
Licenciatura em Engenharia Informática e de Computadores

Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores

3ª Aula de exercícios, 11 de Novembro de 2019

Sistemas de Informação II

Objectivos de aprendizagem - Os alunos no final da aula de laboratório devem ser capazes de:

- (i) Utilizar ADO.NET para aceder a uma fonte de dados;
- (ii) Gerir correctamente o tempo de vida dos recursos críticos, libertando-os assim que estes não forem necessários;
- (iii) Utilizar correctamente o tipo `DbDataReader`;
- (iv) Utilizar parâmetros na execução de comandos ADO.NET;
- (v) Parametrizar correctamente os comandos ADO.NET para a execução de procedimentos armazenados;
- (vi) Utilizar correctamente contextos transaccionais para garantir coerência nos dados.

Guião

1. Abra e execute o *script* `db_model.sql`, disponível na *paste code*.
Nota: Caso use um servidor local instalado na sua máquina, verifique que foram criados dois utilizadores, User1 e User2 na base de dados. Devem ser estes os utilizadores a usar no acesso à BD.
2. Abra a solução `APADO.NET` no Visual Studio, disponível na mesma pasta, e observe a estrutura da classe `App`. Não altere nem acrescente nenhum código na zona denominada `'DO_NOT_CHANGE_NOTHING_IN_THIS_REGION_'`.
3. Resolva os problemas de compilação existentes e corra o programa.
4. Com as credenciais recolhidas da consola, construa uma `ConnectionString` para que seja possível fazer *login* na base de dados sem gerar excepções. Considere os utilizadores criados na alínea 1. Garanta que todas as ligações usadas no programa são retiradas do *Connection Pool* e que cada utilizador pode no máximo ter acesso a 10 ligações.
5. Adicione o método `void printResults(IEnumerable results)` que apresenta na consola os tuplos iterados por *results*.

Nota: Pode acrescentar os métodos auxiliares que achar convenientes para a sua implementação.

6. Implemente o método `void ListStudent()` que lista os estudantes existentes na base de dados. Use o método `void printResults(IEnumerator)` implementado anteriormente.
7. Implemente o método `void ListCourse()` que lista as disciplinas existentes na base de dados. Use o método `void printResults(IEnumerator)` implementado anteriormente.
8. Implemente o método `RegisterStudent` que insere um estudante na base de dados, recolhendo a informação necessária da consola.
9. Implemente o método `void EnrolStudent()` que inscreve um estudante numa disciplina, recebendo da consola o seu nome e o código da disciplina. Se o estudante não existir, deve usar o método `void RegisterStudent()` para o criar. Note que para inscrever um aluno é necessário que o procedimento armazenado `CanEnrolStudent` devolva 1. Caso contrário, não pode ser feita a inscrição. Garanta que a base de dados fica sempre coerente mesmo em caso de erro.