| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MessageFactory.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**](http://docs.google.com/javax/xml/soap/DetailEntry.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/soap/MimeHeader.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/soap/MessageFactory.html)    [**NO FRAMES**](http://docs.google.com/MessageFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **javax.xml.soap**

Class MessageFactory

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.xml.soap.MessageFactory**

public abstract class **MessageFactory**extends [Object](http://docs.google.com/java/lang/Object.html)

A factory for creating SOAPMessage objects.

A SAAJ client can create a MessageFactory object using the method newInstance, as shown in the following lines of code.

MessageFactory mf = MessageFactory.newInstance();  
 MessageFactory mf12 = MessageFactory.newInstance(SOAPConstants.SOAP\_1\_2\_PROTOCOL);

All MessageFactory objects, regardless of how they are created, will produce SOAPMessage objects that have the following elements by default:

* A SOAPPart object
* A SOAPEnvelope object
* A SOAPBody object
* A SOAPHeader object

In some cases, specialized MessageFactory objects may be obtained that produce messages prepopulated with additional entries in the SOAPHeader object and the SOAPBody object. The content of a new SOAPMessage object depends on which of the two MessageFactory methods is used to create it.

* createMessage()  
  This is the method clients would normally use to create a request message.
* createMessage(MimeHeaders, java.io.InputStream) -- message has content from the InputStream object and headers from the MimeHeaders object  
  This method can be used internally by a service implementation to create a message that is a response to a request.

| **Constructor Summary** | |
| --- | --- |
| [**MessageFactory**](http://docs.google.com/javax/xml/soap/MessageFactory.html#MessageFactory())() |

| **Method Summary** | |
| --- | --- |
| abstract  [SOAPMessage](http://docs.google.com/javax/xml/soap/SOAPMessage.html) | [**createMessage**](http://docs.google.com/javax/xml/soap/MessageFactory.html#createMessage())()            Creates a new SOAPMessage object with the default SOAPPart, SOAPEnvelope, SOAPBody, and SOAPHeader objects. |
| abstract  [SOAPMessage](http://docs.google.com/javax/xml/soap/SOAPMessage.html) | [**createMessage**](http://docs.google.com/javax/xml/soap/MessageFactory.html#createMessage(javax.xml.soap.MimeHeaders,%20java.io.InputStream))([MimeHeaders](http://docs.google.com/javax/xml/soap/MimeHeaders.html) headers, [InputStream](http://docs.google.com/java/io/InputStream.html) in)            Internalizes the contents of the given InputStream object into a new SOAPMessage object and returns the SOAPMessage object. |
| static [MessageFactory](http://docs.google.com/javax/xml/soap/MessageFactory.html) | [**newInstance**](http://docs.google.com/javax/xml/soap/MessageFactory.html#newInstance())()            Creates a new MessageFactory object that is an instance of the default implementation (SOAP 1.1), This method uses the following ordered lookup procedure to determine the MessageFactory implementation class to load: Use the javax.xml.soap.MessageFactory system property. |
| static [MessageFactory](http://docs.google.com/javax/xml/soap/MessageFactory.html) | [**newInstance**](http://docs.google.com/javax/xml/soap/MessageFactory.html#newInstance(java.lang.String))([String](http://docs.google.com/java/lang/String.html) protocol)            Creates a new MessageFactory object that is an instance of the specified implementation. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### MessageFactory

public **MessageFactory**()

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

### newInstance

public static [MessageFactory](http://docs.google.com/javax/xml/soap/MessageFactory.html) **newInstance**()  
 throws [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html)

Creates a new MessageFactory object that is an instance of the default implementation (SOAP 1.1), This method uses the following ordered lookup procedure to determine the MessageFactory implementation class to load:

* Use the javax.xml.soap.MessageFactory system property.
* Use the properties file "lib/jaxm.properties" in the JRE directory. This configuration file is in standard java.util.Properties format and contains the fully qualified name of the implementation class with the key being the system property defined above.
* Use the Services API (as detailed in the JAR specification), if available, to determine the classname. The Services API will look for a classname in the file META-INF/services/javax.xml.soap.MessageFactory in jars available to the runtime.
* Use the SAAJMetaFactory instance to locate the MessageFactory implementation class.

**Returns:**a new instance of a MessageFactory **Throws:** [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html) - if there was an error in creating the default implementation of the MessageFactory.**See Also:**[SAAJMetaFactory](http://docs.google.com/javax/xml/soap/SAAJMetaFactory.html)

### newInstance

public static [MessageFactory](http://docs.google.com/javax/xml/soap/MessageFactory.html) **newInstance**([String](http://docs.google.com/java/lang/String.html) protocol)  
 throws [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html)

Creates a new MessageFactory object that is an instance of the specified implementation. May be a dynamic message factory, a SOAP 1.1 message factory, or a SOAP 1.2 message factory. A dynamic message factory creates messages based on the MIME headers specified as arguments to the createMessage method. This method uses the SAAJMetaFactory to locate the implementation class and create the MessageFactory instance.

**Parameters:**protocol - a string constant representing the class of the specified message factory implementation. May be either DYNAMIC\_SOAP\_PROTOCOL, DEFAULT\_SOAP\_PROTOCOL (which is the same as) SOAP\_1\_1\_PROTOCOL, or SOAP\_1\_2\_PROTOCOL. **Returns:**a new instance of a MessageFactory **Throws:** [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html) - if there was an error in creating the specified implementation of MessageFactory.**Since:** SAAJ 1.3 **See Also:**[SAAJMetaFactory](http://docs.google.com/javax/xml/soap/SAAJMetaFactory.html)

### createMessage

public abstract [SOAPMessage](http://docs.google.com/javax/xml/soap/SOAPMessage.html) **createMessage**()  
 throws [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html)

Creates a new SOAPMessage object with the default SOAPPart, SOAPEnvelope, SOAPBody, and SOAPHeader objects. Profile-specific message factories can choose to prepopulate the SOAPMessage object with profile-specific headers.

Content can be added to this message's SOAPPart object, and the message can be sent "as is" when a message containing only a SOAP part is sufficient. Otherwise, the SOAPMessage object needs to create one or more AttachmentPart objects and add them to itself. Any content that is not in XML format must be in an AttachmentPart object.

**Returns:**a new SOAPMessage object **Throws:** [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html) - if a SOAP error occurs [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the protocol of this MessageFactory instance is DYNAMIC\_SOAP\_PROTOCOL

### createMessage

public abstract [SOAPMessage](http://docs.google.com/javax/xml/soap/SOAPMessage.html) **createMessage**([MimeHeaders](http://docs.google.com/javax/xml/soap/MimeHeaders.html) headers,  
 [InputStream](http://docs.google.com/java/io/InputStream.html) in)  
 throws [IOException](http://docs.google.com/java/io/IOException.html),  
 [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html)

Internalizes the contents of the given InputStream object into a new SOAPMessage object and returns the SOAPMessage object.

**Parameters:**in - the InputStream object that contains the data for a messageheaders - the transport-specific headers passed to the message in a transport-independent fashion for creation of the message **Returns:**a new SOAPMessage object containing the data from the given InputStream object **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if there is a problem in reading data from the input stream [SOAPException](http://docs.google.com/javax/xml/soap/SOAPException.html) - may be thrown if the message is invalid [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the MessageFactory requires one or more MIME headers to be present in the headers parameter and they are missing. MessageFactory implementations for SOAP\_1\_1\_PROTOCOL or SOAP\_1\_2\_PROTOCOL must not throw IllegalArgumentException for this reason.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MessageFactory.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**](http://docs.google.com/javax/xml/soap/DetailEntry.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/soap/MimeHeader.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/soap/MessageFactory.html)    [**NO FRAMES**](http://docs.google.com/MessageFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).