

Procedural Language

- Parte 2

Alterações:

Crie uma tabela extra para controle de arquivos. A proposta é que arquivos que não são editados ou operados com frequência tendem a ser cada vez menos usados, a plataforma quer destinar acesso de qualidade aos arquivos com maior atividade. Nenhum usuário pode ter acesso à esta tabela exceto o usuário Root. Segue a estrutura da tabela:

Atividades_recentes:

Id_arquivo(FK) - id do arquivo que está sendo avaliado;

Ultima-versao – um atributo tipo date que registra a última data editada do arquivo;

Acesso – pode ser prioritário ou não prioritário.

Atividades:

Para a segunda parte do projeto implemente as seguintes visões:

- O usuário não deve ver informações de arquivos que ele não tenha acesso, ele só verá e irá operar arquivos que ele possua acesso;
- Os administradores terão acesso a toda base de dados;
- O usuário terá acesso ao histórico de operações, mas ele pode apenas visualizar ela;

Em todas as visões implementadas os ids estão ocultos e o usuário não terá acesso a eles;

Para a segunda parte do projeto implemente as seguintes roles:

- PapelUsuario – Ele terá acesso apenas aos seus arquivos ele pode selecionar, inserir e atualizar os arquivos que ele tenha permissão de acesso;
- PapelEmpresa – Ele terá acesso aos usuários que fazem parte da empresa e aos arquivos dos funcionários, ela poderá apenas visualizar os usuários e os arquivos;
- PapelAdm – Ele terá acesso à toda base de dados podendo visualizar, inserir, atualizar ou deletar.

Crie os seguintes procedures:

- Verificar_atividades – o procedure deve atualizar toda a tabela de atividades_recentes com a data atual;
- Conta_usuários - Crie um procedure que receba um id de um arquivo e conte quantos usuários diferentes o mesmo possui;
- Chavear – Atualiza o arquivo de prioritário para não prioritário e vice-versa.

- Remover_acessos – Crie um procedure que recebe o id de um arquivo e remove o acesso de todos os usuários menos do proprietário.

Crie as seguintes funções:

- Crie uma função que verifica se tem mais de 100 dias que o arquivo recebeu uma última alteração, caso sim retorne verdadeiro, caso não, retorne falso;

Crie os seguintes triggers:

- Safe_security – impeça que arquivos executáveis sejam inseridos no drive;
- Registrar_operacao – Sempre que um arquivo tiver uma nova operação atualize a data da operação na tabela atividades_recentes;
- Atualizar_acesso – Sempre que um usuário conseguir acesso a um novo arquivo, atualize os registros de arquivo dele também;