| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NameClassPair.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/naming/NameAlreadyBoundException.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/NameNotFoundException.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/NameClassPair.html)    [**NO FRAMES**](http://docs.google.com/NameClassPair.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#2s8eyo1) |

## **javax.naming**

Class NameClassPair

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.naming.NameClassPair**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html) **Direct Known Subclasses:** [Binding](http://docs.google.com/javax/naming/Binding.html)

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

This class represents the object name and class name pair of a binding found in a context.

A context consists of name-to-object bindings. The NameClassPair class represents the name and the class of the bound object. It consists of a name and a string representing the package-qualified class name.

Use subclassing for naming systems that generate contents of a name/class pair dynamically.

A NameClassPair instance is not synchronized against concurrent access by multiple threads. Threads that need to access a NameClassPair concurrently should synchronize amongst themselves and provide the necessary locking.

**Since:** 1.3 **See Also:**[Context.list(javax.naming.Name)](http://docs.google.com/javax/naming/Context.html#list(javax.naming.Name)), [Serialized Form](http://docs.google.com/serialized-form.html#javax.naming.NameClassPair)

| **Constructor Summary** | |
| --- | --- |
| [**NameClassPair**](http://docs.google.com/javax/naming/NameClassPair.html#NameClassPair(java.lang.String,%20java.lang.String))([String](http://docs.google.com/java/lang/String.html) name, [String](http://docs.google.com/java/lang/String.html) className)            Constructs an instance of a NameClassPair given its name and class name. |
| [**NameClassPair**](http://docs.google.com/javax/naming/NameClassPair.html#NameClassPair(java.lang.String,%20java.lang.String,%20boolean))([String](http://docs.google.com/java/lang/String.html) name, [String](http://docs.google.com/java/lang/String.html) className, boolean isRelative)            Constructs an instance of a NameClassPair given its name, class name, and whether it is relative to the listing context. |

| **Method Summary** | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | [**getClassName**](http://docs.google.com/javax/naming/NameClassPair.html#getClassName())()            Retrieves the class name of the object bound to the name of this binding. |
| [String](http://docs.google.com/java/lang/String.html) | [**getName**](http://docs.google.com/javax/naming/NameClassPair.html#getName())()            Retrieves the name of this binding. |
| [String](http://docs.google.com/java/lang/String.html) | [**getNameInNamespace**](http://docs.google.com/javax/naming/NameClassPair.html#getNameInNamespace())()            Retrieves the full name of this binding. |
| boolean | [**isRelative**](http://docs.google.com/javax/naming/NameClassPair.html#isRelative())()            Determines whether the name of this binding is relative to the target context (which is named by the first parameter of the list() method). |
| void | [**setClassName**](http://docs.google.com/javax/naming/NameClassPair.html#setClassName(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Sets the class name of this binding. |
| void | [**setName**](http://docs.google.com/javax/naming/NameClassPair.html#setName(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Sets the name of this binding. |
| void | [**setNameInNamespace**](http://docs.google.com/javax/naming/NameClassPair.html#setNameInNamespace(java.lang.String))([String](http://docs.google.com/java/lang/String.html) fullName)            Sets the full name of this binding. |
| void | [**setRelative**](http://docs.google.com/javax/naming/NameClassPair.html#setRelative(boolean))(boolean r)            Sets whether the name of this binding is relative to the target context (which is named by the first parameter of the list() method). |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/naming/NameClassPair.html#toString())()            Generates the string representation of this name/class pair. |

| **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()), [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** |
| --- |

### NameClassPair

public **NameClassPair**([String](http://docs.google.com/java/lang/String.html) name,  
 [String](http://docs.google.com/java/lang/String.html) className)

Constructs an instance of a NameClassPair given its name and class name.

**Parameters:**name - The non-null name of the object. It is relative to the *target context* (which is named by the first parameter of the list() method)className - The possibly null class name of the object bound to name. It is null if the object bound is null.**See Also:**[getClassName()](http://docs.google.com/javax/naming/NameClassPair.html#getClassName()), [setClassName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setClassName(java.lang.String)), [getName()](http://docs.google.com/javax/naming/NameClassPair.html#getName()), [setName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setName(java.lang.String))

### NameClassPair

public **NameClassPair**([String](http://docs.google.com/java/lang/String.html) name,  
 [String](http://docs.google.com/java/lang/String.html) className,  
 boolean isRelative)

Constructs an instance of a NameClassPair given its name, class name, and whether it is relative to the listing context.

**Parameters:**name - The non-null name of the object.className - The possibly null class name of the object bound to name. It is null if the object bound is null.isRelative - true if name is a name relative to the target context (which is named by the first parameter of the list() method); false if name is a URL string.**See Also:**[getClassName()](http://docs.google.com/javax/naming/NameClassPair.html#getClassName()), [setClassName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setClassName(java.lang.String)), [getName()](http://docs.google.com/javax/naming/NameClassPair.html#getName()), [setName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setName(java.lang.String)), [isRelative()](http://docs.google.com/javax/naming/NameClassPair.html#isRelative()), [setRelative(boolean)](http://docs.google.com/javax/naming/NameClassPair.html#setRelative(boolean))

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

### getClassName

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

Retrieves the class name of the object bound to the name of this binding. If a reference or some other indirect information is bound, retrieves the class name of the eventual object that will be returned by Binding.getObject().

**Returns:**The possibly null class name of object bound. It is null if the object bound is null.**See Also:**[Binding.getObject()](http://docs.google.com/javax/naming/Binding.html#getObject()), [Binding.getClassName()](http://docs.google.com/javax/naming/Binding.html#getClassName()), [setClassName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setClassName(java.lang.String))

### getName

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

Retrieves the name of this binding. If isRelative() is true, this name is relative to the target context (which is named by the first parameter of the list()). If isRelative() is false, this name is a URL string.

**Returns:**The non-null name of this binding.**See Also:**[isRelative()](http://docs.google.com/javax/naming/NameClassPair.html#isRelative()), [setName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setName(java.lang.String))

### setName

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

Sets the name of this binding.

**Parameters:**name - the non-null string to use as the name.**See Also:**[getName()](http://docs.google.com/javax/naming/NameClassPair.html#getName()), [setRelative(boolean)](http://docs.google.com/javax/naming/NameClassPair.html#setRelative(boolean))

### setClassName

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

Sets the class name of this binding.

**Parameters:**name - the possibly null string to use as the class name. If null, Binding.getClassName() will return the actual class name of the object in the binding. The class name will be null if the object bound is null.**See Also:**[getClassName()](http://docs.google.com/javax/naming/NameClassPair.html#getClassName()), [Binding.getClassName()](http://docs.google.com/javax/naming/Binding.html#getClassName())

### isRelative

public boolean **isRelative**()

Determines whether the name of this binding is relative to the target context (which is named by the first parameter of the list() method).

**Returns:**true if the name of this binding is relative to the target context; false if the name of this binding is a URL string.**See Also:**[setRelative(boolean)](http://docs.google.com/javax/naming/NameClassPair.html#setRelative(boolean)), [getName()](http://docs.google.com/javax/naming/NameClassPair.html#getName())

### setRelative

public void **setRelative**(boolean r)

Sets whether the name of this binding is relative to the target context (which is named by the first parameter of the list() method).

**Parameters:**r - If true, the name of binding is relative to the target context; if false, the name of binding is a URL string.**See Also:**[isRelative()](http://docs.google.com/javax/naming/NameClassPair.html#isRelative()), [setName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setName(java.lang.String))

### getNameInNamespace

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

Retrieves the full name of this binding. The full name is the absolute name of this binding within its own namespace. See [Context.getNameInNamespace()](http://docs.google.com/javax/naming/Context.html#getNameInNamespace()).

In naming systems for which the notion of full name does not apply to this binding an UnsupportedOperationException is thrown. This exception is also thrown when a service provider written before the introduction of the method is in use.

The string returned by this method is not a JNDI composite name and should not be passed directly to context methods.

**Returns:**The full name of this binding. **Throws:** [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the notion of full name does not apply to this binding in the naming system.**Since:** 1.5 **See Also:**[setNameInNamespace(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setNameInNamespace(java.lang.String)), [getName()](http://docs.google.com/javax/naming/NameClassPair.html#getName())

### setNameInNamespace

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

Sets the full name of this binding. This method must be called to set the full name whenever a NameClassPair is created and a full name is applicable to this binding.

Setting the full name to null, or not setting it at all, will cause getNameInNamespace() to throw an exception.

**Parameters:**fullName - The full name to use.**Since:** 1.5 **See Also:**[getNameInNamespace()](http://docs.google.com/javax/naming/NameClassPair.html#getNameInNamespace()), [setName(java.lang.String)](http://docs.google.com/javax/naming/NameClassPair.html#setName(java.lang.String))

### toString

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

Generates the string representation of this name/class pair. The string representation consists of the name and class name separated by a colon (':'). The contents of this string is useful for debugging and is not meant to be interpreted programmatically.

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

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NameClassPair.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/naming/NameAlreadyBoundException.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/NameNotFoundException.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/NameClassPair.html)    [**NO FRAMES**](http://docs.google.com/NameClassPair.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#2s8eyo1) |

[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).