| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NamedNodeMap.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/org/w3c/dom/EntityReference.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/NameList.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/NamedNodeMap.html)    [**NO FRAMES**](http://docs.google.com/NamedNodeMap.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **org.w3c.dom**

Interface NamedNodeMap

public interface **NamedNodeMap**

Objects implementing the NamedNodeMap interface are used to represent collections of nodes that can be accessed by name. Note that NamedNodeMap does not inherit from NodeList; NamedNodeMaps are not maintained in any particular order. Objects contained in an object implementing NamedNodeMap may also be accessed by an ordinal index, but this is simply to allow convenient enumeration of the contents of a NamedNodeMap, and does not imply that the DOM specifies an order to these Nodes.

NamedNodeMap objects in the DOM are live.

See also the [Document Object Model (DOM) Level 3 Core Specification](http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407).

| **Method Summary** | |
| --- | --- |
| int | [**getLength**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#getLength())()            The number of nodes in this map. |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**getNamedItem**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#getNamedItem(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Retrieves a node specified by name. |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**getNamedItemNS**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#getNamedItemNS(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) localName)            Retrieves a node specified by local name and namespace URI. |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**item**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#item(int))(int index)            Returns the indexth item in the map. |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**removeNamedItem**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#removeNamedItem(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Removes a node specified by name. |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**removeNamedItemNS**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#removeNamedItemNS(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) localName)            Removes a node specified by local name and namespace URI. |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**setNamedItem**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#setNamedItem(org.w3c.dom.Node))([Node](http://docs.google.com/org/w3c/dom/Node.html) arg)            Adds a node using its nodeName attribute. |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**setNamedItemNS**](http://docs.google.com/org/w3c/dom/NamedNodeMap.html#setNamedItemNS(org.w3c.dom.Node))([Node](http://docs.google.com/org/w3c/dom/Node.html) arg)            Adds a node using its namespaceURI and localName. |

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

### getNamedItem

[Node](http://docs.google.com/org/w3c/dom/Node.html) **getNamedItem**([String](http://docs.google.com/java/lang/String.html) name)

Retrieves a node specified by name.

**Parameters:**name - The nodeName of a node to retrieve. **Returns:**A Node (of any type) with the specified nodeName, or null if it does not identify any node in this map.

### setNamedItem

[Node](http://docs.google.com/org/w3c/dom/Node.html) **setNamedItem**([Node](http://docs.google.com/org/w3c/dom/Node.html) arg)  
 throws [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html)

Adds a node using its nodeName attribute. If a node with that name is already present in this map, it is replaced by the new one. Replacing a node by itself has no effect.

As the nodeName attribute is used to derive the name which the node must be stored under, multiple nodes of certain types (those that have a "special" string value) cannot be stored as the names would clash. This is seen as preferable to allowing nodes to be aliased.

**Parameters:**arg - A node to store in this map. The node will later be accessible using the value of its nodeName attribute. **Returns:**If the new Node replaces an existing node the replaced Node is returned, otherwise null is returned. **Throws:** [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html) - WRONG\_DOCUMENT\_ERR: Raised if arg was created from a different document than the one that created this map.

NO\_MODIFICATION\_ALLOWED\_ERR: Raised if this map is readonly.

INUSE\_ATTRIBUTE\_ERR: Raised if arg is an Attr that is already an attribute of another Element object. The DOM user must explicitly clone Attr nodes to re-use them in other elements.

HIERARCHY\_REQUEST\_ERR: Raised if an attempt is made to add a node doesn't belong in this NamedNodeMap. Examples would include trying to insert something other than an Attr node into an Element's map of attributes, or a non-Entity node into the DocumentType's map of Entities.

### removeNamedItem

[Node](http://docs.google.com/org/w3c/dom/Node.html) **removeNamedItem**([String](http://docs.google.com/java/lang/String.html) name)  
 throws [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html)

Removes a node specified by name. When this map contains the attributes attached to an element, if the removed attribute is known to have a default value, an attribute immediately appears containing the default value as well as the corresponding namespace URI, local name, and prefix when applicable.

**Parameters:**name - The nodeName of the node to remove. **Returns:**The node removed from this map if a node with such a name exists. **Throws:** [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html) - NOT\_FOUND\_ERR: Raised if there is no node named name in this map.

NO\_MODIFICATION\_ALLOWED\_ERR: Raised if this map is readonly.

### item

[Node](http://docs.google.com/org/w3c/dom/Node.html) **item**(int index)

Returns the indexth item in the map. If index is greater than or equal to the number of nodes in this map, this returns null.

**Parameters:**index - Index into this map. **Returns:**The node at the indexth position in the map, or null if that is not a valid index.

### getLength

int **getLength**()

The number of nodes in this map. The range of valid child node indices is 0 to length-1 inclusive.

### getNamedItemNS

[Node](http://docs.google.com/org/w3c/dom/Node.html) **getNamedItemNS**([String](http://docs.google.com/java/lang/String.html) namespaceURI,  
 [String](http://docs.google.com/java/lang/String.html) localName)  
 throws [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html)

Retrieves a node specified by local name and namespace URI.

Per [[XML Namespaces](http://www.w3.org/TR/1999/REC-xml-names-19990114/)] , applications must use the value null as the namespaceURI parameter for methods if they wish to have no namespace.

**Parameters:**namespaceURI - The namespace URI of the node to retrieve.localName - The local name of the node to retrieve. **Returns:**A Node (of any type) with the specified local name and namespace URI, or null if they do not identify any node in this map. **Throws:** [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html) - NOT\_SUPPORTED\_ERR: May be raised if the implementation does not support the feature "XML" and the language exposed through the Document does not support XML Namespaces (such as [[HTML 4.01](http://www.w3.org/TR/1999/REC-html401-19991224/)]).**Since:** DOM Level 2

### setNamedItemNS

[Node](http://docs.google.com/org/w3c/dom/Node.html) **setNamedItemNS**([Node](http://docs.google.com/org/w3c/dom/Node.html) arg)  
 throws [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html)

Adds a node using its namespaceURI and localName. If a node with that namespace URI and that local name is already present in this map, it is replaced by the new one. Replacing a node by itself has no effect.

Per [[XML Namespaces](http://www.w3.org/TR/1999/REC-xml-names-19990114/)] , applications must use the value null as the namespaceURI parameter for methods if they wish to have no namespace.

**Parameters:**arg - A node to store in this map. The node will later be accessible using the value of its namespaceURI and localName attributes. **Returns:**If the new Node replaces an existing node the replaced Node is returned, otherwise null is returned. **Throws:** [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html) - WRONG\_DOCUMENT\_ERR: Raised if arg was created from a different document than the one that created this map.

NO\_MODIFICATION\_ALLOWED\_ERR: Raised if this map is readonly.

INUSE\_ATTRIBUTE\_ERR: Raised if arg is an Attr that is already an attribute of another Element object. The DOM user must explicitly clone Attr nodes to re-use them in other elements.

HIERARCHY\_REQUEST\_ERR: Raised if an attempt is made to add a node doesn't belong in this NamedNodeMap. Examples would include trying to insert something other than an Attr node into an Element's map of attributes, or a non-Entity node into the DocumentType's map of Entities.

NOT\_SUPPORTED\_ERR: May be raised if the implementation does not support the feature "XML" and the language exposed through the Document does not support XML Namespaces (such as [[HTML 4.01](http://www.w3.org/TR/1999/REC-html401-19991224/)]).**Since:** DOM Level 2

### removeNamedItemNS

[Node](http://docs.google.com/org/w3c/dom/Node.html) **removeNamedItemNS**([String](http://docs.google.com/java/lang/String.html) namespaceURI,  
 [String](http://docs.google.com/java/lang/String.html) localName)  
 throws [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html)

Removes a node specified by local name and namespace URI. A removed attribute may be known to have a default value when this map contains the attributes attached to an element, as returned by the attributes attribute of the Node interface. If so, an attribute immediately appears containing the default value as well as the corresponding namespace URI, local name, and prefix when applicable.

Per [[XML Namespaces](http://www.w3.org/TR/1999/REC-xml-names-19990114/)] , applications must use the value null as the namespaceURI parameter for methods if they wish to have no namespace.

**Parameters:**namespaceURI - The namespace URI of the node to remove.localName - The local name of the node to remove. **Returns:**The node removed from this map if a node with such a local name and namespace URI exists. **Throws:** [DOMException](http://docs.google.com/org/w3c/dom/DOMException.html) - NOT\_FOUND\_ERR: Raised if there is no node with the specified namespaceURI and localName in this map.

NO\_MODIFICATION\_ALLOWED\_ERR: Raised if this map is readonly.

NOT\_SUPPORTED\_ERR: May be raised if the implementation does not support the feature "XML" and the language exposed through the Document does not support XML Namespaces (such as [[HTML 4.01](http://www.w3.org/TR/1999/REC-html401-19991224/)]).**Since:** DOM Level 2

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NamedNodeMap.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/org/w3c/dom/EntityReference.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/NameList.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/NamedNodeMap.html)    [**NO FRAMES**](http://docs.google.com/NamedNodeMap.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).