analista Infraestrutura/CVM 2010) Na linguagem SQL

- a) char(n): uma string de caracteres de tamanho fixo n, especificado pelo usuário.**
- b) floatchar(n): uma string de caracteres de tamanho variável máximo n, especificado pelo usuário.**
- c) var(n): um número de ponto variável, com precisão de até n dígitos.**
- d) close(n): uma string de aproximação de caracteres numéricos de tamanho fixo n, definido pela linguagem.**
- e) doublefloat(n): um número de ponto flutuante duplo, com precisão modificada.**

3. (ESAF/Analista Infraestrutura/CVM 2010) Assinale a opção correta.

- a) A linguagem de definição de dados permite expressar as consultas e atualizações do banco de dados.**
- b) A linguagem de manipulação de dados permite a especificação do esquema do banco de dados.**
- c) A linguagem de manutenção de dados permite expressar as consultas e atualizações do banco de dados.**
- d) A linguagem de definição de dados permite a especificação do esquema do banco de dados.**
- e) A linguagem de manipulação de consultas permite a atualizações do banco de dados.**

4. (ESAF/APOTI/MPOG 2010) Em uma SQL

- a) a Linguagem de Manipulação de Relacionamentos compreende os comandos para construir tabelas em um banco de dados.**
- b) a Linguagem de Definição de Dados fornece tabelas para criação e modificação de comandos.**
- c) os comandos básicos da Linguagem de Definição de Dados são Select, Insert, Update e Delete.**
- d) a Linguagem de Manipulação de Dados compreende os comandos para inserir, remover e modificar informações em um**

banco de dados.

e) os comandos básicos da Linguagem de Definição de Dados são Sort, Insert, Undo e Store.

5. (ESAF/Analista Desen Sistemas/ANA 2009) Em SQL, a cláusula check aplicada a uma declaração de domínio

a) permite especificar um predicado que deve ser satisfeito por qualquer valor atribuído a uma variável de determinado domínio.

b) especifica um predicado que deve ser satisfeito por uma tupla em uma relação.

c) proíbe a inserção de um valor nulo para as variáveis do domínio.

d) verifica se os atributos considerados formam uma chave candidata.

e) não tem efeito, pois não se aplica esta cláusula a declarações de domínio.

6. (ESAF/PSS TI 2008) Na linguagem SQL, o uso da cláusula WHERE, quando apropriado,

a) agrupa o resultado em subconjuntos que possuem valores correspondentes em uma ou mais colunas. Em cada grupo não há duas linhas com o mesmo valor na(s) coluna(s) de agrupamento.

b) permite selecionar linhas baseadas em uma expressão booleana. Somente as linhas para as quais a expressão é avaliada como TRUE são retornadas no resultado.

c) especifica se o ResultSet de uma instrução SELECT simples, que atende aos requisitos para um cursor, é atualizável ou não.

d) permite selecionar colunas baseado em uma expressão numérica. Somente as colunas para as quais a expressão é avaliada como TRUE são retornadas no resultado. O uso desta cláusula obriga que os valores nulos sejam considerados diferentes nos agrupamentos.

e) cria uma ou mais linhas e as armazena na tabela especificada. O número de valores especificados na instrução WHERE deve ser idêntico ao número de colunas especificadas ou implícitas.

7. (ESAF/Auditor Informática/Pref Natal 2008) Quanto à estrutura, propriedades e sintaxe da linguagem SQL, é correto afirmar que

a) o CREATE e o DROP são comandos básicos da DDL (Data Definition Language - Linguagem de Definição de Dados).

b) o SELECT e o INSERT são comandos básicos da DDL (Data Definition Language - Linguagem de Definição de Dados).

c) o CREATE e o DROP são comandos básicos da DCL (Data Control Language - Linguagem de Controle de Dados).

d) o GRANT e o REVOKE são comandos básicos da DDL (Data

**Manipulation Language - Linguagem de Manipulação de Dados).
e) o SELECT e o INSERT são comandos básicos da DCL (Data Control Language - Linguagem de Controle de Dados).**

8. (ESAF/Auditor Informática/Pref Natal 2008) Com relação às características da linguagem SQL e dos SGBD, é correto afirmar que

- a) os "Triggers" são utilizados para auxiliar a manutenção da consistência dos dados. Também podem ser utilizados para propagar alterações ocorridas em um determinado dado de uma tabela para outra tabela.**
- b) considerando o SQL_ANSI, o código a seguir é um exemplo correto para a criação de um Trigger relacionado a ações na tabela "cadastro": CREATE TRIGGER trigger_cpf AS Codigo a ser executado FOR UPDATE ON cpf.cadastro**
- c) os "Triggers" são relações que não armazenam dados, mas são definidos dinamicamente por uma consulta a uma tabela previamente analisada e otimizada.**
- d) os "Triggers" são relações que armazenam dados, definidos dinamicamente por uma regra de inserção a uma tabela.**
- e) os "Triggers" são um arquivo auxiliar associado a uma ou mais tabelas. Sua função é acelerar o tempo de acesso aos dados de uma tabela.**

9. (ESAF/AFC/CGU 2008) Em um banco de dados que utiliza a linguagem SQL para definição, manipulação e controle de dados, é correto afirmar que os comandos

- a) CREATE, DROP e INSERT fazem parte da DML (Linguagem de Manipulação de Dados).**
- b) GRANT e REVOKE fazem parte da DCL (Linguagem de Controle de Dados).**
- c) INSERT, UPDATE e SELECT fazem parte da DDL (Linguagem de Definição de Dados).**
- d) ALTER, DELETE e REVOKE fazem parte da DML (Linguagem de Manipulação de Dados).**
- e) CREATE, GRANT e DROP fazem parte da DCL (Linguagem de Controle de Dados).**

10. (ESAF/AFC/CGU 2008) Uma subquery (subconsulta) é um comando SELECT que foi "embutido" em outro comando SELECT, UPDATE, DELETE ou dentro de outra subquery. A finalidade da subquery é retornar um conjunto de linhas para a query (consulta) ou comando principal. Com relação às subqueries, é correto afirmar que

- a) uma subquery não precisa estar incluída entre parêntesis.**
- b) uma subquery sempre deve estar do lado esquerdo do**

operador de comparação.

c) uma subquery pode conter a cláusula ORDER BY.

d) o operador IN não pode ser utilizado em uma subquery que retorne múltiplas linhas.

e) o operador igual "=" não pode ser utilizado em uma consulta que contenha uma subquery que retorne múltiplas linhas.

11. (ESAF/AFC/CGU 2008) Considere as seguintes consultas envolvendo dados de duas tabelas (T1 e T2) que possuem o atributo CODIGO em comum e assinale a opção correta.

Consulta 1

SELECT * FROM T1 WHERE CODIGO IN (SELECT CODIGO FROM T2)

Consulta 2

SELECT * FROM T1 WHERE NOT EXISTS (SELECT CODIGO FROM T2 WHERE T1.CODIGO = T2.CODIGO)

Consulta 3

SELECT T1.* , T2.* FROM T1 INNER JOIN T2 ON T1.CODIGO = T2.CODIGO

Consulta 4

SELECT * FROM T1 WHERE CODIGO NOT IN (SELECT CODIGO FROM T2)

Consulta 5

SELECT T1.* FROM T1 LEFT OUTER JOIN T2 ON T1.CODIGO = T2.CODIGO WHERE T2.CODIGO IS NULL

Consulta 6

SELECT T1.* , T2.* FROM T1 , T2 WHERE T1.CODIGO = T2.CODIGO

a) O resultado da consulta 3 é igual ao resultado da consulta 4.

b) O resultado da consulta 2 é igual ao resultado da consulta 6.

c) O resultado da consulta 2 é igual ao resultado da consulta 5.

d) O resultado da consulta 5 é igual ao resultado da consulta 6.

e) O resultado da consulta 1 é igual ao resultado da consulta 5

12. (ESAF/Analista de TI/SEFAZ-CE 2007) Na linguagem SQL, os procedimentos gravados para serem executados implicitamente quando ocorrer determinada ação do usuário, como, por exemplo, a modificação de uma tabela, são denominados

a) Selects.

b) Inserts.

c) Views.

d) Queries.

e) Triggers.

13. (ESAF/Analista de TI/SEFAZ-CE 2007) No SQL, considerando o uso das cláusulas SELECT, GROUP BY e HAVING, indique a opção que apresenta uma sintaxe correta.

a) **SELECT Ano, COUNT(*) AS 'Total' FROM Matricula WHERE Cod =**

171 GROUP BY Ano HAVING count(*) > 5
b) SELECT Ano, COUNT(*) AS 'Total' FROM Matricula HAVING count(*) > 5 WHERE Cod = 171 GROUP BY Ano
c) SELECT Ano, COUNT(*) AS 'Total' WHERE Cod = 171 FROM Matricula GROUP BY Ano HAVING count(*) > 5
d) SELECT Ano, COUNT(*) AS 'Total' WHERE Cod = 171 GROUP BY Ano HAVING count(*) > 5 FROM Matricula
e) SELECT Ano WHERE Cod = 171 GROUP BY Ano HAVING count(*) > 5 COUNT(*) AS 'Total' FROM Matricula

14. (ESAF/Analista de Sistemas/ENAP 2006) Analise as seguintes afirmações relacionadas a Gerenciamento de Banco de Dados.

- I. A *Data Definition Language* (DDL) é composta, entre outros, pelo comando *Create*, que é utilizado para criar bancos de dados, tabelas e relações existentes entre as tabelas.**
II. Os comandos da *Data Manipulation Language* (DML) são destinados a consultas, inserções, exclusões e alterações em um ou mais registros de uma ou mais tabelas de maneira simultânea.
III. Os comandos *Truncate* e *Delete* são exemplos de comandos da *Data Definition Language* (DDL).
IV. Os comandos *Alter*, *Drop* e *Rename* são exemplos de comandos da subclasse de comandos da *Data Manipulation Language* (DML) chamada DCL (*Data Control Language*).
Indique a opção que contenha todas as afirmações verdadeiras.
a) II e III
b) I e II
c) III e IV
d) I e III
e) II e IV

15. (ESAF/Analista de Sistemas/ENAP 2006) Quanto ao uso de diagramas da linguagem SQL é correto afirmar que os comandos *Grant* e *Revoke*

- a) são usados para alterar estruturas de tabelas e de seus atributos.**
b) possibilitam incluir o nome dos atributos e das tabelas.
c) não permitem retrocessos e por isso são classificados como *Data Definition Language* (DDL).
d) não permitem retrocessos e por isso são classificados como *Data Manipulation Language* (DML).
e) são responsáveis por dar ou retirar permissões para os usuários.

16. (ESAF/Analista de Sistemas/ENAP 2006) A SQL possui recursos para apagar tabelas e bancos de dados a partir do

comando

- a) CREATE TABLE NULL.**
- b) TRUNCATE PRIMARY KEY.**
- c) DROP.**
- d) INSERT INTO TABLE NULL.**
- e) DELETE TABLE NULL.**

17. (ESAF/Analista Técnico/SUSEP 2006) Analise as seguintes afirmações relacionadas a conceitos básicos sobre Banco de Dados.

- I. O comando SQL responsável por fechar uma transação confirmando as operações feitas é o INSERT.**
- II. O comando SQL responsável por fechar uma transação e desfazer todas as operações é o COMMIT.**
- III. Quando uma transação ainda está aberta para um usuário, enquanto não é executado um comando COMMIT, o próprio usuário pode ver as suas alterações, mas outros usuários não podem vê-las.**
- IV. Uma transação assegura um espaço de trabalho que contém várias alterações, inclusões e exclusões de dados em uma ou mais tabelas, com a possibilidade de confirmação ou cancelamento das operações sem comprometimento dos dados.**

Indique a opção que contenha todas as afirmações verdadeiras.

- a) I e II**
- b) II e III**
- c) III e IV**
- d) I e III**
- e) II e IV**

18. (ESAF/Analista Informática/IRB 2006) Considere um banco de dados com uma tabela com o nome "Carro", onde estão cadastrados os mais de 1000 veículos de uma empresa. Esta tabela contém um campo de nome "Cor", que pode ser nulo, contendo a informação da cor de cada veículo cadastrado. Ao se executar a instrução SQL ALTER TABLE Carro DROP COLUMN Cor

- a) a tabela Carro será totalmente apagada devido à execução do comando DROP.**
- b) todos os registros preenchidos com o valor Null serão alterados.**
- c) todos os registros preenchidos com o valor diferente de Null serão alterados.**
- d) todas as informações das cores dos veículos contidas neste campo serão perdidas.**
- e) nada irá acontecer com os dados da tabela Carro.**

19. (ESAF/Analista Informática/IRB 2006) Analise as seguintes afirmações relacionadas a banco de dados e ao uso da linguagem SQL.

I. A cláusula DISTINCT é usada com o SELECT nos casos em que se deseja eliminar as tuplas repetidas deixando apenas uma tupla de cada valor no resultado da consulta.

II. As funções UPPER, LOWER e REPLACE são exemplos de funções para manipulação de chaves primárias.

III. Considere que em um banco de dados uma tabela chamada "Carro" contenha os campos CodCarro, Descricao e Cor.

Condiderando, ainda, que os campos Descricao e Cor podem receber o valor Null, a execução do comando SQL SELECT CodCarro, Descricao, NVL(Cor,'Branca') FROM Carro WHERE Cor is NULL caso retorne registros, todos os valores do campo Cor destes registros estarão preenchidos com o valor Branca.

IV. A função COUNT faz a soma dos valores das linhas retornadas em uma consulta.

Indique a opção que contenha todas as afirmações verdadeiras.

- a) I e II
- b) II e III
- c) III e IV
- d) I e III
- e) II e IV

20. (ESAF/Analista Informática/IRB 2006) Analise as seguintes afirmações relacionadas a banco de dados e ao uso da linguagem SQL.

I. A cláusula ORDER BY permite criar consultas com as linhas em uma ordem específica, modificando a ordem de apresentação do resultado da pesquisa. As linhas podem ser ordenadas de forma ascendente ou descendente de acordo com as colunas declaradas na cláusula ORDER BY.

II. Os dois comandos SELECT apresentados a seguir são equivalentes.

**SELECT *
FROM Carro
WHERE CodCarro <> ANY (SELECT CodCarro
FROM Vendido).**

**SELECT *
FROM Carro WHERE CodCarro NOT IN (SELECT
CodCarro FROM Vendido).**

III. O comando SQL abaixo retornará a relação dos carros cujo código CodCarro seja igual a 51.

**SELECT Ano, COUNT(*) AS Total de Carros
FROM Veiculos
GROUP BY Ano**

HAVING CodCarro = 51

IV. A cláusula GROUP BY pode ser colocada antes ou depois da cláusula HAVING. Se ela for colocada antes, os grupos são formados e as funções de grupos são calculadas antes de resolver a cláusula HAVING. Se for colocada depois, a cláusula HAVING filtra a seleção antes da formação dos grupos e do cálculo das funções.

Indique a opção que contenha todas as afirmações verdadeiras.

- a) I e II**
- b) II e III**
- c) III e IV**
- d) I e III**
- e) II e IV**

21. (ESAF/AFC/CGU 2006) Um procedimento armazenado (stored procedure) é uma coleção de comandos em SQL que

a) provoca um aumento no tráfego na rede e reduz a performance do sistema, mas continua sendo largamente utilizado para criar mecanismos de segurança em bancos de dados relacionais.

b) encapsula tarefas repetitivas, aceita parâmetros de entrada e pode retornar um valor de status para indicar sucesso ou falha na execução.

c) estão armazenados no banco de dados e que são executadas diretamente na máquina do usuário.

d) estão armazenados na máquina do usuário e que são executadas diretamente no servidor do banco de dados.

e) são utilizados unicamente para autenticar um usuário, dando a ele direitos de acesso a escrita/alteração em tabelas do banco de dados.

22. (ESAF/AFC/CGU 2006) Analise as seguintes afirmações relacionadas a conceitos básicos de banco de dados e linguagem SQL.

I. Na linguagem SQL um INNER JOIN retorna todas as tuplas comuns às duas tabelas.

II. Em uma Junção entre duas tabelas a cláusula USING só poderá ser usada quando o nome do atributo for igual nas duas tabelas.

III. Na linguagem SQL um RIGHT OUTER JOIN retorna todas as tuplas que não são comuns às duas tabelas.

IV. Uma Junção é usada para compor informações complexas a partir de tabelas sem nenhum tipo de relacionamento.

Indique a opção que contenha todas as afirmações verdadeiras.

- a) I e III**

- b) II e III
- c) III e IV
- d) I e II
- e) II e IV

23. (ESAF/TRF/SRF 2006) Analise as seguintes afirmações relacionadas a Bancos de Dados e à linguagem SQL:

I. A cláusula *GROUP BY* do comando *SELECT* é utilizada para dividir colunas em conjuntos maiores de dados relacionados entre si.

II. Uma *view* é uma forma predeterminada de visualizar dados de uma ou mais tabelas como se fosse apenas uma tabela.

III. Quando o comando *DROP TABLE* é executado, as estruturas da tabela e os dados são excluídos. Porém, quando o *DROP VIEW* é executado, nenhum dado referenciado pela *view* é afetado.

IV. O *trigger* é um tipo de *view* criado quando um evento em particular ocorre. Indique a opção que contenha todas as afirmações verdadeiras.

- a) I e II
- b) III e IV
- c) II e III
- d) I e III
- e) II e IV

24. (ESAF/TRF/SRF 2006) Em um Banco de Dados Relacional, utilizando-se a linguagem SQL para a manipulação de dados, é correto afirmar que

a) para se efetuar a *JUNÇÃO* entre tabelas é essencial que não exista relacionamento entre elas.

b) a DML (*Data Manipulation Language*) é responsável pela manipulação de estruturas de dados, sendo composta por comandos destinados à criação, alteração e exclusão de bancos de dados e tabelas. Os comandos *Create*, *Alter*, *Drop* e *Rename* são exemplos de comandos da classe DML.

c) na *JUNÇÃO* entre tabelas, a chave estrangeira da entidade "1" no relacionamento "1 : N" deve ser exportada como chave primária da entidade "N".

d) uma transação é uma unidade lógica que contém um conjunto de operações que formam uma unidade de processamento. O comando SQL responsável por fechar uma transação, confirmando as operações feitas é o *COMMIT*. Para desfazer todas as operações, o comando é *ROLLBACK*.

e) na *JUNÇÃO* entre tabelas, a cláusula *ON* só poderá ser usada quando o nome do atributo for igual nas duas tabelas. Quando os nomes dos atributos tiverem valores diferentes, a cláusula a ser usada será a *USING*.

25. (ESAF/AFRF/SRF 2005) SQL é uma linguagem de consulta, criada para extrair, organizar e atualizar informações em bancos de dados relacionais. Com relação aos conceitos básicos de SQL, é correto afirmar que a instrução COMMIT

- a) realiza todas as alterações feitas até o ponto em questão.**
- b) altera a definição de uma tabela ou de outro objeto de banco de dados.**
- c) elimina uma tabela ou outro objeto de um banco de dados.**
- d) desfaz todas as alterações feitas até o ponto em questão.**
- e) define um cursor, que pode ser utilizado para processar linhas acessadas por uma consulta. A instrução COMMIT inclui uma instrução SELECT, que é utilizada para processar a consulta.**

26. (ESAF/AFRF/SRF 2005) Com relação ao uso da SQL na manipulação de dados, caso se queira eliminar linhas repetidas do conjunto resultado, deve-se utilizar a palavra-chave DISTINCT, da seguinte forma:

- a) SELECT {colunas} FROM {tabelas} DISTINCT.**
- b) DISTINCT SELECT {colunas} FROM {tabelas}.**
- c) SELECT FROM {tabelas} DISTINCT {colunas}.**
- d) SELECT DISTINCT {colunas} FROM {tabelas}.**
- e) FROM {tabelas} SELECT DISTINCT {colunas}.**

27. (FCC/Analista de Sistemas/COPERGAS 2011) São respectivos a DML, DDL, DCL e DTL:

- a) Update, Drop, Grant e Commit.**
- b) Select, Revoke, Delete e Rollback.**
- c) Delete, Alter, Insert e Drop.**
- d) Insert, Commit, Revoke e Select.**
- e) Alter, Update, Revoke e Rollback.**

28. (FCC/ACE-TI/TCE-SE 2011) Durante a criação de uma tabela - Create Table, em SQL, deseja-se especificar que uma coluna só possa incluir, por exemplo, valores maiores que zero. Uma constraint utilizada para isso é

- a) Verify.**
- b) Check.**
- c) Max.**
- d) Avg.**
- e) Having.**

29. (FCC/ACE-TI/TCE-SE 2011) A linguagem de manipulação de dados (DML)

- a) sempre se utiliza de outra linguagem de programação hospedeira para que possa ser executada.**

- b) utiliza instruções **INSERT, DROP, SELECT, UPDATE**, entre outras.
- c) pode ser executada mesmo que suas sentenças não estejam escritas em um código de linguagem de programação.
- d) se utiliza de instruções que permitem a inclusão e alteração de dados, assim como, atribuição e negação de permissões.
- e) é uma linguagem exclusivamente procedural, para especificar como os dados devem ser obtidos do banco.

30. (FCC/Analista de BD/INFRAERO 2011) Nas linguagens DML, DDL, DCL e DTL são usadas, respectivamente, as instruções

- a) **Select, Drop, Start Transaction e Create Synonym.**
- b) **Delete, Create, Commit e Revoke.**
- c) **Create Synonym, Rollback, Select e Commit.**
- d) **Select, Delete, Grant e Commit.**
- e) **Update, Drop, Revoke e Rollback.**

31. (FCC/Analista de BD/INFRAERO 2011) ALTER PASSWORD e INSERT são, respectivamente, pertinentes à

- a) **DCL e DDL.**
- b) **DCL e DML.**
- c) **DTL e DCL.**
- d) **DTL e DML.**
- e) **DDL e DTL.**

32. (FCC/Analista de Arquitetura de Software/INFRAERO 2011) Considere:

- I. Retorna linhas quando houver pelo menos uma correspondência entre duas tabelas.**
- II. Operador usado para combinar o resultado do conjunto de duas ou mais instruções SELECT.**
- III. Operador usado em uma cláusula WHERE para pesquisar um padrão específico em uma coluna.**

I, II e III correspondem em SQL, respectivamente, a

- a) **SELECT, UNIQUE e BETWEEN.**
- b) **INNER JOIN, JOIN e DISTINCT.**
- c) **LEFT JOIN, UNIQUE e LIKE.**
- d) **SELECT, JOIN e BETWEEN.**
- e) **INNER JOIN, UNION e LIKE.**

33. (ISAE/Administrador BD/ALE-AM 2011) O Administrador de Bancos de Dados da Assembleia Legislativa do Estado do Amazonas deseja excluir um banco de dados denominado AUDITORIA_CONTROLE. Para isso, ele deve executar o seguinte comando SQL:

- a) EXCLUDE database AUDITORIA_CONTROLE
- b) REMOVE database AUDITORIA_CONTROLE
- c) DELETE database AUDITORIA_CONTROLE
- d) ERASE database AUDITORIA_CONTROLE
- e) DROP database AUDITORIA_CONTROLE

34. (ISAE/Administrador BD/ALE-AM 2011) Uma variação importante do comando SELECT é dada por uma cláusula SQL, que remove as entradas duplicadas do conjunto de resultados. Esta cláusula é mostrada na seguinte opção:

- a) SELECT UNIQUE GERENTE IN CADASTRO_GERAL_ALEAM
- b) SELECT UNIQUE GERENTE ON CADASTRO_GERAL_ALEAM
- c) SELECT DISTINCT GERENTE IN CADASTRO_GERAL_ALEAM
- d) SELECT DISTINCT GERENTE ON CADASTRO_GERAL_ALEAM
- e) SELECT DISTINCT GERENTE FROM CADASTRO_GERAL_ALEAM

35. (CEPS-UFGA/Analista de TI/UFGA 2011). Em relação à linguagem SQL, é correto afirmar:

- a) Pode ser usada através de dois modos: SQL Interativa e SQL Embutida.
- b) Divide-se em linguagem de definição de dados e linguagem de controle de dados.
- c) É composta somente pela Linguagem de Transação de Dados e Linguagem de Consulta de Dados.
- d) A manipulação de dados é feita utilizando-se uma biblioteca à parte que é um adendo dessa linguagem.
- e) A linguagem de controle de dados tem como objetivo definir, alterar e eliminar as tabelas usadas para armazenar os dados.

36. (CEPS-UFGA/Analista de TI/UFGA 2011) O comando SQL utilizado para apagar os dados de uma tabela, fazendo com que esta retorne ao estado de origem, é

- a) ALTER TABLE
- b) DROP TABLE
- c) RESTART TABLE
- d) TRUNCATE TABLE
- e) DELETE ONLY DATA

37. (CEPS-UFGA/Analista de TI/UFGA 2011) O comando SQL Select é utilizado para

- a) atualizar um valor de um atributo.
- b) excluir uma linha de uma tabela.
- c) calcular um valor de uma expressão aritmética.
- d) incluir uma linha em uma tabela.
- e) remover um atributo de uma tabela.

38. (FCC/Analista Judiciário - TI/TRT19 2011) Considere as linguagens inseridas no contexto SQL: DML, DDL, DTL, DCL e DQL. Desta forma, Grant, Commit, Update, Delete e Alter, correspondem, respectivamente, a

- a) DCL, DCL, DDL, DDL e DDL.
- b) DTL, DCL, DML, DDL e DQL.
- c) DCL, DTL, DML, DML e DDL.
- d) DML, DTL, DCL, DQL e DML.
- e) DQL, DDL, DML, DML e DDL.

38. (FCC/Técnico Judiciário - TI/TRT19 2011) Uma tabela A contém apenas o código (cod_pac) de todos os pacientes internados em um hospital. Uma tabela B, com a mesma estrutura, contém o código (cod_pac) de todos os pacientes deste hospital que estão internados há mais de 15 dias. Para listar o código de todos os pacientes deste hospital internados por um período menor ou igual a 15 dias, deve-se utilizar a seguinte instrução SQL:

- a) **SELECT cod_pac FROM A DIFF SELECT cod_pac FROM B**
- b) **SELECT cod_pac FROM (A UNION B)**
- c) **SELECT cod_pac FROM A where cod_pac not in (SELECT cod_pac FROM B)**
- d) **SELECT cod_pac FROM A where cod_pac in (A MINUS B)**
- e) **SELECT cod_pac FROM (A DIFF B)**

40. (FCC/Analista Judiciário - TI/TRT23 2011) As operações da álgebra relacional Seleção, Projeção e Produto Cartesiano são implementadas na linguagem SQL, respectivamente, pelas cláusulas

- a) **Select, From e Where.**
- b) **Select, Where e From.**
- c) **Where, Select e From.**
- d) **Where, From e Select.**
- e) **Select, Select e Join.**

41. (FCC/Analista Judiciário - TI/TRT14 2011) A função SQL usada para extrair caracteres de um campo de texto é

- a) **AVG().**
- b) **MID().**
- c) **FORMAT().**
- d) **CASE().**
- e) **LEN().**

- 42. (FCC/Analista Judiciário - TI/TRT24 2011) Um wildcard (curinga) SQL deve ser usado com um operador LIKE. Assim,**
- a) _ (símbolo underline) é um substituto para zero ou mais caracteres.**
 - b) % (símbolo percentual) é um substituto para zero ou mais caracteres.**
 - c) % (símbolo percentual) é um substituto para exatamente um caracter.**
 - d) [!lista] indica qualquer caracter simples da lista.**
 - e) [lista] indica nenhum caracter simples da lista.**

4. GABARITO

1	d	2	a	3	d	4	d	5	a
6	b	7	a	8	a	9	b	10	c
11	c	12	e	13	a	14	b	15	e
16	c	17	c	18	d	19	d	20	a
21	b	22	d	23	c	24	d	25	a
26	d	27	a	28	b	29	c	30	e
31	b	32	e	33	e	34	d	35	a
36	d	37	c	38	c	39	c	40	c
41	b	42	b						