

-- INSTRUÇÕES PARA A BASE DE DADOS

BibliotecaXPTO

Um trabalho realizado por Breno Lucena // Laís Pinto // Severino Camacho

-- OBJETIVO

Criar uma base de dados de uma biblioteca para informatizar os serviços e tornar mais fácil a sua gestão mantendo sempre atualizados todos os registos e procedimentos efetuados pelos bibliotecários e leitores, utilizando ferramentas como "Functions" e "Procedures" para automatização de cálculos.

PASSOS

-- STORED PROCEDURES E PARÂMETROS NECESSÁRIOS

Criámos diversas stored procedures para efetuar todas as ações necessárias à gestão da base de dados. Segue uma explicação de como as chamar e que parâmetros usar.

-- ANTES DE COMEÇAR

Descompactar o ficheiro img.rar dentro do caminho 'C:\temp'

OPCIONAL: Se necessário apagar ou repopular os dados das tabelas usar as Stored Procedures 'EliminarConteudoTabelas' e 'PopularDB'.

BIBLIOTECA

- - 1. Consultar o número total de obras existentes:

- Executar a stored procedure 'ConsultarTotalObras',
- Não necessita de parâmetros;

- - 2. Consultar o número de obras existentes por tipo de assunto

- Executar a stored procedure 'ConsultarObrasPorGenero',
- Não necessita de parâmetros;

- - 3. Consultar a lista das dez obras mais requisitadas num intervalo de tempo por ordem decrescente

- Executar a stored procedure 'Top10ObrasRequisitadas',
- Data Inicial,
- Data Final;

- - 4. Consultar a lista dos núcleos, por ordem decrescente de requisições num intervalo de tempo

- Executar a stored procedure 'NucleosPorRequisicoes',
- Data Inicial,
- Data Final;

- - 5. Acrescentar novas obras

É necessário ter o ficheiro da imagem no local

'C:\temp\img\<nome_imagem>.jpg'

- Executar a stored procedure 'AdicionarObra',
- Título,
- Autor,
- Ano Publicação,
- Género,
- Descrição;
- Path da imagem;

- - 6. Atualizar o número de exemplares de determinada obra

Os exemplares serão sempre adicionados primeiro ao núcleo principal que é o Núcleo de Lisboa. Depois de serem adicionados podem ser transferidos para os núcleos regionais usando stored procedure do ponto 7.

- Executar a stored procedure 'AdicionarExemplaresAoNucleoPrincipal',
- ID do Bibliotecário (fazer verificação se é ou não bibliotecário),

- ID da Obra,
- Quantidade de exemplares a adicionar;

- - 7. Transferir um ou mais exemplares de uma obra, de um núcleo para outro

- Executar a stored procedure 'TransferirExemplar',
- ID do Bibliotecário (fazer verificação se é ou não bibliotecário),
- ID da Obra,
- Núcleo de origem,
- Núcleo de destino,
- Quantidade de exemplares a transferir;

- - 8. Registrar novos leitores

Adicionamos uma verificação aos parâmetros que são introduzidos. Tem também uma verificação ao número de telefone e email para não existirem repetidos.

- Executar a stored procedure 'RegistrarNovoLeitor',
- ID do Bibliotecário (fazer verificação se é ou não bibliotecário),
- Nome, (tem de haver um espaço entre o primeiro e o último nome)
- Data de nascimento, (tem de seguir a lógica de ano - mês - dia)
- Email, (tem de ter um '@' e um '.')
- Telefone, (números de 0 a 9 e não podem ser usadas letras)
- Username,
- Password; (converte para HASH)

- - 9. Suspende o acesso a requisições a leitores que tenham procedido a devoluções atrasadas em mais que três ocasiões

- Executar a stored procedure 'SuspendeAcessoLeitoresComDevolucoesAtrasadas',
- ID do Bibliotecário (fazer verificação se é ou não bibliotecário);

- - 10. Reativar o acesso a um leitor suspenso

- Executar a stored procedure 'ReativarLeitor',
- ID do Bibliotecário (fazer verificação se é ou não bibliotecário),
- ID do leitor;

- - **11.** Eliminar leitores que estejam há mais de um ano sem fazer qualquer requisição, desde que não tenham nenhuma requisição ativa nesse momento Consideramos leitores inativos todos aqueles com data de registo superior a 1 ano e com datas de requisições superiores a 1 ano. Um leitor que esteja registado mas que não faça uma requisição durante um ano também é eliminado. Os dados são inseridos da tabela de histórico de requisições, se não houver requisições guarda apenas o nome e o número de telefone.

- Executar a stored procedure 'EliminarLeitoresInativos',
- ID do Bibliotecário (fazer verificação se é ou não bibliotecário),
- ID do leitor;

LEITOR //////////////////////////////////////

- - **1.** Fazer o registo de leitor

- Executar a stored procedure 'RegistrarLeitor',
- Nome,
- Data de nascimento,
- Email,
- Telefone,
- Username,
- Password; (converte para HASH)

- - **2.** Cancelar a inscrição, devendo assumir-se que, nesse caso, é feita a devolução de todos os exemplares que possa ter requisitado e não tenha ainda devolvido

- Executar a stored procedure 'CancelarInscrição',
- Não necessita de parâmetros;

- - **3.** Pesquisar as obras disponíveis, em geral ou por núcleo e/ou por tema Todos os parâmetros são opcionais. Se não houver nenhum parâmetro, a procedure vai devolver todas as obras disponíveis no total, ou seja, contando todos os núcleos

- Executar a stored procedure 'PesquisarObrasDisponiveis'
- ID do núcleo,
- Género,
- Título,
- Autor;

- - **4.1** Obras requisitadas, em cada momento, por núcleo, devendo aparecer a indicação "ATRASSO", "Devolução URGENTE" ou "Devolver em breve", caso o prazo já tenha sido ultrapassado, falem menos de três dias para a devolução ou falem cinco dias para a devolução, respetivamente

- Executar a stored procedure 'VerificarSituacaoAtual',
- ID do leitor,

- - **4.2** Histórico de todas as obras requisitadas, podendo opcionalmente indicar um agrupamento por núcleo e/ou um intervalo de datas

- Executar a stored procedure 'HistoricoRequisicoesLeitor',
- ID do leitor,
- Data de início,
- Data final;

OUTRAS SP'S IMPORTANTES //////////////////////////////////////

- - **1.** Efetuar requisição de livro

- Executar a stored procedure 'FazerRequisicaoLivro',
- ID do leitor,
- ID da Obra,
- ID do núcleo
- ID do bibliotecário;

- - **2.** Efetuar devolução de livro

- Executar a stored procedure 'DevolverObra',
- ID do leitor,
- ID da requisição,

- - **3.** Limpar dados das tabelas

- Executar a stored procedure 'EliminarConteudoTabelas',
- Não necessita de parâmetros;

- - **4.** Colocar dados pré existentes nas tabelas

- Executar a stored procedure 'PopularDB',
- Não necessita de parâmetros;