| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AttributeList.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV CLASS   [**NEXT CLASS**](http://docs.google.com/org/xml/sax/Attributes.html) | [**FRAMES**](http://docs.google.com/index.html?org/xml/sax/AttributeList.html)    [**NO FRAMES**](http://docs.google.com/AttributeList.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **org.xml.sax**

Interface AttributeList

**All Known Implementing Classes:** [AttributeListImpl](http://docs.google.com/org/xml/sax/helpers/AttributeListImpl.html)**Deprecated.** *This interface has been replaced by the SAX2* [*Attributes*](http://docs.google.com/org/xml/sax/Attributes.html) *interface, which includes Namespace support.*

public interface **AttributeList**

Interface for an element's attribute specifications.

*This module, both source code and documentation, is in the Public Domain, and comes with* ***NO WARRANTY****.* See <http://www.saxproject.org> for further information.

This is the original SAX1 interface for reporting an element's attributes. Unlike the new [Attributes](http://docs.google.com/org/xml/sax/Attributes.html) interface, it does not support Namespace-related information.

When an attribute list is supplied as part of a [startElement](http://docs.google.com/org/xml/sax/DocumentHandler.html#startElement(java.lang.String,%20org.xml.sax.AttributeList)) event, the list will return valid results only during the scope of the event; once the event handler returns control to the parser, the attribute list is invalid. To save a persistent copy of the attribute list, use the SAX1 [AttributeListImpl](http://docs.google.com/org/xml/sax/helpers/AttributeListImpl.html) helper class.

An attribute list includes only attributes that have been specified or defaulted: #IMPLIED attributes will not be included.

There are two ways for the SAX application to obtain information from the AttributeList. First, it can iterate through the entire list:

public void startElement (String name, AttributeList atts) {  
 for (int i = 0; i < atts.getLength(); i++) {  
 String name = atts.getName(i);  
 String type = atts.getType(i);  
 String value = atts.getValue(i);  
 [...]  
 }  
 }

(Note that the result of getLength() will be zero if there are no attributes.)

As an alternative, the application can request the value or type of specific attributes:

public void startElement (String name, AttributeList atts) {  
 String identifier = atts.getValue("id");  
 String label = atts.getValue("label");  
 [...]  
 }

**Since:** SAX 1.0 **See Also:**[startElement](http://docs.google.com/org/xml/sax/DocumentHandler.html#startElement(java.lang.String,%20org.xml.sax.AttributeList)), [AttributeListImpl](http://docs.google.com/org/xml/sax/helpers/AttributeListImpl.html)

| **Method Summary** | |
| --- | --- |
| int | [**getLength**](http://docs.google.com/org/xml/sax/AttributeList.html#getLength())()  **Deprecated.** Return the number of attributes in this list. |
| [String](http://docs.google.com/java/lang/String.html) | [**getName**](http://docs.google.com/org/xml/sax/AttributeList.html#getName(int))(int i)  **Deprecated.** Return the name of an attribute in this list (by position). |
| [String](http://docs.google.com/java/lang/String.html) | [**getType**](http://docs.google.com/org/xml/sax/AttributeList.html#getType(int))(int i)  **Deprecated.** Return the type of an attribute in the list (by position). |
| [String](http://docs.google.com/java/lang/String.html) | [**getType**](http://docs.google.com/org/xml/sax/AttributeList.html#getType(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)  **Deprecated.** Return the type of an attribute in the list (by name). |
| [String](http://docs.google.com/java/lang/String.html) | [**getValue**](http://docs.google.com/org/xml/sax/AttributeList.html#getValue(int))(int i)  **Deprecated.** Return the value of an attribute in the list (by position). |
| [String](http://docs.google.com/java/lang/String.html) | [**getValue**](http://docs.google.com/org/xml/sax/AttributeList.html#getValue(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)  **Deprecated.** Return the value of an attribute in the list (by name). |

| **Method Detail** |
| --- |

### getLength

int **getLength**()

**Deprecated.** Return the number of attributes in this list.

The SAX parser may provide attributes in any arbitrary order, regardless of the order in which they were declared or specified. The number of attributes may be zero.

**Returns:**The number of attributes in the list.

### getName

[String](http://docs.google.com/java/lang/String.html) **getName**(int i)

**Deprecated.** Return the name of an attribute in this list (by position).

The names must be unique: the SAX parser shall not include the same attribute twice. Attributes without values (those declared #IMPLIED without a value specified in the start tag) will be omitted from the list.

If the attribute name has a namespace prefix, the prefix will still be attached.

**Parameters:**i - The index of the attribute in the list (starting at 0). **Returns:**The name of the indexed attribute, or null if the index is out of range.**See Also:**[getLength()](http://docs.google.com/org/xml/sax/AttributeList.html#getLength())

### getType

[String](http://docs.google.com/java/lang/String.html) **getType**(int i)

**Deprecated.** Return the type of an attribute in the list (by position).

The attribute type is one of the strings "CDATA", "ID", "IDREF", "IDREFS", "NMTOKEN", "NMTOKENS", "ENTITY", "ENTITIES", or "NOTATION" (always in upper case).

If the parser has not read a declaration for the attribute, or if the parser does not report attribute types, then it must return the value "CDATA" as stated in the XML 1.0 Recommentation (clause 3.3.3, "Attribute-Value Normalization").

For an enumerated attribute that is not a notation, the parser will report the type as "NMTOKEN".

**Parameters:**i - The index of the attribute in the list (starting at 0). **Returns:**The attribute type as a string, or null if the index is out of range.**See Also:**[getLength()](http://docs.google.com/org/xml/sax/AttributeList.html#getLength()), [getType(java.lang.String)](http://docs.google.com/org/xml/sax/AttributeList.html#getType(java.lang.String))

### getValue

[String](http://docs.google.com/java/lang/String.html) **getValue**(int i)

**Deprecated.** Return the value of an attribute in the list (by position).

If the attribute value is a list of tokens (IDREFS, ENTITIES, or NMTOKENS), the tokens will be concatenated into a single string separated by whitespace.

**Parameters:**i - The index of the attribute in the list (starting at 0). **Returns:**The attribute value as a string, or null if the index is out of range.**See Also:**[getLength()](http://docs.google.com/org/xml/sax/AttributeList.html#getLength()), [getValue(java.lang.String)](http://docs.google.com/org/xml/sax/AttributeList.html#getValue(java.lang.String))

### getType

[String](http://docs.google.com/java/lang/String.html) **getType**([String](http://docs.google.com/java/lang/String.html) name)

**Deprecated.** Return the type of an attribute in the list (by name).

The return value is the same as the return value for getType(int).

If the attribute name has a namespace prefix in the document, the application must include the prefix here.

**Parameters:**name - The name of the attribute. **Returns:**The attribute type as a string, or null if no such attribute exists.**See Also:**[getType(int)](http://docs.google.com/org/xml/sax/AttributeList.html#getType(int))

### getValue

[String](http://docs.google.com/java/lang/String.html) **getValue**([String](http://docs.google.com/java/lang/String.html) name)

**Deprecated.** Return the value of an attribute in the list (by name).

The return value is the same as the return value for getValue(int).

If the attribute name has a namespace prefix in the document, the application must include the prefix here.

**Parameters:**name - the name of the attribute to return **Returns:**The attribute value as a string, or null if no such attribute exists.**See Also:**[getValue(int)](http://docs.google.com/org/xml/sax/AttributeList.html#getValue(int))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AttributeList.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV CLASS   [**NEXT CLASS**](http://docs.google.com/org/xml/sax/Attributes.html) | [**FRAMES**](http://docs.google.com/index.html?org/xml/sax/AttributeList.html)    [**NO FRAMES**](http://docs.google.com/AttributeList.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).