

Table of Contents

If you work with XML files, it's not too easy to use the right parser because there are many of them and, if you want a simple parser without writing a lot of source code, you need some luck.

We added two classes to JVx that solves simple XML handling.

Example

XML file:

[simple.xml](#)

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- Before begin! -->
<server>

    <leer/>
    <leer></leer>

    <!-- Test: STARTPORT -->
    <startport>2001</startport>

    <audio>off</audio>
    <serial>COM1</serial>
    <buttondelay>1000</buttondelay>

    <media>
        <directory>C:\Temp\smedia</directory>
    </media>

    <!-- Test: TCP information
        second row
        third row

after an empty row
-->
    <tcp>
        <bindaddress>0.0.0.0</bindaddress>
        <user>user</user>
        <pwd>password</pwd>
    </tcp>

    <db>
        <name>SERVER_1</name>
        <path>c:\temp\pbxdb</path>
        <user>user</user>
        <pwd>password</pwd>
    </db>

    <report name="Standard">
```

```

    <fontdirectory>C:\temp\pdffonts</fontdirectory>
    <format>pdf</format>
</report>

<report name="User">
    <fontdirectory>C:\temp\xlsfonts</fontdirectory>
    <format>xls</format>
</report>

<text>
Hans
    <comment>Comment</comment>
Huber
    <comment2>Comment2</comment2>
Pauli
</text>

<text>This is a text
from Hansi Huber
</text>

</server>
<!-- After End -->

```

Read/Insert/Set tags:

```

XmlWorker xmw = new XmlWorker();
XmlNode xmn = xmw.read("simple.xml");

//-----
// Read nodes
//-----

//Access a single TAG
xmn.getNodeValue("/server/media");
//Access a TAG from a list
xmn.getNodeValue("/server/report(1)/format")

//-----
// Insert new nodes
//-----

//quick inserts
xmn.insertNode("/server/report(0)", null);
xmn.setNode("/server/report(0)/format", "ABC");

xmn.insertNode("/server/startport(1)", "ABCD");

//with nodes
XmlNode xmnInsert = new XmlNode("ABCD");

```

```

xmnsAttrib = new XmlNode(XmlNode.TYPE_ATTRIBUTE, "attrib1", xmnsInsert);
xmnsAttrib.setValue("Attribut");

xmns.insertNode("/server/startport(1)", xmnsInsert);

//-----
// Set nodes
//-----

xmns.setNode("/server/domain", "www.domain.com")

//-----
// Save
//-----

xmns.write("simple_new.xml", xmns);

```

It's easy to access and modify information in an XML file. We use '/' as separator for navigating through the hierarchy and **(n)** to access tags from a list of tags.

There are more useful methods, e.g., find the index of a tag in a list of tags. The following example creates a XML structure in memory.

```

XmlNode xmns = XmlNode.createXmlDeclaration();

xmns.insertNode("/server/a/b/c", "A");
xmns.insertNode("/server/a/b/c", "B");
xmns.insertNode("/server/a/b/c", "A");
xmns.insertNode("/server/a/b/c", "B");
xmns.insertNode("/server/a/b/c", "A");
xmns.insertNode("/server/a/b(1)/c", "A");
xmns.insertNode("/server/a/b/b", "A");

XmlNode xmnsSearch = xmnsRead.getNode("/server/a/b(0)");

//returns 0
xmnsSearch.indexOf("/c", "A", 0);
//returns 2
xmnsSearch.indexOf("/c", "A", 1);
//returns 3
xmnsSearch.indexOf("/c", "A", 3);
//returns 0
xmnsSearch.indexOf("/b", "A", 0);

xmns.insertNode("/server/a/b(1)/1", "A");
xmns.insertNode("/server/a/b(2)/2", "A");
xmns.insertNode("/server/a(1)/b/3", "A");

//returns 0
xmns.indexOf("/server/a/b/b", "A", 0)

```

The `<XmlNode>.toString()` method returns the XML representation of the node and all subnodes.

From:

<https://doc.sibvisions.com/> - **Documentation**

Permanent link:

<https://doc.sibvisions.com/jvx/common/util/xml>



Last update: **2020/06/29 14:52**