

Observações:

- O valor desta avaliação é 10 pontos.
- Ao completar a avaliação você deve salvar as respostas de cada questão em um arquivo txt com o nome do arquivo de acordo com o número da questão como segue: Q1.txt, Q2.txt, Q3.txt.
- Enviar todas as questões em um arquivo compactado (.zip, etc).

Questões

1. [4,0 pontos] Banco de dados: bancoQ1.sql

Você é o responsável pelo banco de dados de uma empresa e recebeu duas tabelas do sistema antigo populadas com alguns dados. Porém, no sistema atual da empresa as tabelas estão projetadas de modo diferente. Como responsável você precisa definir um conjunto de comandos SQL para copiar os dados das tabelas recebidas para as tabelas atuais do sistema garantindo a integridade dos dados.

Neste exercício você pode considerar que ainda não existem dados nas tabelas atuais do sistema e assim você pode migrar os dados sem se preocupar com a parte de integridade referente aos ids das tabelas antigas versus ids das tabelas novas.

Tabelas recebidas: cliente_antigo e pessoa_antigo.

Tabelas atuais do sistema: usuario e telefone.

Entregas da questão:

- Lista de comandos SQL em ordem correta de execução considerando os dados que devem ser incluídos antes dos outros.

2. [2,0 pontos] Banco de dados: empresa.sql

Uma aplicação precisa de dados disponíveis em nosso banco de dados EMPRESA e solicita que você disponibilize uma view com os dados solicitados. Os dados solicitados são: lista com o primeiro nome do funcionário, a quantidade de projetos e a quantidade de horas que cada funcionário trabalha. Para isso você precisa:

- a. criar uma view chamada **viewFuncProj**;
- b. criar um usuário chamado **appFuncProj**;
- c. dar acesso apenas de SELECT ao usuário **appFuncProj** na view **viewFuncProj**;

Entregas da questão:

- Script de criação da view;
- Script de criação do usuário;
- Script que fornece o acesso ao usuário.

3. [4,0 pontos] **Banco de dados: empresa.sql**

Conforme discutimos em aula, trabalhar com views realizando consultas diretamente nas tabelas utilizadas pelo sistema, dependendo da frequência com que ocorrem as consultas nas views e o volume de dados envolvidos, pode causar problemas de performance no uso sistema. Desta forma, você decidiu simular uma view materializada no MySQL para a consulta envolvendo a questão anterior (questão 2). Para isso você precisa:

- a. criar uma tabela chamada **funcionarios_projetos** com as colunas (funcionario, num_projetos e num_horas) para armazenar os dados solicitados;
- b. criar um script SQL para popular a tabela considerando os dados atuais do sistema;
- c. criar triggers para manter a tabela atualizada conforme os projetos e horas trabalhadas são alterados no sistema (funcionário passa a trabalhar em novo projeto, funcionário deixa de trabalhar em algum projeto, horas trabalhadas pelo funcionário em um projeto são alteradas);
- d. criar uma view chamada **viewFuncionarios_Projetos** sobre a tabela **funcionarios_projetos**.

Entregas da questão:

- Script SQL para criar a tabela funcionarios_projetos;
- Script SQL para popular a tabela funcionarios_projetos com os dados atuais do sistema;
- Script de criação das triggers;
- Script de criação da view.