

Demystifying XML Parsing for Web Dynpro Applications



Applies to:

Web Dynpro for Java UI Development, SAP NetWeaver 2004, SAP NetWeaver 7.0
For more information, visit the [Web Dynpro Java homepage](#).

Summary

This article demonstrates the technique of parsing an XML document to obtain the data contained in the XML tags which can then be used to populate the context variables and make it available for use in a Web Dynpro application. Here, I have demonstrated how the XML values can be extracted and displayed in a table.

Author: Sudeep Sura

Company: Larsen and Toubro InfoTech Limited

Created on: March 10, 2010

Author Bio



After his studies in Bachelor of Engineering, Sudeep Sura has been involved with Larsen and Toubro Infotech Limited as a SAP NetWeaver Consultant. The author has more than three years of experience on SAP technologies and has knowledge in diverse aspects of Web Dynpro development, SAP Enterprise Portal, Java/J2EE, Adobe Document Services and Adobe Interactive Forms.

Table of Contents

Getting started	3
Scenario	3
Creating an XML file	4
Creating Web Dynpro application	6
Context creation	8
Designing the table UI element.....	9
The 'dataSource' property of the table UI is set to the value of 'Book' node and the individual table columns are set to the value attributes n the Book node.Methods.....	9
Methods	10
Method 'wdDoinit' - Hook method called to initialize controller	10
Method 'mPopulate'	11
Method 'mPopulate'	11
Deploy and run the project.....	13
The output displayed in the table is populated completely from the XML file.Related Content	13
Related Content	14
Disclaimer and Liability Notice.....	15

Getting started

XML files are commonly used in SAP environment for tasks such as data transfer between Web Dynpro and interfacing technologies (XI, PI etc.), uploading bulk data from legacy systems in Web Dynpro applications, data manipulation between Web Dynpro applications and Web services etc.

Scenario

The application shows a convenient and easy to use methodology for accessing data from XML file into Web Dynpro code. The data obtained is then used to populate the context and displayed in a table on the screen.

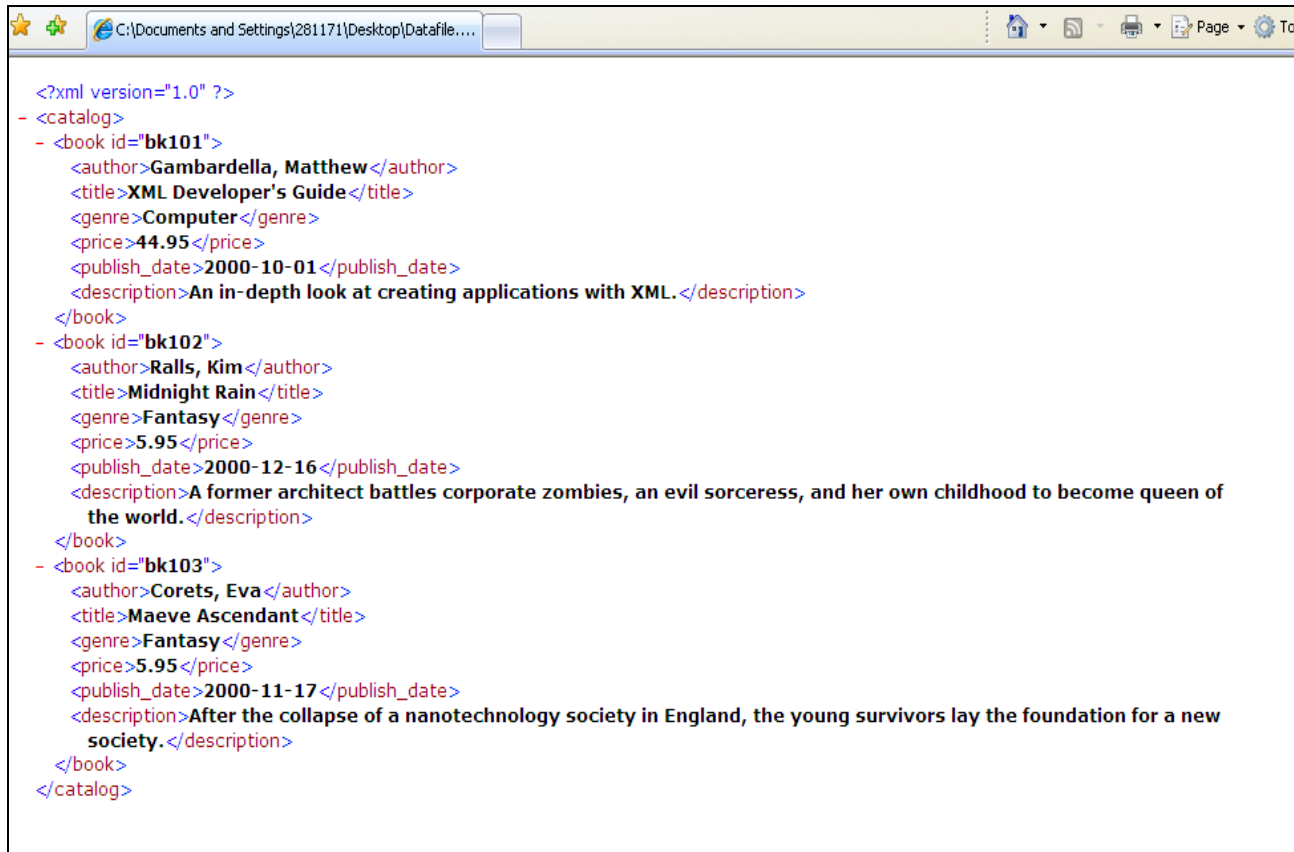
Creating an XML file

The XML file can be created by creating a new .txt file and changing the extension of the file to .xml.

The following code should be written in the XML file:

```
<?xml version="1.0"?>
<catalog>
  <book id="bk101">
    <author>Gambardella, Matthew</author>
    <title>XML Developer's Guide</title>
    <genre>Computer</genre>
    <price>44.95</price>
    <publish_date>2000-10-01</publish_date>
    <description>An in-depth look at creating applications
    with XML.</description>
  </book>
  <book id="bk102">
    <author>Ralls, Kim</author>
    <title>Midnight Rain</title>
    <genre>Fantasy</genre>
    <price>5.95</price>
    <publish_date>2000-12-16</publish_date>
    <description>A former architect battles corporate zombies,
    an evil sorceress, and her own childhood to become queen
    of the world.</description>
  </book>
  <book id="bk103">
    <author>Corets, Eva</author>
    <title>Maeve Ascendant</title>
    <genre>Fantasy</genre>
    <price>5.95</price>
    <publish_date>2000-11-17</publish_date>
    <description>After the collapse of a nanotechnology
    society in England, the young survivors lay the
    foundation for a new society.</description>
  </book>
</catalog>
```

Opening the XML file in the browser should display the XML as below:

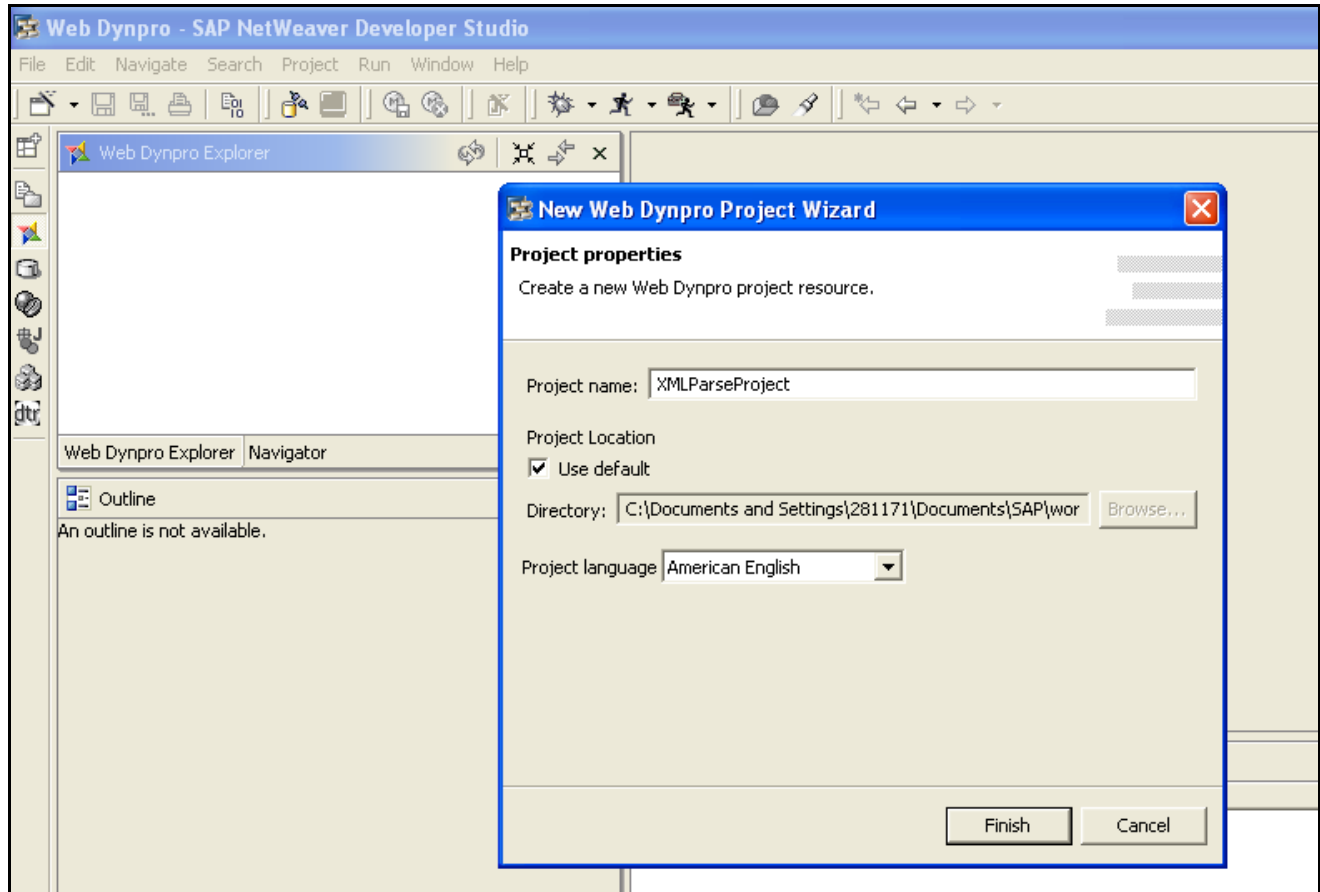
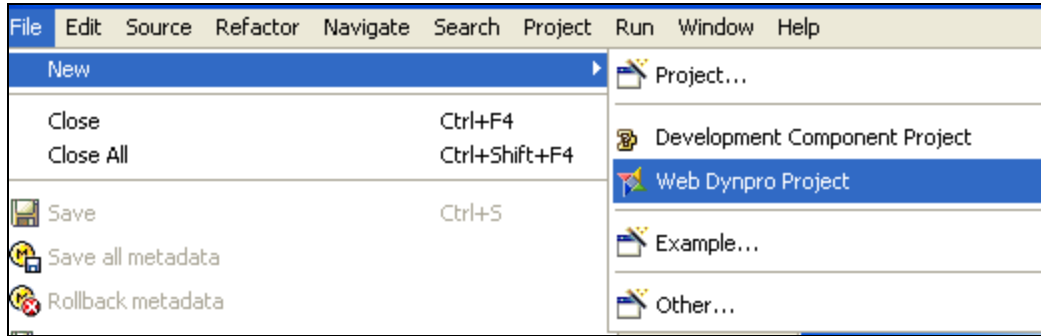


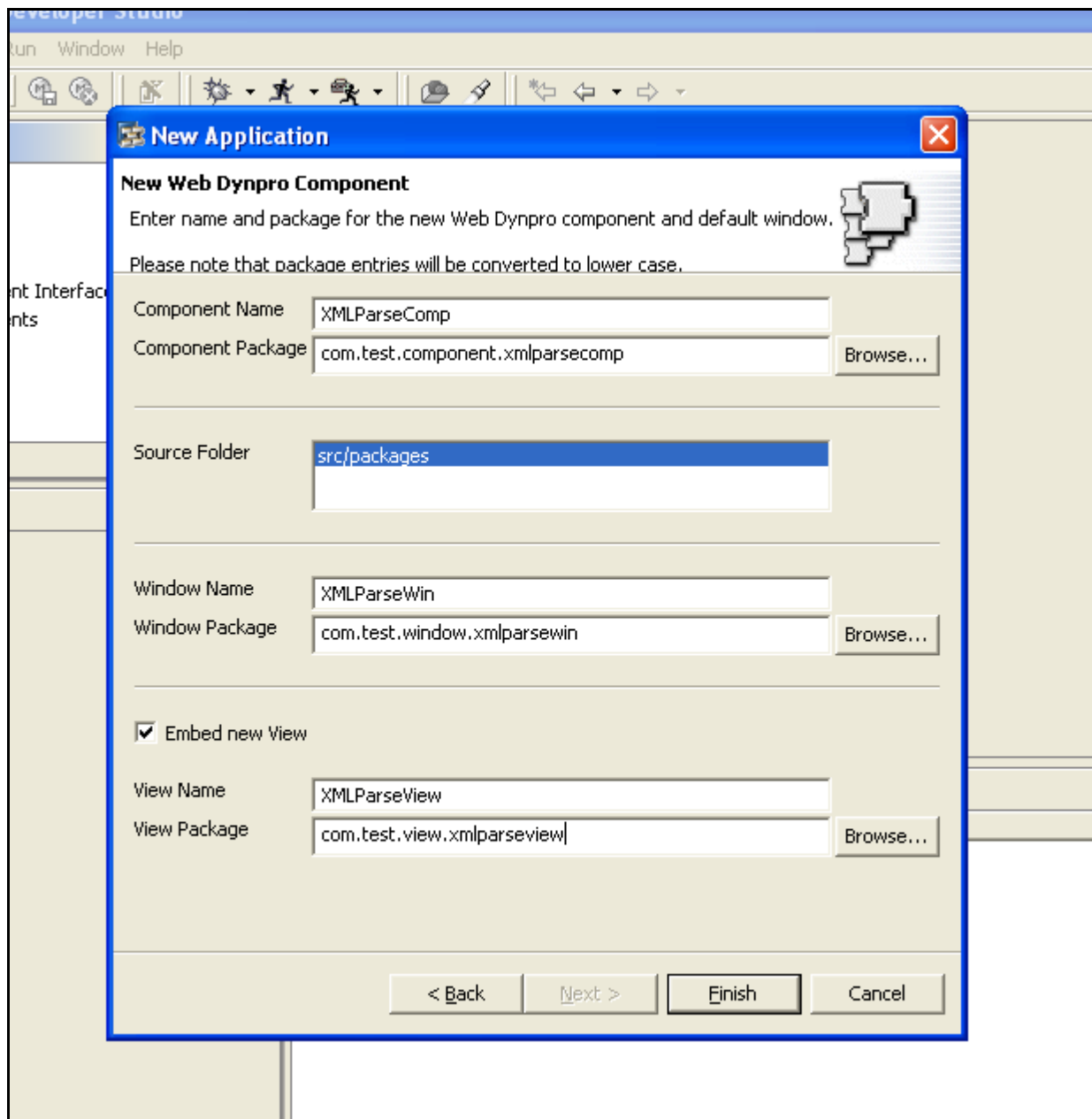
```
<?xml version="1.0" ?>
- <catalog>
- <book id="bk101">
  <author>Gambardella, Matthew</author>
  <title>XML Developer's Guide</title>
  <genre>Computer</genre>
  <price>44.95</price>
  <publish_date>2000-10-01</publish_date>
  <description>An in-depth look at creating applications with XML.</description>
</book>
- <book id="bk102">
  <author>Ralls, Kim</author>
  <title>Midnight Rain</title>
  <genre>Fantasy</genre>
  <price>5.95</price>
  <publish_date>2000-12-16</publish_date>
  <description>A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of
the world.</description>
</book>
- <book id="bk103">
  <author>Corets, Eva</author>
  <title>Maevae Ascendant</title>
  <genre>Fantasy</genre>
  <price>5.95</price>
  <publish_date>2000-11-17</publish_date>
  <description>After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new
society.</description>
</book>
</catalog>
```

Note: If the XML fails to open in a browser, it indicates that the XML is not well-formed.

Creating Web Dynpro application

Create a new Web Dynpro application named 'XMLParseProject', having a view named 'XMLParseView'.

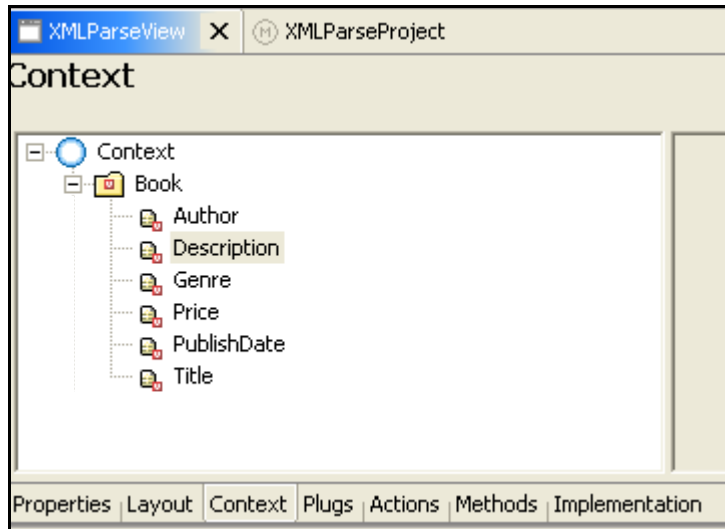




Context creation

We create a value node by the name 'Book' for the book details.

The following node will hold the data populated from the XML file, that will be displayed on the screen.



Designing the table UI element

The screenshot displays the SAP NetWeaver Developer Studio interface for designing a table UI element. The main preview window shows a table titled "Table populated from XML file" with the following data:

Author	Title	Genre	Price	PublishDate	Description
Book.Author	Book.Title	Book.Genre	Book.Price	Book.PublishDate	Book.Description
Book.Author	Book.Title	Book.Genre	Book.Price	Book.PublishDate	Book.Description
Book.Author	Book.Title	Book.Genre	Book.Price	Book.PublishDate	Book.Description
Book.Author	Book.Title	Book.Genre	Book.Price	Book.PublishDate	Book.Description
Book.Author	Book.Title	Book.Genre	Book.Price	Book.PublishDate	Book.Description

The Properties panel at the bottom shows the following properties for the table:

Property	Value
accessibilityDescription	
compatibilityMode	auto
dataSource	Book
design	standard
emptyTableText	
enabled	true
firstVisibleRow	0
firstVisibleScrollableCol	
fixedTableLayout	false
footerVisible	true

The text fields and the table fields have to be bound to the corresponding fields from the context nodes.

The 'dataSource' property of the table UI is set to the value of 'Book' node and the individual table columns are set to the value attributes in the Book node.

Methods

Method 'wdDoInit' - Hook method called to initialize controller

This method is called to initialize the controller.

The method mPopulate populates the context node from the XML file.

```
public void wdDoInit()
{
    //@@begin wdDoInit()
    wdThis.mPopulate();
    //@@end
}
```

Method 'mPopulate'

The method mPopulate parses the XML file and populates the context node.

The XML file is loaded from the mimes folder into the application code.

The XML file is then parsed to access the data held in the tags.

The data is populated into a context node, which then displays the data in a table on the screen.

```

public void mPopulate( )
{
    //@@begin mPopulate()

    String l_strXMLPath = null;
    try {

        l_strXMLPath =
WDURLGenerator.getResourcePath(wdComponentAPI.getDeployableObjectPart(), "Datafile.xml");

        FileReader l_xmlTemplate = new FileReader(l_strXMLPath);
        InputSource l_inputSource = new InputSource(l_xmlTemplate);
        DocumentBuilderFactory l_dbf = DocumentBuilderFactory.newInstance();
        DocumentBuilder l_db = l_dbf.newDocumentBuilder();
        Document l_xmlDoc = null;
        l_xmlDoc = l_db.parse(l_inputSource);

        NodeList XML_BookNode = l_xmlDoc.getElementsByTagName("book");
        for (int i = 0; i < XML_BookNode.getLength(); i++)
        {
            Node XML_Node = XML_BookNode.item(i);

            IPrivateXMLParseView.IBookElement l_bookEle =
wdContext.nodeBook().createBookElement();

            l_bookEle.setAuthor(XML_Node.getChildNodes().item(1).getChildNodes().item(0).toString());

            l_bookEle.setTitle(XML_Node.getChildNodes().item(3).getChildNodes().item(0).toString());

            l_bookEle.setGenre(XML_Node.getChildNodes().item(5).getChildNodes().item(0).toString());

            l_bookEle.setPrice(XML_Node.getChildNodes().item(7).getChildNodes().item(0).toString());

            l_bookEle.setPublishDate(XML_Node.getChildNodes().item(9).getChildNodes().item(0).toString());

            l_bookEle.setDescription(XML_Node.getChildNodes().item(11).getChildNodes().item(0).toString());

            wdContext.nodeBook().addElement(l_bookEle);
        }
    }
}

```

```
        wdComponentAPI.getMessageManager().reportSuccess("WDAliasException"+e.getMessage());
    } catch (FileNotFoundException e) {

        wdComponentAPI.getMessageManager().reportSuccess("FileNotFoundException"+e.getMessage());
    } catch (FactoryConfigurationError e) {

        wdComponentAPI.getMessageManager().reportSuccess("FactoryConfigurationError"+e.getMessage());
    } catch (SAXException e) {

        wdComponentAPI.getMessageManager().reportSuccess("SAXException"+e.getMessage());
    } catch (ParserConfigurationException e) {

        wdComponentAPI.getMessageManager().reportSuccess("ParserConfigurationException"+e.getMessage());
    } catch (IOException e) {

        wdComponentAPI.getMessageManager().reportSuccess("IOException"+e.getMessage());
    }
}

//@@end
```

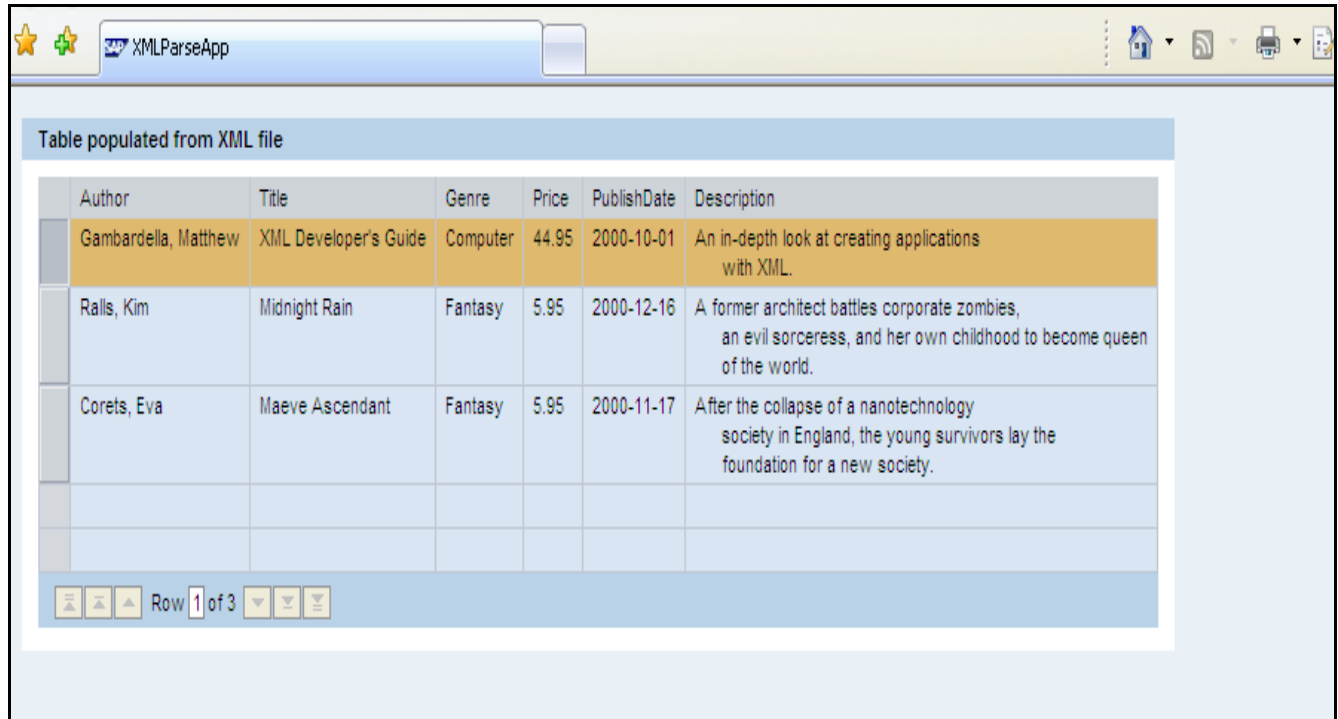
Deploy and run the project

The project is ready to be built, deployed and run.

Right click on the application name XMLParseApp and choose Deploy New Archive and Run.

Provide your SDM password when prompted. This will rebuild the project, generate the archive and deploy the application. The default web browser will be pop up with the application

The application opens in a browser as shown:



The screenshot shows a web browser window with the address bar displaying 'XMLParseApp'. The main content area features a table titled 'Table populated from XML file'. The table has six columns: Author, Title, Genre, Price, PublishDate, and Description. It contains three rows of data. Below the table, there is a pagination control showing 'Row 1 of 3'.

Author	Title	Genre	Price	PublishDate	Description
Gambardella, Matthew	XML Developer's Guide	Computer	44.95	2000-10-01	An in-depth look at creating applications with XML.
Rails, Kim	Midnight Rain	Fantasy	5.95	2000-12-16	A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.
Corets, Eva	Maeve Ascendant	Fantasy	5.95	2000-11-17	After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new society.

The output displayed in the table is populated completely from the XML file.

Related Content

[XML- Library and Parsing](#)

[Processing XML Using SAX](#)

[Document Object Model - Parsing](#)

[XML Parsing - DOM and SAX](#)

For more information, visit the [Web Dynpro Java homepage](#).

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.