

Hortolândia, 13 de novembro de 2013.

Aulas 61 e 62 – Acesso a Banco de Dados usando JDBC – Parte II

1. Introdução

Na presente aula será criada uma classe Java de acesso a Banco de Dados. Esta classe utilizará muitos dos comandos e classes utilizados na aula passada, acessando desta forma o banco de dados MySQL por meio do JDBC.

A classe a ser criada é: **AcessoBD.java**.

Os métodos a serem criados nesta classe são: **conecta()**, **executa()**, **consulta()** e **fechaConexao()**.

A principal diferença desta classe para a que foi feita na aula anterior é que esta classe encapsula os comandos de acesso e manipulação da base de dados em métodos da classe AcessoBD, a qual pode, portanto, ser incorporada a diferentes projetos orientados a objetos que os alunos vierem a realizar.

2. Criação do Projeto

O projeto pode ser criado com o nome **AcessoBancoDados**. A classe principal, de mesmo nome, será responsável por instanciar um objeto da classe AcessoBD, e executar os seus métodos.

Além da classe principal, o projeto de exemplo conta com uma classe "AcessoBD.java", que possui os métodos para a conexão, o acesso (leitura e escrita) e o fechamento da conexão com o banco de dados.

Obs.: Em cada novo projeto criado, lembrar de adicionar o caminho do driver "Conector J" (*mysql connector*) ao projeto. Isto pode ser feito clicando-se com o botão direito no nome do projeto, escolhendo-se "Propriedades", em seguida selecionando-se a aba "Bibliotecas", pressionando-se o botão "Adicionar JAR/Pasta", e escolhendo-se o conector mysql no caminho em que ele estiver instalado no computador.

3. Classe AcessoBancoDados

A classe **AcessoBancoDados** será a classe principal da aplicação, e contará com um método main().

A primeira versão desta classe, inserindo um novo cliente e posteriormente listando os clientes já cadastrados, está descrita na Listagem 1.

```
package acessobancodados;
import java.sql.ResultSet;
import java.sql.SQLException;
public class AcessoBancoDados {
    public static AcessoBD acessoBD;
    public static ResultSet rs = null;
    public static String pedidoSQL1;
    public static String pedidoSQL2;
    public static String nome;
    public static String endereco;
    public static String codigo;
    public static void main(String[] args){
        pedidoSQL1 = "INSERT INTO cliente VALUES (null, 'Afonso
Santana', 'Av.Santana', '4345345', '19-81339000', 'jonas@gmail.com');";
```

```
pedidoSQL2 = "SELECT codigo, nome, endereco FROM cliente;";
acessoBD = new AcessoBD();
acessoBD.conecta();
acessoBD.executa(pedidoSQL1);
rs = acessoBD.consulta(pedidoSQL2);
try{
    while(rs.next()){
        codigo = rs.getString("codigo");
        nome = rs.getString("nome");
        endereco = rs.getString("endereco");
        System.out.println("Código: " + codigo);
        System.out.println("Nome: " + nome);
        System.out.println("Endereço: " + endereco);
    }
}
catch(SQLException e){
    System.out.println("Exceção: " + e.toString());
}

acessoBD.fechaConexao();
}
```

Listagem 1: Classe AcessoBancoDados.java.

4. Classe AcessoBD

```
/* Data de Criação: 10/11/2013
 * Autor: Leandro C. Ledel
 * Disciplina / Curso: LOG A2 / ADS - IFSP
 * Última Alteração: 13/11/2013
 */
package acessobancodados;
import java.sql.*;
public class AcessoBD {
    private Connection con;
    private Statement stm;
    public AcessoBD(){
        con = null;
        stm = null;
    }
    public void conecta(){
        try{
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://localhost/testejava?
user=root");
```

```
        System.out.println("Conexão bem sucedida!");
        stm = con.createStatement();
    }
    catch(ClassNotFoundException cnf){
        System.out.println("Exceção: " + cnf.toString());
    }
    catch(SQLException se){
        System.out.println("Exceção: " + se.toString());
    }
}
public void executa(String pedido){
    try{
        stm.executeUpdate(pedido);
        System.out.println("Execução de Pedido Completada!");
    }
    catch(SQLException e){
        System.out.println("Exceção: " + e.toString());
    }
    catch(NullPointerException npe){
        System.out.println("Exceção: " + npe.toString());
    }
}
public ResultSet consulta(String consulta){
    ResultSet rs = null;
    try{
        rs = stm.executeQuery(consulta);
    }
    catch(SQLException e){
        System.out.println("Exceção: " + e.toString());
    }
    finally{
        return rs;
    }
}
public void fechaConexao(){
    try{
        stm.close();
        con.close();
    }
    catch(SQLException e){
        System.out.println("Exceção: " + e.toString());
    }
}
}
```

Listagem 2: Classe de conexão com o banco de dados MySQL.

Obs.: A classe AcessoBD.java foi desenvolvida para acessar um banco MySQL que está protegido por senha. Se o banco não tiver configurada uma senha de acesso, a linha:

```
con = DriverManager.getConnection("jdbc:mysql://localhost/testejava?user=root&password=root9");
```

deve ser substituída por:

```
con = DriverManager.getConnection("jdbc:mysql://localhost/testejava?user=root");
```

5. Referências Bibliográficas

BORATTI, Isaias Camilo. **Programação Orientada a Objetos usando Delphi**. Quarta Edição. Editora Visual Books. Florianópolis, 2007.

DEITEL, H.M., DEITEL, P.J. **Java – Como programar**. Terceira edição. Porto Alegre: Bookman Editora, 2001.

HORSTMAN, C. **Conceitos de Computação com o Essencial de Java**. 3ª Edição. Porto Alegre: Bookman, 2003.

ORACLE. **API da classe ArrayList**. URL: <http://docs.oracle.com/javase/6/docs/api/java/util/ArrayList.html>. Última Consulta: 13/11/2013.

SANTOS, Rafael. **Introdução à Programação orientada a objetos usando Java**. 8ª Reimpressão. Rio de Janeiro: Campus - Elsevier, 2003.