

Java XML格式化，文件到XML，XML到文档



Maxsu（我是小小编）

🕒 2018-09-07 👁 808 👍 4

扩展标记语言(XML)是不同应用程序之间的消息传递和通信的流行媒介之一。由于XML是开源的，并且通过DTD和XSD提供对数据格式的控制，因此它广泛用于各种技术。

Java XML格式化程序

第三方AP将Document对象和XML消息作为String返回，这里编写了一个简单的 `XmlFormatter` 类来使用适当的缩进格式化XML并将Document对象转换为XML String。

```
package com.test.java.xmlutil;

import org.apache.xml.serialize.OutputFormat;
import org.apache.xml.serialize.XMLSerializer;
import org.w3c.dom.Document;
import org.xml.sax.InputSource;
import org.xml.sax.SAXException;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;
import javax.xml.transform.OutputKeys;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;

import java.io.IOException;
import java.io.StringReader;
import java.io.StringWriter;
import java.io.Writer;

/**
 * Utility Class for formatting XML
 * @author Pankaj
 */
public class XmlFormatter {

    /**
     *
     * @param unformattedXml - XML String
     * @return Properly formatted XML String
     */
    public String format(String unformattedXml) {
        try {
            Document document = parseXmlFile(unformattedXml);

            OutputFormat format = new OutputFormat(document);
            format.setLineWidth(65);
            format.setIndenting(true);
            format.setIndent(2);
            Writer out = new StringWriter();
            XMLSerializer serializer = new XMLSerializer(out, format);
            serializer.serialize(document);

            return out.toString();
        } catch (IOException e) {
            e.printStackTrace();
            return "";
        }
    }

    /**
     * This function converts String XML to Document object
     * @param in - XML String
     * @return Document object
     */
    private Document parseXmlFile(String in) {
        try {
            DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
            DocumentBuilder db = dbf.newDocumentBuilder();
            InputSource is = new InputSource(new StringReader(in));
            return db.parse(is);
        } catch (ParserConfigurationException e) {
            throw new RuntimeException(e);
        } catch (SAXException e) {
            throw new RuntimeException(e);
        } catch (IOException e) {
            e.printStackTrace();
        }
        return null;
    }

    /**
     * Takes an XML Document object and makes an XML String. Just a utility
     * function.
     *
     * @param doc - The DOM document
     * @return the XML String
     */
}
```

推荐

- » [JSoup快速入门](#)
- » [Linux与Unix区别](#)
- » [在Eclipse中安装Spring Tool Suite](#)
- » [Java如何从Set中获取第一个数据项？](#)
- » [Python如何列出目录中的所有文件？](#)
- » [Axis2教程](#)
- » [Java JSch在SSH Unix上运行shell命令的](#)
- » [Java程序：使用JSch和JDBC通过SSH连](#)

最受关注

- » [Java 8 BiConsumer接口示例](#)
- » [如何在Java中使用Array实现ArrayList？](#)
- » [在java中使用Array实现队列](#)
- » [Java克隆对象](#)
- » [如何解压缩zip文件中的文件](#)
- » [Java 8 谓词/断言的例子](#)
- » [使用Shell脚本更改多个文件的扩展名](#)
- » [Java如何以zip格式压缩文件](#)



```

*/
public String makeXMLString(Document doc) {
    String xmlString = "";
    if (doc != null) {
        try {
            TransformerFactory transfac = TransformerFactory.newInstance();
            Transformer trans = transfac.newTransformer();
            trans.setOutputProperty(OutputKeys.OMIT_XML_DECLARATION "yes");
            trans.setOutputProperty(OutputKeys.INDENT "yes");
            StringWriter sw = new StringWriter();
            StreamResult result = new StreamResult(sw);
            DOMSource source = new DOMSource(doc);
            trans.transform(source, result);
            xmlString = sw.toString();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    return xmlString;
}

public static void main(String args[]){
    XmlFormatter formatter = new XmlFormatter();
    String book = "<?xml version='1.0'><catalog><book id='bk101'><author>Gambardella Matthew</author><title>XML Developers C
    System.out.println(formatter.format(book));
}
}

```

要使用此类，需要 [Apache xercesImpl.jar](#)，可以从他们的网站下载。上述类的输出是格式正确的XML字符串：

```

<?xml version="1.0" encoding="UTF-8"?>
<catalog >
  <book id="bk101">
    <author > Gambardella Matthew </author >
    <title > XML Developers 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 >
</catalog >

```

这个实用程序类用于在java中格式化xml并将XML转换为Document，反之亦然。

👍 点赞 (4)

[推荐/最新教程](#)

[最新更新](#)

[站点信息](#)

[易百教程官方QQ群](#)