

Curso de Especialização Tecnológica

Tecnologia e Programação de Sistemas de Informação – 3ª Edição

Componente de formação: Integração de SI – Ferramentas

JAXB

Introdução - JAXB

Java Architecture para XML Binding é uma norma do Java que define como os objetos do Java são convertidos de e para XML. Utiliza um conjunto padrão de distribuição e alojamento.

Define uma API para leitura e escrita de objetos Java para e de documentos XML.

É possível definir diferentes implementações para este padrão. JAXB define um serviço que permite uma seleção das implementações do JAXB.

Recorrendo a estes padrões é muito mais fácil manipular o XML através de Java.

Anotações para indicar elementos de destaque:

Anotação	Descrição
@XmlElement(namespace = "namespace")	Define o root element da árvore XML
@XmlType(propOrder = { "field2", "field1",... })	Permite definir a ordem de cada campo a escrever em cada campo
@XmlElement(name = "neuName")	Define o elemento XML a ser usado.

Exercício

Cria o seguinte package, com as respetivas classes:

```
package de.vogella.xml.jaxb.model;

import javax.xml.bind.annotation.XmlElement;
import javax.xml.bind.annotation.XmlRootElement;
import javax.xml.bind.annotation.XmlType;

@XmlRootElement(name = "book")
// If you want you can define the order in which the fields are written
// Optional
@XmlType(propOrder = { "author", "name", "publisher", "isbn" })
public class Book {

    private String name;
    private String author;
    private String publisher;
    private String isbn;

    // If you like the variable name, e.g. "name", you can easily change this
    // name for your XML-Output:
    @XmlElement(name = "title")
    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAuthor() {
        return author;
    }
}
```

Curso de Especialização Tecnológica

Tecnologia e Programação de Sistemas de Informação – 3ª Edição

```
public void setAuthor(String author) {  
    this.author = author;  
}  
  
public String getPublisher() {  
    return publisher;  
}  
  
public void setPublisher(String publisher) {  
    this.publisher = publisher;  
}  
  
public String getIsbn() {  
    return isbn;  
}  
  
public void setIsbn(String isbn) {  
    this.isbn = isbn;  
}  
}
```

```
package de.vogella.xml.jaxb.model;  
  
import java.util.ArrayList;  
  
import javax.xml.bind.annotation.XmlElement;  
import javax.xml.bind.annotation.XmlElementWrapper;  
import javax.xml.bind.annotation.XmlRootElement;
```

//This statement means that class "Bookstore.java" is the root-element of our example

@XmlRootElement(namespace = "de.vogella.xml.jaxb.model")

public class Bookstore {

// XmlElementWrapper generates a wrapper element around XML representation

@XmlElementWrapper(name = "bookList")

// XmlElement sets the name of the entities

@XmlElement(name = "book")

private ArrayList<Book> bookList;

private String name;

private String location;

public void setBookList(ArrayList<Book> bookList) {

this.bookList = bookList;

}

public ArrayList<Book> getBooksList() {

return bookList;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getLocation() {

return location;

}

public void setLocation(String location) {

this.location = location;

}

Curso de Especialização Tecnológica

Tecnologia e Programação de Sistemas de Informação – 3ª Edição

```
}
```

Agora crie o ficheiro para testar a leitura e escrita dos ficheiros anteriores.

```
package test;

import java.io.File;
import java.io.FileReader;
import java.io.IOException;
import java.util.ArrayList;

import javax.xml.bind.JAXBContext;
import javax.xml.bind.JAXBException;
import javax.xml.bind.Marshaller;
import javax.xml.bind.Unmarshaller;

import de.vogella.xml.jaxb.model.Book;
import de.vogella.xml.jaxb.model.Bookstore;

public class BookMain {

    private static final String BOOKSTORE_XML = "./bookstore-jaxb.xml";

    public static void main(String[] args) throws JAXBException, IOException
    {

        ArrayList<Book> bookList = new ArrayList<Book>();

        // create books
        Book book1 = new Book();
        book1.setIsbn("978-0060554736");
```

```

book1.setName("The Game");
book1.setAuthor("Neil Strauss");
book1.setPublisher("Harpercollins");
bookList.add(book1);

Book book2 = new Book();
book2.setIsbn("978-3832180577");
book2.setName("Feuchtgebiete");
book2.setAuthor("Charlotte Roche");
book2.setPublisher("Dumont Buchverlag");
bookList.add(book2);

// create bookstore, assigning book
Bookstore bookstore = new Bookstore();
bookstore.setName("Fraport Bookstore");
bookstore.setLocation("Frankfurt Airport");
bookstore.setBookList(bookList);

// create JAXB context and instantiate marshaller
JAXBContext context = JAXBContext.newInstance(Bookstore.class);
Marshaller m = context.createMarshaller();
m.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE);

// Write to System.out
m.marshal(bookstore, System.out);

// Write to File
m.marshal(bookstore, new File(BOOKSTORE_XML));

// get variables from our xml file, created before
System.out.println();
System.out.println("Output from our XML File: ");
Unmarshaller um = context.createUnmarshaller();
Bookstore bookstore2 = (Bookstore) um.unmarshal(new
FileReader(BOOKSTORE_XML));
ArrayList<Book> list = bookstore2.getBooksList();

```

Curso de Especialização Tecnológica

Tecnologia e Programação de Sistemas de Informação – 3ª Edição

```
for (Book book : list) {  
    System.out.println("Book: " + book.getName() + " from "  
        + book.getAuthor());  
}  
}  
}
```

Bom trabalho!