org.xml.sax.Attributes

getIndex

Look up the index of an attribute by XML 1.0 qualified name.
```
int getIndex(String qName)
```
Look up the index of an attribute by Namespace name.
```
int getIndex(String uri, String localPart)
```

getLength

Return the number of attributes in the list.
```
int getLength()
```

getLocalName

Look up an attribute's local name by index.
```
String getLocalName(int index)
```

getQName

Look up an attribute's XML 1.0 qualified name by index.
```
String getQName(int index)
```

getType

Look up an attribute's type by index.
```
String getType(int index)
```
Look up an attribute's type by XML 1.0 qualified name.
```
String getType(String qName)
```
Look up an attribute's type by Namespace name.
```
String getType(String uri, String localName)
```

getURI

Look up an attribute's Namespace URI by index.
```
String getURI(int index)
```

getValue

Look up an attribute's value by index.
```
String getValue(int index)
```
Look up an attribute's value by XML 1.0 qualified name.
```
String getValue(String qName)
```
Look up an attribute's value by Namespace name.
```
String getValue(String uri, String localName)
```

org.xml.sax.ContentHandler

characters

Receive notification of character data.
```
void characters(char[] ch, int start, int len)
```

endDocument

Receive notification of the end of a document.
```
void endDocument()
```

endElement

Receive notification of the end of an element.
```
void endElement(String namespaceURI,
                String localName,
                String qName)
```

endPrefixMapping

End the scope of a prefix-URI mapping.
```
void endPrefixMapping(String prefix)
```

ignorableWhitespace

Receive notification of ignorable whitespace in element content.
```
void ignorableWhitespace(char[] ch, int start, int len)
```

processingInstruction

Receive notification of a processing instruction.
```
void processingInstruction(String target, String data)
```

setDocumentLocator

Receive an object for locating the origin of SAX document events.
```
void setDocumentLocator(Locator locator)
```

skippedEntity

Receive notification of a skipped entity.
```
void skippedEntity(String name)
```

startDocument

Receive notification of the beginning of a document.
```
void startDocument()
```

startElement

Receive notification of the beginning of an element.
```
void startElement(String namespaceURI,
                  String localName,
                  String qName,
                  Attributes atts)
```

startPrefixMapping

Begin the scope of a prefix-URI Namespace mapping.
```
void startPrefixMapping(String prefix, String uri)
```

org.xml.sax.DTDHandler

notationDecl

Receive notification of a notation declaration event.
```
void notationDecl(String name,
                  String publicId, String systemId)
```

unparsedEntityDecl

Receive notification of an unparsed entity declaration event.
```
void unparsedEntityDecl(String name,
                        String publicId,
                        String systemId,
                        String notationName)
```

org.xml.sax.EntityResolver

resolveEntity

Allow the application to resolve external entities.
```
InputSource resolveEntity(String publicId,
                          String systemId)
```

org.xml.sax.ErrorHandler

error

Receive notification of a recoverable error.
```
void error(SAXParseException exception)
```

fatalError

Receive notification of a non-recoverable error.
```
void fatalError(SAXParseException exception)
```

warning

Receive notification of a warning.
```
void warning(SAXParseException exception)
```

org.xml.sax.Locator

getColumnNumber

Return the column number where the current document event ends.
```
int getColumnNumber()
```

getLineNumber

Return the line number where the current document event ends.
```
int getLineNumber()
```

getPublicId

Return the public identifier for the current document event.
```
String getPublicId()
```

getSystemId

Return the system identifier for the current document event.
```
String getSystemId()
```

org.xml.sax.XMLFilter

getParent

Get the parent reader.
```
XMLReader getParent()
```

setParent

Set the parent reader.
```
void setParent(XMLReader parent)
```

org.xml.sax.XMLReader

getContentHandler

Return the current content handler.
```
ContentHandler getContentHandler()
```

getDTDHandler

Return the current DTD handler.
```
DTDHandler getDTDHandler()
```

getEntityResolver

Return the current entity resolver.
```
EntityResolver getEntityResolver()
```

getErrorHandler

Return the current error handler.
```
ErrorHandler getErrorHandler()
```

getFeature

Look up the value of a feature.
```
boolean getFeature(String name)
```

getProperty

Look up the value of a property.
```
java.lang.Object getProperty(String name)
```

parse

Parse an XML document.
```
void parse(InputSource input)
```
Parse an XML document from a system identifier (URI).
```
void parse(String systemId)
```

setContentHandler

Allow an application to register a content event handler.
```
void setContentHandler(ContentHandler handler)
```

setDTDHandler

Allow an application to register a DTD event handler.
```
void setDTDHandler(DTDHandler handler)
```

setEntityResolver

Allow an application to register an entity resolver.
```
void setEntityResolver(EntityResolver resolver)
```

setErrorHandler

Allow an application to register an error event handler.
```
void setErrorHandler(ErrorHandler handler)
```

setFeature

Set the state of a feature.
```
void setFeature(String name, boolean value)
```

setProperty

Set the value of a property.
```
void setProperty(String name, Object value)
```

CLASSES

org.xml.sax.InputSource

A single input source for an XML entity.

InputSource

Zero-argument default constructor.
```
InputSource()
```
Create a new input source with a byte stream.
```
InputSource(java.io.InputStream byteStream)
```
Create a new input source with a character stream.
```
InputSource(java.io.Reader characterStream)
```
Create a new input source with a system identifier.
```
InputSource(String systemId)
```

getByteStream

Get the byte stream for this input source.
```
java.io.InputStream getByteStream()
```

getCharacterStream

Get the character stream for this input source.
```
java.io.Reader getCharacterStream()
```

getEncoding

Get the character encoding for a byte stream or URI.
```
String getEncoding()
```

getPublicId

Get the public identifier for this input source.
```
String getPublicId()
```

getSystemId

Get the system identifier for this input source.
```
String getSystemId()
```

setByteStream

Set the byte stream for this input source.
```
void setByteStream(java.io.InputStream byteStream)
```

setCharacterStream

Set the character stream for this input source.
```
void setCharacterStream(java.io.Reader characterStream)
```

setEncoding

Set the character encoding, if known.
```
void setEncoding(String encoding)
```

setPublicId

Set the public identifier for this input source.
```
void setPublicId(String publicId)
```

setSystemId

Set the system identifier for this input source.

```
void setSystemId(String systemId)
```

EXCEPTIONS

org.xml.sax.SAXException

SAXException

Create a new SAXException wrapping an existing exception.
```
SAXException(java.lang.Exception e)
```
Create a new SAXException.
```
SAXException(String message)
```
Create a new SAXException from an existing exception.
```
SAXException(String message, java.lang.Exception e)
```

getException

Return the embedded exception, if any.
```
java.lang.Exception getException()
```

getMessage

Return a detail message for this exception.
```
String getMessage()
```

toString

Override toString to pick up any embedded exception.
```
String toString()
```

org.xml.sax.SAXNotRecognizedException

SAXNotRecognizedException

Create a new SAXNotRecognizedException.
```
SAXNotRecognizedException(String message)
```

org.xml.sax.SAXNotSupportedException

SAXNotSupportedException

Create a new SAXNotSupportedException.
```
SAXNotSupportedException(String message)
```

org.xml.sax.SAXParseException

SAXParseException

Create a new SAXParseException from a message and a Locator.
```
SAXParseException(String msg, Locator l)
```
Create a SAXParseException from an existing Exception.
```
SAXParseException(String msg, Locator l, Exception e)
```
Create a new SAXParseException or wrap and existing Exception.
```
SAXParseException(String msg,
                  String publicId,
                  String systemId,
                  int lineNumber,
                  int columnNumber [, Exception e])
```

getColumnNumber

The column number of the end of the text where the exception occurred.
```
int getColumnNumber()
```

getLineNumber

The line number of the end of the text where the exception occurred.
```
int getLineNumber()
```

getPublicId

Get the public identifier of the entity where the exception occurred.
```
java.lang.String getPublicId()
```

getSystemId

Get the system identifier of the entity where the exception occurred.
```
java.lang.String getSystemId()
```

---

deepX

Quick Reference

Simple API for XML (SAX)
Version 2 for Java (Core)

5 May 2000

http://www.saxproject.org/

The interfaces, classes and exceptions outlined below are taken from the *org.xml.sax* package. This quick reference does not include deprecated features.