

DESENVOLVIMENTO DE APLICAÇÕES PARA INTERNET

Parte VIII

Prof. Msc. Felipe Diniz Dallilo



Banco de dados

- Como vimos anteriormente em outras disciplinas, os bancos de dados ou bases de dados são um conjunto de arquivos relacionados entre si com registros sobre pessoas, lugares ou coisas. São coleções organizadas de dados...



Gerando uma conexão no java

Classe com método para gerar uma conexão com o banco de dados:

```
public class ConnectionFactory {  
    public Connection getConnection() {  
        try {  
            return DriverManager.getConnection(  
                "jdbc:mysql://localhost/bd", "login", "senha");  
        } catch (SQLException e) {  
            throw new RuntimeException(e);  
        }  
    }  
}
```

Obtendo a Conexão

- Com o comando abaixo é possível obter a conexão (gerada no slide anterior):

```
Connection con = new ConnectionFactory().getConnection();
```

Registrar Driver JDBC

- Até a versão 3 do JDBC, antes de chamar o `DriverManager.getConnection()` era necessário registrar o driver JDBC que iria ser utilizado através da chamada ao método (no caso do MySQL):

`Class.forName("com.mysql.jdbc.Driver");`

O mesmo carregava essa classe e se comunicava com o `DriverManager`.

Import do jar

- Existe uma implementação do JDBC por banco de dados com nomes distintos, quando estiver trabalhando com um BD, é necessário adicionar a biblioteca ao projeto (como é feito com os demais jars).
- Se consultar pelas frases abaixo é possível encontrara rapidamente na internet o .jar adequado para o banco do projeto:
- “JDBC Driver” ou “JDBC **BANCO_DE_DADOS_EM_UTILIZAÇÃO**”

Abrir e Fechar conexão

O objeto gerado através da chamada a fábrica de conexão já está iniciado, portanto, é necessário apenas fechar a conexão após o uso:

```
Connection connection = new ConnectionFactory().getConnection();  
System.out.println("Conexão aberta!");  
connection.close();
```

Inserir dados no bd

Para inserir dados no BD é adequado utilizar os comandos abaixo:

```
String sql = "insert into contatos" +  
            " (nome,email,endereco,dataNascimento)" +  
            " values (?, ?, ?, ?)";
```

```
PreparedStatement stmt = con.prepareStatement(sql);  
stmt.setString(1, "aluno1");  
stmt.setString(2, "aluno@uniara.com.br");  
stmt.setString(3, "Expressa");
```


Inserir dados no bd

```
stmt.setDate(4, new java.sql.Date(  
Calendar.getInstance().getTimeInMillis()));
```

```
stmt.execute();  
stmt.close();
```

```
System.out.println("Registro inserido com sucesso!");
```

```
con.close();
```

Remover registros no bd

Para remover dados no BD:

```
PreparedStatement stmt =  
connection.prepareStatement("delete" +  
                             "from contatos where  
id=?");  
stmt.setLong(1, 1); // Id do contato.  
stmt.execute();  
stmt.close();
```

Atualizar registros

```
String sql = "update contatos set nome=?, email=?, endereco=?"  
+  
    "dataNascimento=? where id=?";  
PreparedStatement stmt = connection.prepareStatement(sql);  
stmt.setString(1, "aluno 2");  
stmt.setString(2, "aluno2@uniara.com");
```

Atualizar registros

```
stmt.setString(3, "Via expressa ");  
stmt.setDate(4, new Date(contato.getDataNascimento()  
    .getTimeInMillis()));  
stmt.setLong(5, "1"); // Id do contato  
stmt.execute();  
stmt.close();
```

Listar dados no bd

O código abaixo é utilizado para buscar dados no BD:

```
Connection con = new ConnectionFactory().getConnection();  
PreparedStatement stmt = con.prepareStatement("select * from  
contatos");  
// executa um select  
ResultSet rs = stmt.executeQuery();  
// percorre o ResultSet
```


Listar dados no bd

```
while (rs.next()) {  
    String nome = rs.getString("nome"); //coluna  
    String email = rs.getString("email"); // coluna  
}  
rs.close();  
stmt.close();  
con.close();
```

Inserir dados de objetos

Para inserir dados no BD é adequado utilizar os comandos abaixo:

```
String sql = "insert into contatos" +  
            " (nome,email,endereco,dataNascimento)" +  
            " values (?, ?, ?, ?)";
```

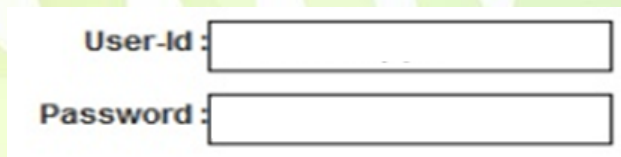
```
PreparedStatement stmt = con.prepareStatement(sql);  
stmt.setString(1, p.getNome());  
stmt.setString(2, p.getEmail());  
stmt.setString(3, p.getEndereco());  
stmt.setDate(4, p.getDataNascimento());
```

Listar dados com Objetos

```
List<Pessoa> lista = new ArrayList<Pessoa>();  
while(rs.next()){  
    Pessoa p = new Pessoa();  
    p.setNome(rs.getString("nome"));  
    p.setEmail(rs.getString("email"));  
    lista.add(p);  
}
```

SQL Injection

- É uma ameaça que permite a inserção de comandos SQL pelos formulários de forma personalizada e indevida.



A login form with two input fields. The first field is labeled 'User-Id' and the second is labeled 'Password'. Both fields are empty and have a light gray border.

Select * from User where id = “+id+” and password = “+password+” ;

SQL Injection

Select * from User where id = ""+id+" and password = ""+password+" ;

User-Id :	<input type="text" value="' OR 1= 1; /*"/>
Password :	<input type="text" value="*/--"/>



Select * from User where id = ""+id+" or 1 = 1; /*and password = ""+password+" ;

Select * from User where id = " or 1 = 1; /*and password = ""+password+" */--'

Queries Tratadas

Utilize o `PreparedStatement` para validar os parâmetros de entrada da sua consulta e assegurar que os parâmetros estão sendo associados de acordo com o esperado.

Referências

- Deitel, Paul J.; Deitel, Harvey M. Java: como programar - 8ª edição. Editora Pearson. ISBN: 9788576055631
- Barnes, David.; Kölling, Michael. Programação Orientada a Objetos com Java: uma introdução prática usando o Blue J - 1ª edição. 2013. Editora Pearson. ISBN: 9788576050124
- Puga, Sandra; Rissetti, Gerson. Lógica de Programação e Estruturas de Dados: com aplicações em Java. 1ª edição. 2013. Editora Pearson. ISBN: 9788587918826
- Lemay, Laura; Colburn, Rafe; Tyler, Denise. Aprenda a Criar Páginas Web com HTML e XHTML em 21 Dias. Editora Pearson. 2013. 1ª Edição. ISBN: 9788534614283
- Chak, Andrew. Como Criar Sites Persuasivos. Editora Pearson. 2012. 1ª Edição. ISBN: 9788534615112
- Fábio Flatschart; Clécio Bachini; Cesar Cusin. Open Web Platform. Editora Pearson. 2013. 1ª Edição. ISBN: 9788574526140