

## Descrição Exercício JDBC

**Objetivo:** Este exercício tem como objetivo permitir a fixação dos conceitos relacionados ao JDBC.

### Pré-condições:

1. Este exercício utiliza, como ponto de partida, o arquivo **JA\_MOD05\_STARTUP.zip**.

### Passos para execução do exercício

#### Parte 1

1. Criar o pacote **br.com.qualiti.java.avancado.modulo05** no diretório **src**
2. Criar uma classe chamada **TesteJDBC**.
3. No método **main** da classe carregar o driver JDBC do HSQLDB (*org.hsqldb.jdbcDriver*), criar uma conexão com o banco **qib** (url: *jdbc:hsqldb:hsqldb://localhost:9090/qib*), criar um Statement a partir dessa conexão e executar uma consulta listando todos os clientes no terminal.
4. Melhore o exemplo criando uma classe **JDBCConnectionUtil** que gerencia a conexão aberta pela aplicação.

#### Parte 2

1. Observar a implementação do **RepositorioClientesBDR** e executar a aplicação testando as funcionalidades de clientes a partir da fachada.
2. Implementar o repositório de contas **RepositorioContasBDR** que acesse um banco de dados relacional, similar ao **RepositorioClientesBDR**. A classe deve implementar a interface **RepositorioContas** (deve implementar todos os métodos). Em seguida, substituir a inicialização do cadastro de contas na **Fachada** para que esta receba a implementação do **RepositorioContasBDR**. Após esse passo, antes de testar, alterar os métodos **transferir**, **creditar** e **debitar** do **CadastroContas** para que eles atualizem a Conta no banco de dados (chamar método **atualizar**).
3. Crie o método `public void atualizar(ContaAbstrata c1, ContaAbstrata c2) throws ContaInexistenteException, ErroAcessoRepositorioException, NO` **RepositorioContasBDR** e no **RepositorioContas**, que recebe duas contas e atualiza ambas no banco de dados dentro de uma transação. Altere o método **transferir** do **CadastroContas** para usar este método. Teste o método. Qual o problema desse tipo de implementação de transação?
4. Completar a implementação do modelo de dados apresentado. Contemplando a classe **Endereço** e a classe **Gerente**.
5. **(Opcional)** Entender a implementação das classes do pacote **qualiti.persistência** e inserir na implementação do projeto o controle transacional para a operação de transferência entre contas, usando a classe **JDBCConnectionUtil**. Aguardar explicações do instrutor.