

JAVA JDOM PARSER - OVERVIEW

JDOM is an open source, java based library to parse XML document and it is typically java developer friendly API. It is java optimized, it uses java collection like List and Arrays. It works with DOM and SAX APIs and combines the best of the two. It is of low memory footprint and is nearly as fast as SAX.

Environment Setup

In order to use JDOM parser, you should have jdom.jar in your application's classpath. Download [jdom-2.0.5.zip](#).

When to use?

You should use a JDOM parser when:

- You need to know a lot about the structure of a document
- You need to move parts of the document around *you might want to sort certain elements, for example*
- You need to use the information in the document more than once
- You are a java developer and want to leverage java optimized parsing of XML.

What you get?

When you parse an XML document with a JDOM parser, you get the flexibility to get back a tree structure that contains all of the elements of your document without impacting the memory footprint of the application. The JDOM provides a variety of utility functions you can use to examine the contents and structure of the document in case document is well structured and its structure is known.

Advantages

JDOM gives java developers flexibility and easy maintainability of xml parsing code. It is light weight and quick API.

JDOM classes

The JDOM defines several Java classes. Here are the most common classes:

- **Document** - Represents the entire XML document. A Document object is often referred to as a DOM tree.
- **Element** - Represents an XML element. Element object has methods to manipulate its child elements, its text, attributes and namespaces.
- **Attribute** Represents an attribute of an element. Attribute has method to get and set the value of attribute. It has parent and attribute type.
- **Text** Represents the text of XML tag.
- **Comment** Represents the comments in a XML document.

Common JDOM methods

When you are working with the JDOM, there are several methods you'll use often:

- **SAXBuilder.buildxmlSource** - Build the JDOM document from the xml source.
- **Document.getRootElement** - Get the root element of the XML.
- **Element.getName** - Get the name of the XML node.

- **Element.getChildren** - Get all the direct child nodes of an element.
- **Node.getChildrenName** - Get all the direct child nodes with a given name.
- **Node getChildName** - Get first child node with given name.

Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js