| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/QName.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/namespace/NamespaceContext.html)   NEXT CLASS | [**FRAMES**](http://docs.google.com/index.html?javax/xml/namespace/QName.html)    [**NO FRAMES**](http://docs.google.com/QName.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#17dp8vu) |

## **javax.xml.namespace**

Class QName

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

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html)

public class **QName**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html)

QName represents a **qualified name** as defined in the XML specifications: [XML Schema Part2: Datatypes specification](http://www.w3.org/TR/xmlschema-2/#QName), [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/#ns-qualnames), [Namespaces in XML Errata](http://www.w3.org/XML/xml-names-19990114-errata).

The value of a QName contains a **Namespace URI**, **local part** and **prefix**.

The prefix is included in QName to retain lexical information ***when present*** in an [XML input source](http://docs.google.com/javax/xml/transform/Source.html). The prefix is ***NOT*** used in [QName.equals(Object)](http://docs.google.com/javax/xml/namespace/QName.html#equals(java.lang.Object)) or to compute the [QName.hashCode()](http://docs.google.com/javax/xml/namespace/QName.html#hashCode()). Equality and the hash code are defined using ***only*** the Namespace URI and local part.

If not specified, the Namespace URI is set to [XMLConstants.NULL\_NS\_URI](http://docs.google.com/javax/xml/XMLConstants.html#NULL_NS_URI). If not specified, the prefix is set to [XMLConstants.DEFAULT\_NS\_PREFIX](http://docs.google.com/javax/xml/XMLConstants.html#DEFAULT_NS_PREFIX).

QName is immutable.

**Since:** 1.5 **See Also:** [XML Schema Part2: Datatypes specification](http://www.w3.org/TR/xmlschema-2/#QName),  [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/#ns-qualnames),  [Namespaces in XML Errata](http://www.w3.org/XML/xml-names-19990114-errata), [Serialized Form](http://docs.google.com/serialized-form.html#javax.xml.namespace.QName)

| **Constructor Summary** | |
| --- | --- |
| [**QName**](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String))([String](http://docs.google.com/java/lang/String.html) localPart)            QName constructor specifying the local part. |
| [**QName**](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String,%20java.lang.String))([String](http://docs.google.com/java/lang/String.html) namespaceURI, [String](http://docs.google.com/java/lang/String.html) localPart)            QName constructor specifying the Namespace URI and local part. |
| [**QName**](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String,%20java.lang.String,%20java.lang.String))([String](http://docs.google.com/java/lang/String.html) namespaceURI, [String](http://docs.google.com/java/lang/String.html) localPart, [String](http://docs.google.com/java/lang/String.html) prefix)            QName constructor specifying the Namespace URI, local part and prefix. |

| **Method Summary** | |
| --- | --- |
| boolean | [**equals**](http://docs.google.com/javax/xml/namespace/QName.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) objectToTest)            Test this QName for equality with another Object. |
| [String](http://docs.google.com/java/lang/String.html) | [**getLocalPart**](http://docs.google.com/javax/xml/namespace/QName.html#getLocalPart())()            Get the local part of this QName. |
| [String](http://docs.google.com/java/lang/String.html) | [**getNamespaceURI**](http://docs.google.com/javax/xml/namespace/QName.html#getNamespaceURI())()            Get the Namespace URI of this QName. |
| [String](http://docs.google.com/java/lang/String.html) | [**getPrefix**](http://docs.google.com/javax/xml/namespace/QName.html#getPrefix())()            Get the prefix of this QName. |
| int | [**hashCode**](http://docs.google.com/javax/xml/namespace/QName.html#hashCode())()            Generate the hash code for this QName. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/xml/namespace/QName.html#toString())()            String representation of this QName. |
| static [QName](http://docs.google.com/javax/xml/namespace/QName.html) | [**valueOf**](http://docs.google.com/javax/xml/namespace/QName.html#valueOf(java.lang.String))([String](http://docs.google.com/java/lang/String.html) qNameAsString)            QName derived from parsing the formatted String. |

| **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()), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [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** |
| --- |

### QName

public **QName**([String](http://docs.google.com/java/lang/String.html) namespaceURI,  
 [String](http://docs.google.com/java/lang/String.html) localPart)

QName constructor specifying the Namespace URI and local part.

If the Namespace URI is null, it is set to [XMLConstants.NULL\_NS\_URI](http://docs.google.com/javax/xml/XMLConstants.html#NULL_NS_URI). This value represents no explicitly defined Namespace as defined by the [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/#ns-qualnames) specification. This action preserves compatible behavior with QName 1.0. Explicitly providing the [XMLConstants.NULL\_NS\_URI](http://docs.google.com/javax/xml/XMLConstants.html#NULL_NS_URI) value is the preferred coding style.

If the local part is null an IllegalArgumentException is thrown. A local part of "" is allowed to preserve compatible behavior with QName 1.0.

When using this constructor, the prefix is set to [XMLConstants.DEFAULT\_NS\_PREFIX](http://docs.google.com/javax/xml/XMLConstants.html#DEFAULT_NS_PREFIX).

The Namespace URI is not validated as a [URI reference](http://www.ietf.org/rfc/rfc2396.txt). The local part is not validated as a [NCName](http://www.w3.org/TR/REC-xml-names/#NT-NCName) as specified in [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/).

**Parameters:**namespaceURI - Namespace URI of the QNamelocalPart - local part of the QName **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - When localPart is null**See Also:**[QName(String namespaceURI, String localPart, String prefix)](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String,%20java.lang.String,%20java.lang.String))

### QName

public **QName**([String](http://docs.google.com/java/lang/String.html) namespaceURI,  
 [String](http://docs.google.com/java/lang/String.html) localPart,  
 [String](http://docs.google.com/java/lang/String.html) prefix)

QName constructor specifying the Namespace URI, local part and prefix.

If the Namespace URI is null, it is set to [XMLConstants.NULL\_NS\_URI](http://docs.google.com/javax/xml/XMLConstants.html#NULL_NS_URI). This value represents no explicitly defined Namespace as defined by the [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/#ns-qualnames) specification. This action preserves compatible behavior with QName 1.0. Explicitly providing the [XMLConstants.NULL\_NS\_URI](http://docs.google.com/javax/xml/XMLConstants.html#NULL_NS_URI) value is the preferred coding style.

If the local part is null an IllegalArgumentException is thrown. A local part of "" is allowed to preserve compatible behavior with QName 1.0.

If the prefix is null, an IllegalArgumentException is thrown. Use [XMLConstants.DEFAULT\_NS\_PREFIX](http://docs.google.com/javax/xml/XMLConstants.html#DEFAULT_NS_PREFIX) to explicitly indicate that no prefix is present or the prefix is not relevant.

The Namespace URI is not validated as a [URI reference](http://www.ietf.org/rfc/rfc2396.txt). The local part and prefix are not validated as a [NCName](http://www.w3.org/TR/REC-xml-names/#NT-NCName) as specified in [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/).

**Parameters:**namespaceURI - Namespace URI of the QNamelocalPart - local part of the QNameprefix - prefix of the QName **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - When localPart or prefix is null

### QName

public **QName**([String](http://docs.google.com/java/lang/String.html) localPart)

QName constructor specifying the local part.

If the local part is null an IllegalArgumentException is thrown. A local part of "" is allowed to preserve compatible behavior with QName 1.0.

When using this constructor, the Namespace URI is set to [XMLConstants.NULL\_NS\_URI](http://docs.google.com/javax/xml/XMLConstants.html#NULL_NS_URI) and the prefix is set to [XMLConstants.DEFAULT\_NS\_PREFIX](http://docs.google.com/javax/xml/XMLConstants.html#DEFAULT_NS_PREFIX).

*In an XML context, all Element and Attribute names exist in the context of a Namespace. Making this explicit during the construction of a QName helps prevent hard to diagnosis XML validity errors. The constructors* [*QName(String namespaceURI, String localPart)*](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String,%20java.lang.String)) *and* [*QName(String namespaceURI, String localPart, String prefix)*](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String,%20java.lang.String,%20java.lang.String)) *are preferred.*

The local part is not validated as a [NCName](http://www.w3.org/TR/REC-xml-names/#NT-NCName) as specified in [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/).

**Parameters:**localPart - local part of the QName **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - When localPart is null**See Also:**[QName(String namespaceURI, String localPart)](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String,%20java.lang.String)), [QName(String namespaceURI, String localPart, String prefix)](http://docs.google.com/javax/xml/namespace/QName.html#QName(java.lang.String,%20java.lang.String,%20java.lang.String))

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

### getNamespaceURI

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

Get the Namespace URI of this QName.

**Returns:**Namespace URI of this QName

### getLocalPart

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

Get the local part of this QName.

**Returns:**local part of this QName

### getPrefix

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

Get the prefix of this QName.

The prefix assigned to a QName might ***NOT*** be valid in a different context. For example, a QName may be assigned a prefix in the context of parsing a document but that prefix may be invalid in the context of a different document.

**Returns:**prefix of this QName

### equals

public final boolean **equals**([Object](http://docs.google.com/java/lang/Object.html) objectToTest)

Test this QName for equality with another Object.

If the Object to be tested is not a QName or is null, then this method returns false.

Two QNames are considered equal if and only if both the Namespace URI and local part are equal. This method uses String.equals() to check equality of the Namespace URI and local part. The prefix is ***NOT*** used to determine equality.

This method satisfies the general contract of [Object.equals(Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object))

**Overrides:**[equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) in class [Object](http://docs.google.com/java/lang/Object.html) **Parameters:**objectToTest - the Object to test for equality with this QName **Returns:**true if the given Object is equal to this QName else false**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### hashCode

public final int **hashCode**()

Generate the hash code for this QName.

The hash code is calculated using both the Namespace URI and the local part of the QName. The prefix is ***NOT*** used to calculate the hash code.

This method satisfies the general contract of [Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()).

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**hash code for this QName Object**See Also:**[Object.equals(java.lang.Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### toString

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

String representation of this QName.

The commonly accepted way of representing a QName as a String was [defined](http://jclark.com/xml/xmlns.htm) by James Clark. Although this is not a *standard* specification, it is in common use, e.g. [Transformer.setParameter(String name, Object value)](http://docs.google.com/javax/xml/transform/Transformer.html#setParameter(java.lang.String,%20java.lang.Object)). This implementation represents a QName as: "{" + Namespace URI + "}" + local part. If the Namespace URI .equals(XMLConstants.NULL\_NS\_URI), only the local part is returned. An appropriate use of this method is for debugging or logging for human consumption.

Note the prefix value is ***NOT*** returned as part of the String representation.

This method satisfies the general contract of [Object.toString()](http://docs.google.com/java/lang/Object.html#toString()).

**Overrides:**[toString](http://docs.google.com/java/lang/Object.html#toString()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**String representation of this QName

### valueOf

public static [QName](http://docs.google.com/javax/xml/namespace/QName.html) **valueOf**([String](http://docs.google.com/java/lang/String.html) qNameAsString)

QName derived from parsing the formatted String.

If the String is null or does not conform to [QName.toString()](http://docs.google.com/javax/xml/namespace/QName.html#toString()) formatting, an IllegalArgumentException is thrown.

*The String* ***MUST*** *be in the form returned by* [*QName.toString()*](http://docs.google.com/javax/xml/namespace/QName.html#toString())*.*

The commonly accepted way of representing a QName as a String was [defined](http://jclark.com/xml/xmlns.htm) by James Clark. Although this is not a *standard* specification, it is in common use, e.g. [Transformer.setParameter(String name, Object value)](http://docs.google.com/javax/xml/transform/Transformer.html#setParameter(java.lang.String,%20java.lang.Object)). This implementation parses a String formatted as: "{" + Namespace URI + "}" + local part. If the Namespace URI .equals(XMLConstants.NULL\_NS\_URI), only the local part should be provided.

The prefix value ***CANNOT*** be represented in the String and will be set to [XMLConstants.DEFAULT\_NS\_PREFIX](http://docs.google.com/javax/xml/XMLConstants.html#DEFAULT_NS_PREFIX).

This method does not do full validation of the resulting QName.

The Namespace URI is not validated as a [URI reference](http://www.ietf.org/rfc/rfc2396.txt). The local part is not validated as a [NCName](http://www.w3.org/TR/REC-xml-names/#NT-NCName) as specified in [Namespaces in XML](http://www.w3.org/TR/REC-xml-names/).

**Parameters:**qNameAsString - String representation of the QName **Returns:**QName corresponding to the given String **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - When qNameAsString is null or malformed**See Also:**[QName.toString()](http://docs.google.com/javax/xml/namespace/QName.html#toString())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/QName.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/namespace/NamespaceContext.html)   NEXT CLASS | [**FRAMES**](http://docs.google.com/index.html?javax/xml/namespace/QName.html)    [**NO FRAMES**](http://docs.google.com/QName.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#17dp8vu) |

[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).