

Prática 2 – SQL - DDL e DML

Entrega: 22/Agosto (Tidia → Atividades)

- Deve ser entregue um relatório, em formato .doc (**somente .doc**) de acordo com o documento modelo “Relatorio-Modelo.doc” encontrado no repositório do Tidia.

Explicações, justificativas e comentários sobre a resolução do exercício.

Código SQL a ser compilado/executado (incluir **todos** os comandos executados). Na correção, será feito Copy/Paste do código para compilação. **Se o código não compilar, o exercício recebe nota zero até que o grupo, na revisão da correção, identifique o erro de compilação e indique como corrigi-lo.**

Saída: resultados das consultas ou de procedimentos, funções, triggers, etc...

- **Apenas um** membro da dupla deve fazer o *upload* do arquivo.
- O arquivo deve ser identificado da seguinte maneira:
PraticaX_Nome1Sobrenome1_Nome2Sobrenome2

⇒ **CERTIFIQUE-SE DE QUE O RELATÓRIO FOI RECEBIDO PELO SISTEMA – POSTERIORMENTE NÃO SERÁ ACEITO O ARGUMENTO DE QUE HOUVE FALHA NO SISTEMA DE SUBMISSÃO.**

1. Usando o **SQL Developer** (não incluir este exercício no relatório)
 - a. conecte-se com o seu usuário (veja padrão apresentado na aula Oracle);
 - b. altere sua senha (veja comando na aula Oracle)
 - c. desconecte-se;
 - d. conecte-se novamente com a nova senha.
2. Crie o esquema da base de dados usando o projeto lógico do Sistema de Eleições desenvolvido na Prática 1.
 - Nesta prática, se a dupla tiver em mãos 2 projetos lógicos (1 de cada aluno, vindos de duplas diferentes na Prática 1), deve-se escolher o projeto mais correto e usar nesta prática.
 - Se, durante a prática, forem identificados erros no projeto lógico, a dupla deve fazer as correções no projeto usado, e desenvolver a Prática 2 de acordo com as correções feitas.
 - a. (0.5) crie sequências e use-as para atributos numéricos sequenciais de seu projeto lógico – ex: Número de zona eleitoral.
 - b. (5.0) tabelas – crie todas as tabelas do projeto lógico, incluindo nas tabelas:
 - i. restrições de integridade – defina as chaves primárias e secundárias, e as chaves estrangeiras adequadas para o esquema relacional, além das ações adequadas para remoção de tuplas relacionadas;
 - ii. restrições de valor nulo – as restrições devem atender aos requisitos colocados no DER, no projeto lógico e na descrição do sistema (cardinalidade, restrições de especialização, participação total, etc....)
 - iii. valores *default* – usar valores apropriados de acordo com a semântica da aplicação

- iv. restrições de verificação de validade dos dados (ex: intervalo válido ou conjunto de valores possíveis...)

3. Usando o esquema relacional criado no exercício 2:

- a. (2.0) Insira pelos menos 2 tuplas em cada tabela da base de dados. Requisitos:
 - As seqüências criadas devem ser usadas na inserção dos dados.
 - Insira valores *default* e valores *null*.
 - Nos atributos de data e hora devem ser usadas as funções de manipulação e formatação de data e hora dos tipos de dados específicos para data e hora (ex.: DATE, TIMESTAMP).
 - Insira valores que tenham significado semântico, e não coisa do tipo “bla, bla, bla”. Os dados não precisam ser reais, mas precisam ter sentido semântico. Isso será levado em consideração na correção.
 - Todas as inserções devem ser feitas via comando **insert**, obrigatoriamente.
- b. (0.5) Faça as seguintes alterações:
 - Escolha uma tabela e faça uma alteração nos valores de 2 ou mais atributos de um conjunto de tuplas. Use uma condição de localização (um predicado) para encontrar as tuplas a serem modificadas.
 - Escolha uma tabela e remova uma ou mais tuplas. Use uma condição de localização (um predicado) para encontrar as tuplas a serem removidas.

4. Faça as seguintes alterações no esquema da base de dados: (OBS: faça escolhas que tenham sentido no domínio do problema...)

- a. (0.5) Escolha uma tabela e insira um novo atributo com as restrições apropriadas. Use **CHECK** e valor *default*. Explique o comportamento do SGBD com relação ao conteúdo da tabela: como ficam os valores deste novo atributo nas tuplas previamente existentes? Faça testes.
- b. (0.5) Escolha uma tabela e insira uma chave secundária.
- c. (0.5) Escolha uma tabela com chave primária (PK) definida por **apenas um** atributo, e que seja referenciada por uma chave estrangeira (FK) de outra tabela. Tente remover a PK. Teste o comando com e sem **CASCADE**. Explique o que acontece. No caso do **CASCADE**, explique o que acontece na tabela em que havia a PK e na tabela em que havia a FK (use comando **DESCRIBE**).
- d. (0.5) Escolha uma tabela que tenha **check**:
 - i. pesquise o comando e desative a restrição de **check**
 - ii. insira uma tupla que viole a condição de **check** previamente existente
 - iii. pesquise o comando e reative a restrição de **check** da tabela. Explique o que aconteceu com relação à tupla inserida no item anterior.