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

## **javax.management**

Interface MBeanServerConnection

**All Known Subinterfaces:** [MBeanServer](http://docs.google.com/javax/management/MBeanServer.html), [MBeanServerForwarder](http://docs.google.com/javax/management/remote/MBeanServerForwarder.html)

public interface **MBeanServerConnection**

This interface represents a way to talk to an MBean server, whether local or remote. The [MBeanServer](http://docs.google.com/javax/management/MBeanServer.html) interface, representing a local MBean server, extends this interface.

**Since:** 1.5

| **Method Summary** | |
| --- | --- |
| void | [**addNotificationListener**](http://docs.google.com/javax/management/MBeanServerConnection.html#addNotificationListener(javax.management.ObjectName,%20javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener, [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter, [Object](http://docs.google.com/java/lang/Object.html) handback)            Adds a listener to a registered MBean. |
| void | [**addNotificationListener**](http://docs.google.com/javax/management/MBeanServerConnection.html#addNotificationListener(javax.management.ObjectName,%20javax.management.ObjectName,%20javax.management.NotificationFilter,%20java.lang.Object))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) listener, [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter, [Object](http://docs.google.com/java/lang/Object.html) handback)            Adds a listener to a registered MBean. |
| [ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) | [**createMBean**](http://docs.google.com/javax/management/MBeanServerConnection.html#createMBean(java.lang.String,%20javax.management.ObjectName))([String](http://docs.google.com/java/lang/String.html) className, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)            Instantiates and registers an MBean in the MBean server. |
| [ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) | [**createMBean**](http://docs.google.com/javax/management/MBeanServerConnection.html#createMBean(java.lang.String,%20javax.management.ObjectName,%20java.lang.Object%5B%5D,%20java.lang.String%5B%5D))([String](http://docs.google.com/java/lang/String.html) className, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [Object](http://docs.google.com/java/lang/Object.html)[] params, [String](http://docs.google.com/java/lang/String.html)[] signature)            Instantiates and registers an MBean in the MBean server. |
| [ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) | [**createMBean**](http://docs.google.com/javax/management/MBeanServerConnection.html#createMBean(java.lang.String,%20javax.management.ObjectName,%20javax.management.ObjectName))([String](http://docs.google.com/java/lang/String.html) className, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) loaderName)            Instantiates and registers an MBean in the MBean server. |
| [ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) | [**createMBean**](http://docs.google.com/javax/management/MBeanServerConnection.html#createMBean(java.lang.String,%20javax.management.ObjectName,%20javax.management.ObjectName,%20java.lang.Object%5B%5D,%20java.lang.String%5B%5D))([String](http://docs.google.com/java/lang/String.html) className, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) loaderName, [Object](http://docs.google.com/java/lang/Object.html)[] params, [String](http://docs.google.com/java/lang/String.html)[] signature)            Instantiates and registers an MBean in the MBean server. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getAttribute**](http://docs.google.com/javax/management/MBeanServerConnection.html#getAttribute(javax.management.ObjectName,%20java.lang.String))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [String](http://docs.google.com/java/lang/String.html) attribute)            Gets the value of a specific attribute of a named MBean. |
| [AttributeList](http://docs.google.com/javax/management/AttributeList.html) | [**getAttributes**](http://docs.google.com/javax/management/MBeanServerConnection.html#getAttributes(javax.management.ObjectName,%20java.lang.String%5B%5D))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [String](http://docs.google.com/java/lang/String.html)[] attributes)            Enables the values of several attributes of a named MBean. |
| [String](http://docs.google.com/java/lang/String.html) | [**getDefaultDomain**](http://docs.google.com/javax/management/MBeanServerConnection.html#getDefaultDomain())()            Returns the default domain used for naming the MBean. |
| [String](http://docs.google.com/java/lang/String.html)[] | [**getDomains**](http://docs.google.com/javax/management/MBeanServerConnection.html#getDomains())()            Returns the list of domains in which any MBean is currently registered. |
| [Integer](http://docs.google.com/java/lang/Integer.html) | [**getMBeanCount**](http://docs.google.com/javax/management/MBeanServerConnection.html#getMBeanCount())()            Returns the number of MBeans registered in the MBean server. |
| [MBeanInfo](http://docs.google.com/javax/management/MBeanInfo.html) | [**getMBeanInfo**](http://docs.google.com/javax/management/MBeanServerConnection.html#getMBeanInfo(javax.management.ObjectName))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)            This method discovers the attributes and operations that an MBean exposes for management. |
| [ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) | [**getObjectInstance**](http://docs.google.com/javax/management/MBeanServerConnection.html#getObjectInstance(javax.management.ObjectName))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)            Gets the ObjectInstance for a given MBean registered with the MBean server. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**invoke**](http://docs.google.com/javax/management/MBeanServerConnection.html#invoke(javax.management.ObjectName,%20java.lang.String,%20java.lang.Object%5B%5D,%20java.lang.String%5B%5D))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [String](http://docs.google.com/java/lang/String.html) operationName, [Object](http://docs.google.com/java/lang/Object.html)[] params, [String](http://docs.google.com/java/lang/String.html)[] signature)            Invokes an operation on an MBean. |
| boolean | [**isInstanceOf**](http://docs.google.com/javax/management/MBeanServerConnection.html#isInstanceOf(javax.management.ObjectName,%20java.lang.String))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [String](http://docs.google.com/java/lang/String.html) className)            Returns true if the MBean specified is an instance of the specified class, false otherwise. |
| boolean | [**isRegistered**](http://docs.google.com/javax/management/MBeanServerConnection.html#isRegistered(javax.management.ObjectName))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)            Checks whether an MBean, identified by its object name, is already registered with the MBean server. |
| [Set](http://docs.google.com/java/util/Set.html)<[ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html)> | [**queryMBeans**](http://docs.google.com/javax/management/MBeanServerConnection.html#queryMBeans(javax.management.ObjectName,%20javax.management.QueryExp))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [QueryExp](http://docs.google.com/javax/management/QueryExp.html) query)            Gets MBeans controlled by the MBean server. |
| [Set](http://docs.google.com/java/util/Set.html)<[ObjectName](http://docs.google.com/javax/management/ObjectName.html)> | [**queryNames**](http://docs.google.com/javax/management/MBeanServerConnection.html#queryNames(javax.management.ObjectName,%20javax.management.QueryExp))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [QueryExp](http://docs.google.com/javax/management/QueryExp.html) query)            Gets the names of MBeans controlled by the MBean server. |
| void | [**removeNotificationListener**](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.NotificationListener))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener)            Removes a listener from a registered MBean. |
| void | [**removeNotificationListener**](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener, [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter, [Object](http://docs.google.com/java/lang/Object.html) handback)            Removes a listener from a registered MBean. |
| void | [**removeNotificationListener**](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.ObjectName))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) listener)            Removes a listener from a registered MBean. |
| void | [**removeNotificationListener**](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.ObjectName,%20javax.management.NotificationFilter,%20java.lang.Object))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) listener, [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter, [Object](http://docs.google.com/java/lang/Object.html) handback)            Removes a listener from a registered MBean. |
| void | [**setAttribute**](http://docs.google.com/javax/management/MBeanServerConnection.html#setAttribute(javax.management.ObjectName,%20javax.management.Attribute))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [Attribute](http://docs.google.com/javax/management/Attribute.html) attribute)            Sets the value of a specific attribute of a named MBean. |
| [AttributeList](http://docs.google.com/javax/management/AttributeList.html) | [**setAttributes**](http://docs.google.com/javax/management/MBeanServerConnection.html#setAttributes(javax.management.ObjectName,%20javax.management.AttributeList))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name, [AttributeList](http://docs.google.com/javax/management/AttributeList.html) attributes)            Sets the values of several attributes of a named MBean. |
| void | [**unregisterMBean**](http://docs.google.com/javax/management/MBeanServerConnection.html#unregisterMBean(javax.management.ObjectName))([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)            Unregisters an MBean from the MBean server. |

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

### createMBean

[ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) **createMBean**([String](http://docs.google.com/java/lang/String.html) className,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)  
 throws [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html),  
 [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html),  
 [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Instantiates and registers an MBean in the MBean server. The MBean server will use its [Default Loader Repository](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html) to load the class of the MBean. An object name is associated to the MBean. If the object name given is null, the MBean must provide its own name by implementing the [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html) interface and returning the name from the [preRegister](http://docs.google.com/javax/management/MBeanRegistration.html#preRegister(javax.management.MBeanServer,%20javax.management.ObjectName)) method.

This method is equivalent to [createMBean(className, name, (Object[]) null, (String[]) null)](http://docs.google.com/javax/management/MBeanServerConnection.html#createMBean(java.lang.String,%20javax.management.ObjectName,%20java.lang.Object%5B%5D,%20java.lang.String%5B%5D)).

**Parameters:**className - The class name of the MBean to be instantiated.name - The object name of the MBean. May be null. **Returns:**An ObjectInstance, containing the ObjectName and the Java class name of the newly instantiated MBean. If the contained ObjectName is n, the contained Java class name is [getMBeanInfo(n)](http://docs.google.com/javax/management/MBeanServerConnection.html#getMBeanInfo(javax.management.ObjectName)).getClassName(). **Throws:** [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - Wraps a java.lang.ClassNotFoundException or a java.lang.Exception that occurred when trying to invoke the MBean's constructor. [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html) - The MBean is already under the control of the MBean server. [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html) - The preRegister (MBeanRegistration interface) method of the MBean has thrown an exception. The MBean will not be registered. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - The constructor of the MBean has thrown an exception [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html) - This class is not a JMX compliant MBean [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The className passed in parameter is null, the ObjectName passed in parameter contains a pattern or no ObjectName is specified for the MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### createMBean

[ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) **createMBean**([String](http://docs.google.com/java/lang/String.html) className,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) loaderName)  
 throws [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html),  
 [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html),  
 [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html),  
 [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Instantiates and registers an MBean in the MBean server. The class loader to be used is identified by its object name. An object name is associated to the MBean. If the object name of the loader is null, the ClassLoader that loaded the MBean server will be used. If the MBean's object name given is null, the MBean must provide its own name by implementing the [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html) interface and returning the name from the [preRegister](http://docs.google.com/javax/management/MBeanRegistration.html#preRegister(javax.management.MBeanServer,%20javax.management.ObjectName)) method.

This method is equivalent to [createMBean(className, name, loaderName, (Object[]) null, (String[]) null)](http://docs.google.com/javax/management/MBeanServerConnection.html#createMBean(java.lang.String,%20javax.management.ObjectName,%20javax.management.ObjectName,%20java.lang.Object%5B%5D,%20java.lang.String%5B%5D)).

**Parameters:**className - The class name of the MBean to be instantiated.name - The object name of the MBean. May be null.loaderName - The object name of the class loader to be used. **Returns:**An ObjectInstance, containing the ObjectName and the Java class name of the newly instantiated MBean. If the contained ObjectName is n, the contained Java class name is [getMBeanInfo(n)](http://docs.google.com/javax/management/MBeanServerConnection.html#getMBeanInfo(javax.management.ObjectName)).getClassName(). **Throws:** [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - Wraps a java.lang.ClassNotFoundException or a java.lang.Exception that occurred when trying to invoke the MBean's constructor. [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html) - The MBean is already under the control of the MBean server. [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html) - The preRegister (MBeanRegistration interface) method of the MBean has thrown an exception. The MBean will not be registered. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - The constructor of the MBean has thrown an exception [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html) - This class is not a JMX compliant MBean [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The specified class loader is not registered in the MBean server. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The className passed in parameter is null, the ObjectName passed in parameter contains a pattern or no ObjectName is specified for the MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### createMBean

[ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) **createMBean**([String](http://docs.google.com/java/lang/String.html) className,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [Object](http://docs.google.com/java/lang/Object.html)[] params,  
 [String](http://docs.google.com/java/lang/String.html)[] signature)  
 throws [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html),  
 [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html),  
 [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Instantiates and registers an MBean in the MBean server. The MBean server will use its [Default Loader Repository](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html) to load the class of the MBean. An object name is associated to the MBean. If the object name given is null, the MBean must provide its own name by implementing the [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html) interface and returning the name from the [preRegister](http://docs.google.com/javax/management/MBeanRegistration.html#preRegister(javax.management.MBeanServer,%20javax.management.ObjectName)) method.

**Parameters:**className - The class name of the MBean to be instantiated.name - The object name of the MBean. May be null.params - An array containing the parameters of the constructor to be invoked.signature - An array containing the signature of the constructor to be invoked. **Returns:**An ObjectInstance, containing the ObjectName and the Java class name of the newly instantiated MBean. If the contained ObjectName is n, the contained Java class name is [getMBeanInfo(n)](http://docs.google.com/javax/management/MBeanServerConnection.html#getMBeanInfo(javax.management.ObjectName)).getClassName(). **Throws:** [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - Wraps a java.lang.ClassNotFoundException or a java.lang.Exception that occurred when trying to invoke the MBean's constructor. [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html) - The MBean is already under the control of the MBean server. [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html) - The preRegister (MBeanRegistration interface) method of the MBean has thrown an exception. The MBean will not be registered. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - The constructor of the MBean has thrown an exception [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html) - This class is not a JMX compliant MBean [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The className passed in parameter is null, the ObjectName passed in parameter contains a pattern or no ObjectName is specified for the MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### createMBean

[ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) **createMBean**([String](http://docs.google.com/java/lang/String.html) className,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) loaderName,  
 [Object](http://docs.google.com/java/lang/Object.html)[] params,  
 [String](http://docs.google.com/java/lang/String.html)[] signature)  
 throws [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html),  
 [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html),  
 [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html),  
 [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Instantiates and registers an MBean in the MBean server. The class loader to be used is identified by its object name. An object name is associated to the MBean. If the object name of the loader is not specified, the ClassLoader that loaded the MBean server will be used. If the MBean object name given is null, the MBean must provide its own name by implementing the [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html) interface and returning the name from the [preRegister](http://docs.google.com/javax/management/MBeanRegistration.html#preRegister(javax.management.MBeanServer,%20javax.management.ObjectName)) method.

**Parameters:**className - The class name of the MBean to be instantiated.name - The object name of the MBean. May be null.params - An array containing the parameters of the constructor to be invoked.signature - An array containing the signature of the constructor to be invoked.loaderName - The object name of the class loader to be used. **Returns:**An ObjectInstance, containing the ObjectName and the Java class name of the newly instantiated MBean. If the contained ObjectName is n, the contained Java class name is [getMBeanInfo(n)](http://docs.google.com/javax/management/MBeanServerConnection.html#getMBeanInfo(javax.management.ObjectName)).getClassName(). **Throws:** [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - Wraps a java.lang.ClassNotFoundException or a java.lang.Exception that occurred when trying to invoke the MBean's constructor. [InstanceAlreadyExistsException](http://docs.google.com/javax/management/InstanceAlreadyExistsException.html) - The MBean is already under the control of the MBean server. [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html) - The preRegister (MBeanRegistration interface) method of the MBean has thrown an exception. The MBean will not be registered. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - The constructor of the MBean has thrown an exception [NotCompliantMBeanException](http://docs.google.com/javax/management/NotCompliantMBeanException.html) - This class is not a JMX compliant MBean [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The specified class loader is not registered in the MBean server. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The className passed in parameter is null, the ObjectName passed in parameter contains a pattern or no ObjectName is specified for the MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### unregisterMBean

void **unregisterMBean**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Unregisters an MBean from the MBean server. The MBean is identified by its object name. Once the method has been invoked, the MBean may no longer be accessed by its object name.

**Parameters:**name - The object name of the MBean to be unregistered. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html) - The preDeregister ((MBeanRegistration interface) method of the MBean has thrown an exception. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The object name in parameter is null or the MBean you are when trying to unregister is the [MBeanServerDelegate](http://docs.google.com/javax/management/MBeanServerDelegate.html) MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### getObjectInstance

[ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html) **getObjectInstance**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Gets the ObjectInstance for a given MBean registered with the MBean server.

**Parameters:**name - The object name of the MBean. **Returns:**The ObjectInstance associated with the MBean specified by name. The contained ObjectName is name and the contained class name is [getMBeanInfo(name)](http://docs.google.com/javax/management/MBeanServerConnection.html#getMBeanInfo(javax.management.ObjectName)).getClassName(). **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### queryMBeans

[Set](http://docs.google.com/java/util/Set.html)<[ObjectInstance](http://docs.google.com/javax/management/ObjectInstance.html)> **queryMBeans**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [QueryExp](http://docs.google.com/javax/management/QueryExp.html) query)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Gets MBeans controlled by the MBean server. This method allows any of the following to be obtained: All MBeans, a set of MBeans specified by pattern matching on the ObjectName and/or a Query expression, a specific MBean. When the object name is null or no domain and key properties are specified, all objects are to be selected (and filtered if a query is specified). It returns the set of ObjectInstance objects (containing the ObjectName and the Java Class name) for the selected MBeans.

**Parameters:**name - The object name pattern identifying the MBeans to be retrieved. If null or no domain and key properties are specified, all the MBeans registered will be retrieved.query - The query expression to be applied for selecting MBeans. If null no query expression will be applied for selecting MBeans. **Returns:**A set containing the ObjectInstance objects for the selected MBeans. If no MBean satisfies the query an empty list is returned. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### queryNames

[Set](http://docs.google.com/java/util/Set.html)<[ObjectName](http://docs.google.com/javax/management/ObjectName.html)> **queryNames**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [QueryExp](http://docs.google.com/javax/management/QueryExp.html) query)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Gets the names of MBeans controlled by the MBean server. This method enables any of the following to be obtained: The names of all MBeans, the names of a set of MBeans specified by pattern matching on the ObjectName and/or a Query expression, a specific MBean name (equivalent to testing whether an MBean is registered). When the object name is null or no domain and key properties are specified, all objects are selected (and filtered if a query is specified). It returns the set of ObjectNames for the MBeans selected.

**Parameters:**name - The object name pattern identifying the MBean names to be retrieved. If null or no domain and key properties are specified, the name of all registered MBeans will be retrieved.query - The query expression to be applied for selecting MBeans. If null no query expression will be applied for selecting MBeans. **Returns:**A set containing the ObjectNames for the MBeans selected. If no MBean satisfies the query, an empty list is returned. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### isRegistered

boolean **isRegistered**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Checks whether an MBean, identified by its object name, is already registered with the MBean server.

**Parameters:**name - The object name of the MBean to be checked. **Returns:**True if the MBean is already registered in the MBean server, false otherwise. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The object name in parameter is null. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### getMBeanCount

[Integer](http://docs.google.com/java/lang/Integer.html) **getMBeanCount**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Returns the number of MBeans registered in the MBean server.

**Returns:**the number of MBeans registered. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### getAttribute

[Object](http://docs.google.com/java/lang/Object.html) **getAttribute**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [String](http://docs.google.com/java/lang/String.html) attribute)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [AttributeNotFoundException](http://docs.google.com/javax/management/AttributeNotFoundException.html),  
 [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Gets the value of a specific attribute of a named MBean. The MBean is identified by its object name.

**Parameters:**name - The object name of the MBean from which the attribute is to be retrieved.attribute - A String specifying the name of the attribute to be retrieved. **Returns:**The value of the retrieved attribute. **Throws:** [AttributeNotFoundException](http://docs.google.com/javax/management/AttributeNotFoundException.html) - The attribute specified is not accessible in the MBean. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps an exception thrown by the MBean's getter. [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - Wraps a java.lang.Exception thrown when trying to invoke the setter. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The object name in parameter is null or the attribute in parameter is null. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[setAttribute(javax.management.ObjectName, javax.management.Attribute)](http://docs.google.com/javax/management/MBeanServerConnection.html#setAttribute(javax.management.ObjectName,%20javax.management.Attribute))

### getAttributes

[AttributeList](http://docs.google.com/javax/management/AttributeList.html) **getAttributes**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [String](http://docs.google.com/java/lang/String.html)[] attributes)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Enables the values of several attributes of a named MBean. The MBean is identified by its object name.

**Parameters:**name - The object name of the MBean from which the attributes are retrieved.attributes - A list of the attributes to be retrieved. **Returns:**The list of the retrieved attributes. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - An exception occurred when trying to invoke the getAttributes method of a Dynamic MBean. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wrap a java.lang.IllegalArgumentException: The object name in parameter is null or attributes in parameter is null. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[setAttributes(javax.management.ObjectName, javax.management.AttributeList)](http://docs.google.com/javax/management/MBeanServerConnection.html#setAttributes(javax.management.ObjectName,%20javax.management.AttributeList))

### setAttribute

void **setAttribute**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [Attribute](http://docs.google.com/javax/management/Attribute.html) attribute)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [AttributeNotFoundException](http://docs.google.com/javax/management/AttributeNotFoundException.html),  
 [InvalidAttributeValueException](http://docs.google.com/javax/management/InvalidAttributeValueException.html),  
 [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Sets the value of a specific attribute of a named MBean. The MBean is identified by its object name.

**Parameters:**name - The name of the MBean within which the attribute is to be set.attribute - The identification of the attribute to be set and the value it is to be set to. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [AttributeNotFoundException](http://docs.google.com/javax/management/AttributeNotFoundException.html) - The attribute specified is not accessible in the MBean. [InvalidAttributeValueException](http://docs.google.com/javax/management/InvalidAttributeValueException.html) - The value specified for the attribute is not valid. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps an exception thrown by the MBean's setter. [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - Wraps a java.lang.Exception thrown when trying to invoke the setter. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The object name in parameter is null or the attribute in parameter is null. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[getAttribute(javax.management.ObjectName, java.lang.String)](http://docs.google.com/javax/management/MBeanServerConnection.html#getAttribute(javax.management.ObjectName,%20java.lang.String))

### setAttributes

[AttributeList](http://docs.google.com/javax/management/AttributeList.html) **setAttributes**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [AttributeList](http://docs.google.com/javax/management/AttributeList.html) attributes)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Sets the values of several attributes of a named MBean. The MBean is identified by its object name.

**Parameters:**name - The object name of the MBean within which the attributes are to be set.attributes - A list of attributes: The identification of the attributes to be set and the values they are to be set to. **Returns:**The list of attributes that were set, with their new values. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - An exception occurred when trying to invoke the getAttributes method of a Dynamic MBean. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps a java.lang.IllegalArgumentException: The object name in parameter is null or attributes in parameter is null. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[getAttributes(javax.management.ObjectName, java.lang.String[])](http://docs.google.com/javax/management/MBeanServerConnection.html#getAttributes(javax.management.ObjectName,%20java.lang.String%5B%5D))

### invoke

[Object](http://docs.google.com/java/lang/Object.html) **invoke**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [String](http://docs.google.com/java/lang/String.html) operationName,  
 [Object](http://docs.google.com/java/lang/Object.html)[] params,  
 [String](http://docs.google.com/java/lang/String.html)[] signature)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Invokes an operation on an MBean.

**Parameters:**name - The object name of the MBean on which the method is to be invoked.operationName - The name of the operation to be invoked.params - An array containing the parameters to be set when the operation is invokedsignature - An array containing the signature of the operation. The class objects will be loaded using the same class loader as the one used for loading the MBean on which the operation was invoked. **Returns:**The object returned by the operation, which represents the result of invoking the operation on the MBean specified. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps an exception thrown by the MBean's invoked method. [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - Wraps a java.lang.Exception thrown while trying to invoke the method. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### getDefaultDomain

[String](http://docs.google.com/java/lang/String.html) **getDefaultDomain**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Returns the default domain used for naming the MBean. The default domain name is used as the domain part in the ObjectName of MBeans if no domain is specified by the user.

**Returns:**the default domain. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### getDomains

[String](http://docs.google.com/java/lang/String.html)[] **getDomains**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Returns the list of domains in which any MBean is currently registered. A string is in the returned array if and only if there is at least one MBean registered with an ObjectName whose [getDomain()](http://docs.google.com/javax/management/ObjectName.html#getDomain()) is equal to that string. The order of strings within the returned array is not defined.

**Returns:**the list of domains. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### addNotificationListener

void **addNotificationListener**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener,  
 [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter,  
 [Object](http://docs.google.com/java/lang/Object.html) handback)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Adds a listener to a registered MBean.

A notification emitted by an MBean will be forwarded by the MBeanServer to the listener. If the source of the notification is a reference to an MBean object, the MBean server will replace it by that MBean's ObjectName. Otherwise the source is unchanged.

**Parameters:**name - The name of the MBean on which the listener should be added.listener - The listener object which will handle the notifications emitted by the registered MBean.filter - The filter object. If filter is null, no filtering will be performed before handling notifications.handback - The context to be sent to the listener when a notification is emitted. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean name provided does not match any of the registered MBeans. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[removeNotificationListener(ObjectName, NotificationListener)](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.NotificationListener)), [removeNotificationListener(ObjectName, NotificationListener, NotificationFilter, Object)](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object))

### addNotificationListener

void **addNotificationListener**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) listener,  
 [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter,  
 [Object](http://docs.google.com/java/lang/Object.html) handback)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Adds a listener to a registered MBean.

A notification emitted by an MBean will be forwarded by the MBeanServer to the listener. If the source of the notification is a reference to an MBean object, the MBean server will replace it by that MBean's ObjectName. Otherwise the source is unchanged.

The listener object that receives notifications is the one that is registered with the given name at the time this method is called. Even if it is subsequently unregistered, it will continue to receive notifications.

**Parameters:**name - The name of the MBean on which the listener should be added.listener - The object name of the listener which will handle the notifications emitted by the registered MBean.filter - The filter object. If filter is null, no filtering will be performed before handling notifications.handback - The context to be sent to the listener when a notification is emitted. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean name of the notification listener or of the notification broadcaster does not match any of the registered MBeans. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps an [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html). The MBean named by listener exists but does not implement the [NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) interface. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[removeNotificationListener(ObjectName, ObjectName)](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.ObjectName)), [removeNotificationListener(ObjectName, ObjectName, NotificationFilter, Object)](http://docs.google.com/javax/management/MBeanServerConnection.html#removeNotificationListener(javax.management.ObjectName,%20javax.management.ObjectName,%20javax.management.NotificationFilter,%20java.lang.Object))

### removeNotificationListener

void **removeNotificationListener**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) listener)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Removes a listener from a registered MBean.

If the listener is registered more than once, perhaps with different filters or callbacks, this method will remove all those registrations.

**Parameters:**name - The name of the MBean on which the listener should be removed.listener - The object name of the listener to be removed. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean name provided does not match any of the registered MBeans. [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html) - The listener is not registered in the MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[addNotificationListener(ObjectName, ObjectName, NotificationFilter, Object)](http://docs.google.com/javax/management/MBeanServerConnection.html#addNotificationListener(javax.management.ObjectName,%20javax.management.ObjectName,%20javax.management.NotificationFilter,%20java.lang.Object))

### removeNotificationListener

void **removeNotificationListener**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) listener,  
 [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter,  
 [Object](http://docs.google.com/java/lang/Object.html) handback)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Removes a listener from a registered MBean.

The MBean must have a listener that exactly matches the given listener, filter, and handback parameters. If there is more than one such listener, only one is removed.

The filter and handback parameters may be null if and only if they are null in a listener to be removed.

**Parameters:**name - The name of the MBean on which the listener should be removed.listener - The object name of the listener to be removed.filter - The filter that was specified when the listener was added.handback - The handback that was specified when the listener was added. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean name provided does not match any of the registered MBeans. [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html) - The listener is not registered in the MBean, or it is not registered with the given filter and handback. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[addNotificationListener(ObjectName, ObjectName, NotificationFilter, Object)](http://docs.google.com/javax/management/MBeanServerConnection.html#addNotificationListener(javax.management.ObjectName,%20javax.management.ObjectName,%20javax.management.NotificationFilter,%20java.lang.Object))

### removeNotificationListener

void **removeNotificationListener**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Removes a listener from a registered MBean.

If the listener is registered more than once, perhaps with different filters or callbacks, this method will remove all those registrations.

**Parameters:**name - The name of the MBean on which the listener should be removed.listener - The listener to be removed. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean name provided does not match any of the registered MBeans. [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html) - The listener is not registered in the MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[addNotificationListener(ObjectName, NotificationListener, NotificationFilter, Object)](http://docs.google.com/javax/management/MBeanServerConnection.html#addNotificationListener(javax.management.ObjectName,%20javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object))

### removeNotificationListener

void **removeNotificationListener**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener,  
 [NotificationFilter](http://docs.google.com/javax/management/NotificationFilter.html) filter,  
 [Object](http://docs.google.com/java/lang/Object.html) handback)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Removes a listener from a registered MBean.

The MBean must have a listener that exactly matches the given listener, filter, and handback parameters. If there is more than one such listener, only one is removed.

The filter and handback parameters may be null if and only if they are null in a listener to be removed.

**Parameters:**name - The name of the MBean on which the listener should be removed.listener - The listener to be removed.filter - The filter that was specified when the listener was added.handback - The handback that was specified when the listener was added. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean name provided does not match any of the registered MBeans. [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html) - The listener is not registered in the MBean, or it is not registered with the given filter and handback. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[addNotificationListener(ObjectName, NotificationListener, NotificationFilter, Object)](http://docs.google.com/javax/management/MBeanServerConnection.html#addNotificationListener(javax.management.ObjectName,%20javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object))

### getMBeanInfo

[MBeanInfo](http://docs.google.com/javax/management/MBeanInfo.html) **getMBeanInfo**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [IntrospectionException](http://docs.google.com/javax/management/IntrospectionException.html),  
 [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

This method discovers the attributes and operations that an MBean exposes for management.

**Parameters:**name - The name of the MBean to analyze **Returns:**An instance of MBeanInfo allowing the retrieval of all attributes and operations of this MBean. **Throws:** [IntrospectionException](http://docs.google.com/javax/management/IntrospectionException.html) - An exception occurred during introspection. [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified was not found. [ReflectionException](http://docs.google.com/javax/management/ReflectionException.html) - An exception occurred when trying to invoke the getMBeanInfo of a Dynamic MBean. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.

### isInstanceOf

boolean **isInstanceOf**([ObjectName](http://docs.google.com/javax/management/ObjectName.html) name,  
 [String](http://docs.google.com/java/lang/String.html) className)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Returns true if the MBean specified is an instance of the specified class, false otherwise.

If name does not name an MBean, this method throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html).

Otherwise, let

X be the MBean named by name,

L be the ClassLoader of X,

N be the class name in X's [MBeanInfo](http://docs.google.com/javax/management/MBeanInfo.html).

If N equals className, the result is true.

Otherwise, if L successfully loads className and X is an instance of this class, the result is true.

Otherwise, if L successfully loads both N and className, and the second class is assignable from the first, the result is true.

Otherwise, the result is false.

**Parameters:**name - The ObjectName of the MBean.className - The name of the class. **Returns:**true if the MBean specified is an instance of the specified class according to the rules above, false otherwise. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The MBean specified is not registered in the MBean server. [IOException](http://docs.google.com/java/io/IOException.html) - A communication problem occurred when talking to the MBean server.**See Also:**[Class.isInstance(java.lang.Object)](http://docs.google.com/java/lang/Class.html#isInstance(java.lang.Object))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MBeanServerConnection.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/management/MBeanServerBuilder.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/MBeanServerDelegate.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/MBeanServerConnection.html)    [**NO FRAMES**](http://docs.google.com/MBeanServerConnection.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).