

Exercícios

Disciplina: Banco de Dados II

Período: 3º

Professor(a): Gilberto Assis

Aluno(a): _____ Nota: _____

1 - Em um sistema gerenciador de bancos de dados, normalmente é possível criar gatilhos (triggers), cujo disparo pode ocorrer em decorrência

- a) execução de um comando de consulta, por meio do SQL, com advento da cláusula GROUP BY.
- b) de alguma operação de exclusão, modificação ou inserção de um registro em uma tabela do banco de dados.
- c) da queda de energia do servidor do banco de dados por um período maior do que uma hora.
- d) da substituição de um disco em uma arquitetura RAID de armazenamento de dados.
- e) da execução de um comando de consulta, por meio do SQL, sem a cláusula where.

2 - Sobre Triggers em SQL, analise as assertivas e assinale a alternativa que aponta a(s) correta(s).

I. Em alguns casos, é conveniente especificar um tipo de ação a ser tomada quando certos eventos ocorrem e quando certas condições são satisfeitas em um banco de dados. Podemos criar esses mecanismos através do comando CREATE TRIGGERS.

II. Triggers permitem a junção entre n tabelas em um banco.

III. Triggers são especialistas na conversão e adaptação de comandos SQL de um SGBD para outro.

IV. Um típico Triggers possui três componentes: evento(s), condição e ação.

- a) Apenas I.
- b) Apenas I e IV.
- c) Apenas I, III e IV.
- d) Apenas II, III e IV
- e) I, II, III e IV.

3 - Suponha um banco de dados com uma tabela FUNCIONARIO contendo os campos cod_func (chave primária), nome, salario. Para impedir que o valor do campo salario em uma linha sofra modificação, um trigger pode ser criado com o seguinte script SQL (ANSI SQL99).

```
a) create trigger TR_BLOQUEIOSALARIO.FUNCIONARIO
for update
as
if update (salario)
begin
funcionario.salario=updated.oldValue
stop
end;
/
```

```
b) create trigger TR_BLOQUEIOSALARIO
on FUNCIONARIO
for update
as
if update (salario)
begin
rollback transaction
end;
/
```

```
c) create trigger TR_BLOQUEIOSALARIO
on FUNCIONARIO.salario
for update (cod_func)
begin
rollback transaction
end;
/
```

```
d) create trigger TR_BLOQUEIOSALARIO.FUNCIONARIO.salario
for update (cod_func)
as
begin
commit transaction
end;
/
```

```
e) create trigger TR_BLOQUEIOSALARIO
on FUNCIONARIO
for update
begin
funcionario.salario=updated.oldValue
stop
end;
/
```

4 - Considere a tabela EMP em um banco de dados Oracle, contendo os campos ID, NOME, CARGO e SALARIO. Considere ainda que a procedure seguinte tem a intenção de reajustar o salário de todos os funcionários em 20%:

```
CREATE OR REPLACE PROCEDURE aumento  
BEGIN  
UPDATE EMP SET SALARIO = SALARIO * 1.2; END;
```

Para que o procedure seja considerado correto é necessário

- a) retirar a instrução OR REPLACE, pois de outra forma, ocorrerá um erro de sintaxe.
- b) trocar a palavra-chave SET por VALUE.
- c) trocar a palavra-chave BEGIN por { e END por }.
- d) acrescentar a palavra-chave IS imediatamente antes da instrução BEGIN.
- e) acrescentar a palavra-chave START imediatamente após a instrução BEGIN.

5 - Na linguagem PL/SQL, deseja-se desabilitar um trigger denominado Abc, mas não excluí-lo do banco de dados. O comando que realiza esta tarefa é

- a) DROP TRIGGER Abc = 0;
- b) CHANGE TRIGGER Abc FOR NULL;
- c) MAKE TRIGGER Abc DOWN;
- d) PUT TRIGGER Abc OFF;
- e) ALTER TRIGGER Abc DISABLE;

7 - Considere o bloco PL/SQL abaixo:

```
CREATE OR REPLACE TRIGGER TG_DEPARTAMENTO  
BEFORE INSERT OR UPDATE ON DEPARTAMENTO  
FOR EACH ROW  
DECLARE  
    FLAG NUMBER;  
BEGIN  
    :NEW.NOME_DEPARTAMENTO := UPPER(:NEW.NOME_DEPARTAMENTO);  
END;  
/
```

é INCORRETO afirmar que este bloco contém comandos para

- a) criar o trigger department_maiusc.
- b) substituir o trigger existente de nome department_maiusc.
- c) disparar o trigger sempre que houver a inclusão de uma nova linha na respectiva tabela do banco de dados.
- d) acionar o trigger sempre que houver mudança num registro da respectiva tabela do banco de dados.
- e) forçar o nome do departamento a ser colocado em letras minúsculas.