

Universidade do Estado de Santa Catarina - UDESC

Curso: **Sistemas de Informação**

Disciplina: **Banco de Dados II**

Professor: Osmar de Oliveira Braz Júnior

Nome: _____ Data: ____/____/____

Exercícios de DML - TRIGGERS

1. Crie uma TRIGGER para preencher o conteúdo de uma coluna caso ele seja deixado em branco durante uma operação de INSERT. Nesse exercício caso seja deixado em branco o campo data de cadastro durante a inclusão, a TRIGGER automaticamente preenche com a data do sistema. Se for necessário altere a tabela piloto adicionando o campo DATA_INSERT DATE.

Piloto

codigo_piloto	Int	Chave Primária	
nome_piloto	Varchar(100)		Obrigatório
salario	Numeric(9,2)		
gratificacao	Numeric(9,2)		
companhia	Varchar(30)		
pais	Varchar(15)		
data_insert	Date		
usuario_insert	Varchar(30)		

Verifique se o campo data_insert é Nulo (IS NULL). Pois a aplicação pode ter preenchido o campo.

Utilize o

```
IF (condição) THEN
    {comando}
END IF;
```

2. Altere a TRIGGER anterior para que armazene também na tupla o nome do usuário que inseriu o registro. Você pega o nome do usuário através da variável USER. É necessário alterar a tabela piloto adicionando o campo USUARIO_INSERT VARCHAR(30).
3. Adicione os campos DATA_UPDATE do tipo DATE e USUARIO_UPDATE do tipo VARCHAR(30) para a tabela piloto. Em seguida crie uma TRIGGER que armazene a data e o nome do usuário que realizou a ultima atualização do registro.
4. Junte as duas TRIGGER anterior em uma só utilizando predicados condicionais (INSERTING,UPDATING).
- 4.5 Crie uma TRIGGER para semelhante as anteriores para armazenar a data e o usuário que excluiu o registro.
5. A TRIGGER anterior esta auditando a tabela aonde foi criada somente para as operações de inserção e atualização. Mas para que também consiga auditar a operação de exclusão é necessário que uma tabela seja criada e a TRIGGER anterior seja excluída. Uma nova TRIGGER utilizando predicados condicionais.(INSERTING,UPDATING e DELETING) deve ser criada desta vez com o DELETING para inserir na tabela de log registros destas operações. A tabela tem os seguintes campos:

LOG

Log_id	Int	Chave Primária	Utilize uma seqüência para gerar o id
Data	Date		Data de inserção do log, utilize SYSDATE
Usuario	Varchar(30)		Nome do usuário utilize USER
Tabela	Varchar(30)		Tabela modificada
Operacao	Char(1)		Operação realizada('I','U','D')
Chave	int		Campo chave modificado

6. Vamos estabelecer um valor máximo para o campo salário. Nenhum Piloto poderá ganhar mais do que 10.000 dólares. Além de ativar a TRIGGER na inclusão, evitando que um novo funcionário tenha um salário superior ao limite, as rotinas de atualização(UPDATE) devem ser verificadas para evitar que os Pilotos existentes sejam alterados para valores não permitidos.

Utilize a operação abaixo para exibir a mensagem de erro e interromper a inserção ou atualização.

```
raise_application_error(-20000, 'Error Checking sal/ Salario acima do limite');
```

7. Vamos criar uma TRIGGER que não permita atualização(insert, update e delete) da tabela de pilotos entre os dias 10 e 20 do mês.
8. Neste exercício todo registro inserido na tabela de Piloto será inserido também na tabela Piloto resumido. Inclua uma TRIGGER para realizar esta tarefa.

Piloto Resumido

Codigo_Piloto	Int	Obrigatório	Chave Primária
Nome_Piloto	Varchar(100)		

9. Para que a tabela de Piloto resumido fique sempre com a mesma quantidade de registros uma outra TRIGGER deve ser criada para excluir registros.
10. Para que a tabela de Piloto resumido fique atualizada em relação a tabela de Piloto uma outra TRIGGER deve ser criada para atualizar os registros.
11. Implemente uma única TRIGGER que faça a replicação da tabela piloto para piloto resumido.