



Nesta aula de laboratório iremos resolver alguns exercícios relacionados com integração de dados, nomeadamente abordando o problema da integração de dados representados em XML com dados provenientes de bases de dados relacionais.

A classe Java cujo código se apresenta abaixo mostra como aceder ao conteúdo de uma base de dados relacional através da API *Java Database Connectivity* (JDBC), convertendo os resultados obtidos através de uma interrogação SQL para um formato de representação em XML – veja-se o código do método `jdbc_execute()`.

```
import java.sql.*;
import java.io.*;
import javax.xml.*;
import org.xml.sax.InputSource;
import javax.xml.parsers.*;
import javax.xml.transform.dom.*;

public class JDBCExtensions {

    public static Connection jdbc_connect (String d, String cs, String u, String p)
        throws Exception {
        DriverManager.registerDriver((java.sql.Driver)(Class.forName(d).newInstance()));
        Connection conn = DriverManager.getConnection (cs, u, p);
        return conn;
    }

    public static void jdbc_close ( Connection c ) throws Exception {
        c.close();
    }

    public static int jdbc_execute_update ( Connection c, String u )
        throws Exception {
        Statement stmt = c.createStatement ();
        return stmt.executeUpdate (u);
    }

    public static org.w3c.dom.Node jdbc_execute ( Connection c, String q )
        throws Exception {
        Statement stmt = c.createStatement ();
        ResultSet rset = stmt.executeQuery (q);
        ResultSetMetaData meta = rset.getMetaData();
        ByteArrayOutputStream os = new ByteArrayOutputStream();
        PrintWriter out = new PrintWriter(os);
        out.print("<relation>");
        while (rset.next ()) {
            out.print("<tuple>");
            for (int i=1; i<= meta.getColumnCount(); i++) {
                out.print("<column name=\""+ meta.getColumnLabel(i) + "\">");
                out.print(rset.getString (i));
                out.print("</column>");
            }
            out.print("</tuple>");
        }
        out.print("</relation>");
        out.close();
        InputSource in = new InputSource(new ByteArrayInputStream(os.toByteArray()));
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        return factory.newDocumentBuilder().parse(in).getDocumentElement();
    }
}
```

Os métodos definidos na classe Java apresentada podem efectivamente ser usados como funções de extensão à biblioteca de funções da XPath 2.0, permitindo a combinação de expressões XQuery para manipular dados representados em XML com interrogações em SQL para o acesso a bases de dados relacionais.

Para mais informação sobre a API *Java Database Connectivity* (JDBC) pode ser consultada a [documentação do Java](#). Mais informações sobre a utilização de expressões XQuery com funções de extensão definidas em Java pode ser obtida na [documentação do QizX](#).

### Exercício 1

Como parte integrante dos serviços associados ao cluster [sigma.ist.utl.pt](http://sigma.ist.utl.pt), todos os alunos do IST têm acesso a uma conta no servidor de bases de dados relacionais (SGBD) MySQL que se encontra no endereço [db.ist.utl.pt](http://db.ist.utl.pt).

Mais informações sobre o acesso e utilização deste serviço de bases de dados podem ser consultadas na página dos serviços de informática do IST (CIIST), especificamente no documento que se encontra no URL <https://ciist.ist.utl.pt/servicos/sql.php>.

O SGBD MySQL é actualmente um dos mais usados no desenvolvimento de aplicações Web, e existem no site oficial do projecto drivers de ligação à API JDBC. Especificamente, estes drivers encontram-se em <http://www.mysql.com/products/connector/>. Para aceder a uma das bases de dados MySQL em [db.ist.utl.pt](http://db.ist.utl.pt) através dos métodos da classe `JDBCExtensions`, um utilizador do sistema Fénix com username `istXXXXX` pode utilizar um *driver name* e uma *connection string* tal como indicado abaixo:

```
jdbc_connect("com.mysql.jdbc.Driver",  
            "jdbc:mysql://db.ist.utl.pt/istXXXXX",  
            "istXXXX", "password")
```

No URL <http://gibson.tagus.ist.utl.pt/~bmartins/sql-script.sql> encontra-se um script SQL que permite criar uma base de dados relacional com informação sobre artigos publicados em conferencias relacionadas com diferentes áreas das ciências da computação. Para resolver este exercício, comece por executar o script na sua base de dados MySQL do servidor que se encontra em [db.ist.utl.pt](http://db.ist.utl.pt).

De seguida, escreva uma expressão XQuery que, através dos métodos da classe `JDBCExtensions`, aceda ao conteúdo da base de dados relacional e apresente todos os títulos de artigos que nela se encontram definidos.

**Exercício 2**

Atente para a resolução dos exercícios sobre integração de dados da aula de laboratório 7, nomeadamente a resolução do Exercício 1 no qual se identificou um formato de representação comum para fontes de dados distintas com informação sobre artigos científicos.

Escreva uma função *wrapper* que aceda aos conteúdos da base de dados relacional do Exercício 1 da presente aula de laboratório e represente os dados no formato comum identificado para o exercício da Aula de Laboratório 7. De seguida, altere o mediador produzido no contexto do exercício da Aula de Laboratório 7 por forma a que o mesmo considere a nova fonte de dados relacional.

**Exercício 3**

Atente novamente para a resolução dos exercícios sobre integração de dados da aula de laboratório 7, especificamente para as funções *wrapper* que representam a informação sobre as diferentes fontes de dados Web num formato representação comum em XML.

Escreva uma função em XQuery que, usando os métodos da classe *JDBCExtensions*, armazenem a informação das várias fontes de dados sob a forma de uma base de dados relacional.