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

## **javax.naming.ldap**

Interface LdapContext

**All Superinterfaces:** [Context](http://docs.google.com/javax/naming/Context.html), [DirContext](http://docs.google.com/javax/naming/directory/DirContext.html) **All Known Implementing Classes:** [InitialLdapContext](http://docs.google.com/javax/naming/ldap/InitialLdapContext.html)

public interface **LdapContext**extends [DirContext](http://docs.google.com/javax/naming/directory/DirContext.html)

This interface represents a context in which you can perform operations with LDAPv3-style controls and perform LDAPv3-style extended operations. For applications that do not require such controls or extended operations, the more generic javax.naming.directory.DirContext should be used instead.

### Usage Details About Controls

This interface provides support for LDAP v3 controls. At a high level, this support allows a user program to set request controls for LDAP operations that are executed in the course of the user program's invocation of Context/DirContext methods, and read response controls resulting from LDAP operations. At the implementation level, there are some details that developers of both the user program and service providers need to understand in order to correctly use request and response controls.

### Request Controls

There are two types of request controls:

* Request controls that affect how a connection is created
* Request controls that affect context methods

The former is used whenever a connection needs to be established or re-established with an LDAP server. The latter is used when all other LDAP operations are sent to the LDAP server. The reason why a distinction between these two types of request controls is necessary is because JNDI is a high-level API that does not deal directly with connections. It is the job of service providers to do any necessary connection management. Consequently, a single connection may be shared by multiple context instances, and a service provider is free to use its own algorithms to conserve connection and network usage. Thus, when a method is invoked on the context instance, the service provider might need to do some connection management in addition to performing the corresponding LDAP operations. For connection management, it uses the *connection request controls*, while for the normal LDAP operations, it uses the *context request controls*.

Unless explicitly qualified, the term "request controls" refers to context request controls.

#### Context Request Controls

There are two ways in which a context instance gets its request controls:

1. ldapContext.newInstance(**reqCtls**)
2. ldapContext.setRequestControls(**reqCtls**)

where ldapContext is an instance of LdapContext. Specifying null or an empty array for reqCtls means no request controls. newInstance() creates a new instance of a context using reqCtls, while setRequestControls() updates an existing context instance's request controls to reqCtls.

Unlike environment properties, request controls of a context instance *are not inherited* by context instances that are derived from it. Derived context instances have null as their context request controls. You must set the request controls of a derived context instance explicitly using setRequestControls().

A context instance's request controls are retrieved using the method getRequestControls().

#### Connection Request Controls

There are three ways in which connection request controls are set:

1. new InitialLdapContext(env, **connCtls**)
2. refException.getReferralContext(env, **connCtls**)
3. ldapContext.reconnect(**connCtls**);

where refException is an instance of LdapReferralException, and ldapContext is an instance of LdapContext. Specifying null or an empty array for connCtls means no connection request controls.

Like environment properties, connection request controls of a context *are inherited* by contexts that are derived from it. Typically, you initialize the connection request controls using the InitialLdapContext constructor or LdapReferralContext.getReferralContext(). These connection request controls are inherited by contexts that share the same connection--that is, contexts derived from the initial or referral contexts.

Use reconnect() to change the connection request controls of a context. Invoking ldapContext.reconnect() affects only the connection used by ldapContext and any new contexts instances that are derived form ldapContext. Contexts that previously shared the connection with ldapContext remain unchanged. That is, a context's connection request controls must be explicitly changed and is not affected by changes to another context's connection request controls.

A context instance's connection request controls are retrieved using the method getConnectControls().

#### Service Provider Requirements

A service provider supports connection and context request controls in the following ways. Context request controls must be associated on a per context instance basis while connection request controls must be associated on a per connection instance basis. The service provider must look for the connection request controls in the environment property "java.naming.ldap.control.connect" and pass this environment property on to context instances that it creates.

### Response Controls

The method LdapContext.getResponseControls() is used to retrieve the response controls generated by LDAP operations executed as the result of invoking a Context/DirContext operation. The result is all of the responses controls generated by the underlying LDAP operations, including any implicit reconnection. To get only the reconnection response controls, use reconnect() followed by getResponseControls().

### Parameters

A Control[] array passed as a parameter to any method is owned by the caller. The service provider will not modify the array or keep a reference to it, although it may keep references to the individual Control objects in the array. A Control[] array returned by any method is immutable, and may not subsequently be modified by either the caller or the service provider.

**Since:** 1.3 **See Also:**[InitialLdapContext](http://docs.google.com/javax/naming/ldap/InitialLdapContext.html), [LdapReferralException.getReferralContext(java.util.Hashtable,javax.naming.ldap.Control[])](http://docs.google.com/javax/naming/ldap/LdapReferralException.html#getReferralContext(java.util.Hashtable,%20javax.naming.ldap.Control%5B%5D))

| **Field Summary** | |
| --- | --- |
| static [String](http://docs.google.com/java/lang/String.html) | [**CONTROL\_FACTORIES**](http://docs.google.com/javax/naming/ldap/LdapContext.html#CONTROL_FACTORIES)            Constant that holds the name of the environment property for specifying the list of control factories to use. |

| **Fields inherited from interface javax.naming.directory.**[**DirContext**](http://docs.google.com/javax/naming/directory/DirContext.html) |
| --- |
| [ADD\_ATTRIBUTE](http://docs.google.com/javax/naming/directory/DirContext.html#ADD_ATTRIBUTE), [REMOVE\_ATTRIBUTE](http://docs.google.com/javax/naming/directory/DirContext.html#REMOVE_ATTRIBUTE), [REPLACE\_ATTRIBUTE](http://docs.google.com/javax/naming/directory/DirContext.html#REPLACE_ATTRIBUTE) |

| **Fields inherited from interface javax.naming.**[**Context**](http://docs.google.com/javax/naming/Context.html) |
| --- |
| [APPLET](http://docs.google.com/javax/naming/Context.html#APPLET), [AUTHORITATIVE](http://docs.google.com/javax/naming/Context.html#AUTHORITATIVE), [BATCHSIZE](http://docs.google.com/javax/naming/Context.html#BATCHSIZE), [DNS\_URL](http://docs.google.com/javax/naming/Context.html#DNS_URL), [INITIAL\_CONTEXT\_FACTORY](http://docs.google.com/javax/naming/Context.html#INITIAL_CONTEXT_FACTORY), [LANGUAGE](http://docs.google.com/javax/naming/Context.html#LANGUAGE), [OBJECT\_FACTORIES](http://docs.google.com/javax/naming/Context.html#OBJECT_FACTORIES), [PROVIDER\_URL](http://docs.google.com/javax/naming/Context.html#PROVIDER_URL), [REFERRAL](http://docs.google.com/javax/naming/Context.html#REFERRAL), [SECURITY\_AUTHENTICATION](http://docs.google.com/javax/naming/Context.html#SECURITY_AUTHENTICATION), [SECURITY\_CREDENTIALS](http://docs.google.com/javax/naming/Context.html#SECURITY_CREDENTIALS), [SECURITY\_PRINCIPAL](http://docs.google.com/javax/naming/Context.html#SECURITY_PRINCIPAL), [SECURITY\_PROTOCOL](http://docs.google.com/javax/naming/Context.html#SECURITY_PROTOCOL), [STATE\_FACTORIES](http://docs.google.com/javax/naming/Context.html#STATE_FACTORIES), [URL\_PKG\_PREFIXES](http://docs.google.com/javax/naming/Context.html#URL_PKG_PREFIXES) |

| **Method Summary** | |
| --- | --- |
| [ExtendedResponse](http://docs.google.com/javax/naming/ldap/ExtendedResponse.html) | [**extendedOperation**](http://docs.google.com/javax/naming/ldap/LdapContext.html#extendedOperation(javax.naming.ldap.ExtendedRequest))([ExtendedRequest](http://docs.google.com/javax/naming/ldap/ExtendedRequest.html) request)            Performs an extended operation. |
| [Control](http://docs.google.com/javax/naming/ldap/Control.html)[] | [**getConnectControls**](http://docs.google.com/javax/naming/ldap/LdapContext.html#getConnectControls())()            Retrieves the connection request controls in effect for this context. |
| [Control](http://docs.google.com/javax/naming/ldap/Control.html)[] | [**getRequestControls**](http://docs.google.com/javax/naming/ldap/LdapContext.html#getRequestControls())()            Retrieves the request controls in effect for this context. |
| [Control](http://docs.google.com/javax/naming/ldap/Control.html)[] | [**getResponseControls**](http://docs.google.com/javax/naming/ldap/LdapContext.html#getResponseControls())()            Retrieves the response controls produced as a result of the last method invoked on this context. |
| [LdapContext](http://docs.google.com/javax/naming/ldap/LdapContext.html) | [**newInstance**](http://docs.google.com/javax/naming/ldap/LdapContext.html#newInstance(javax.naming.ldap.Control%5B%5D))([Control](http://docs.google.com/javax/naming/ldap/Control.html)[] requestControls)            Creates a new instance of this context initialized using request controls. |
| void | [**reconnect**](http://docs.google.com/javax/naming/ldap/LdapContext.html#reconnect(javax.naming.ldap.Control%5B%5D))([Control](http://docs.google.com/javax/naming/ldap/Control.html)[] connCtls)            Reconnects to the LDAP server using the supplied controls and this context's environment. |
| void | [**setRequestControls**](http://docs.google.com/javax/naming/ldap/LdapContext.html#setRequestControls(javax.naming.ldap.Control%5B%5D))([Control](http://docs.google.com/javax/naming/ldap/Control.html)[] requestControls)            Sets the request controls for methods subsequently invoked on this context. |

| **Methods inherited from interface javax.naming.directory.**[**DirContext**](http://docs.google.com/javax/naming/directory/DirContext.html) |
| --- |
| [bind](http://docs.google.com/javax/naming/directory/DirContext.html#bind(javax.naming.Name,%20java.lang.Object,%20javax.naming.directory.Attributes)), [bind](http://docs.google.com/javax/naming/directory/DirContext.html#bind(java.lang.String,%20java.lang.Object,%20javax.naming.directory.Attributes)), [createSubcontext](http://docs.google.com/javax/naming/directory/DirContext.html#createSubcontext(javax.naming.Name,%20javax.naming.directory.Attributes)), [createSubcontext](http://docs.google.com/javax/naming/directory/DirContext.html#createSubcontext(java.lang.String,%20javax.naming.directory.Attributes)), [getAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#getAttributes(javax.naming.Name)), [getAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#getAttributes(javax.naming.Name,%20java.lang.String%5B%5D)), [getAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#getAttributes(java.lang.String)), [getAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#getAttributes(java.lang.String,%20java.lang.String%5B%5D)), [getSchema](http://docs.google.com/javax/naming/directory/DirContext.html#getSchema(javax.naming.Name)), [getSchema](http://docs.google.com/javax/naming/directory/DirContext.html#getSchema(java.lang.String)), [getSchemaClassDefinition](http://docs.google.com/javax/naming/directory/DirContext.html#getSchemaClassDefinition(javax.naming.Name)), [getSchemaClassDefinition](http://docs.google.com/javax/naming/directory/DirContext.html#getSchemaClassDefinition(java.lang.String)), [modifyAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#modifyAttributes(javax.naming.Name,%20int,%20javax.naming.directory.Attributes)), [modifyAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#modifyAttributes(javax.naming.Name,%20javax.naming.directory.ModificationItem%5B%5D)), [modifyAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#modifyAttributes(java.lang.String,%20int,%20javax.naming.directory.Attributes)), [modifyAttributes](http://docs.google.com/javax/naming/directory/DirContext.html#modifyAttributes(java.lang.String,%20javax.naming.directory.ModificationItem%5B%5D)), [rebind](http://docs.google.com/javax/naming/directory/DirContext.html#rebind(javax.naming.Name,%20java.lang.Object,%20javax.naming.directory.Attributes)), [rebind](http://docs.google.com/javax/naming/directory/DirContext.html#rebind(java.lang.String,%20java.lang.Object,%20javax.naming.directory.Attributes)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(javax.naming.Name,%20javax.naming.directory.Attributes)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(javax.naming.Name,%20javax.naming.directory.Attributes,%20java.lang.String%5B%5D)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(javax.naming.Name,%20java.lang.String,%20java.lang.Object%5B%5D,%20javax.naming.directory.SearchControls)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(javax.naming.Name,%20java.lang.String,%20javax.naming.directory.SearchControls)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(java.lang.String,%20javax.naming.directory.Attributes)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(java.lang.String,%20javax.naming.directory.Attributes,%20java.lang.String%5B%5D)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(java.lang.String,%20java.lang.String,%20java.lang.Object%5B%5D,%20javax.naming.directory.SearchControls)), [search](http://docs.google.com/javax/naming/directory/DirContext.html#search(java.lang.String,%20java.lang.String,%20javax.naming.directory.SearchControls)) |

| **Methods inherited from interface javax.naming.**[**Context**](http://docs.google.com/javax/naming/Context.html) |
| --- |
| [addToEnvironment](http://docs.google.com/javax/naming/Context.html#addToEnvironment(java.lang.String,%20java.lang.Object)), [bind](http://docs.google.com/javax/naming/Context.html#bind(javax.naming.Name,%20java.lang.Object)), [bind](http://docs.google.com/javax/naming/Context.html#bind(java.lang.String,%20java.lang.Object)), [close](http://docs.google.com/javax/naming/Context.html#close()), [composeName](http://docs.google.com/javax/naming/Context.html#composeName(javax.naming.Name,%20javax.naming.Name)), [composeName](http://docs.google.com/javax/naming/Context.html#composeName(java.lang.String,%20java.lang.String)), [createSubcontext](http://docs.google.com/javax/naming/Context.html#createSubcontext(javax.naming.Name)), [createSubcontext](http://docs.google.com/javax/naming/Context.html#createSubcontext(java.lang.String)), [destroySubcontext](http://docs.google.com/javax/naming/Context.html#destroySubcontext(javax.naming.Name)), [destroySubcontext](http://docs.google.com/javax/naming/Context.html#destroySubcontext(java.lang.String)), [getEnvironment](http://docs.google.com/javax/naming/Context.html#getEnvironment()), [getNameInNamespace](http://docs.google.com/javax/naming/Context.html#getNameInNamespace()), [getNameParser](http://docs.google.com/javax/naming/Context.html#getNameParser(javax.naming.Name)), [getNameParser](http://docs.google.com/javax/naming/Context.html#getNameParser(java.lang.String)), [list](http://docs.google.com/javax/naming/Context.html#list(javax.naming.Name)), [list](http://docs.google.com/javax/naming/Context.html#list(java.lang.String)), [listBindings](http://docs.google.com/javax/naming/Context.html#listBindings(javax.naming.Name)), [listBindings](http://docs.google.com/javax/naming/Context.html#listBindings(java.lang.String)), [lookup](http://docs.google.com/javax/naming/Context.html#lookup(javax.naming.Name)), [lookup](http://docs.google.com/javax/naming/Context.html#lookup(java.lang.String)), [lookupLink](http://docs.google.com/javax/naming/Context.html#lookupLink(javax.naming.Name)), [lookupLink](http://docs.google.com/javax/naming/Context.html#lookupLink(java.lang.String)), [rebind](http://docs.google.com/javax/naming/Context.html#rebind(javax.naming.Name,%20java.lang.Object)), [rebind](http://docs.google.com/javax/naming/Context.html#rebind(java.lang.String,%20java.lang.Object)), [removeFromEnvironment](http://docs.google.com/javax/naming/Context.html#removeFromEnvironment(java.lang.String)), [rename](http://docs.google.com/javax/naming/Context.html#rename(javax.naming.Name,%20javax.naming.Name)), [rename](http://docs.google.com/javax/naming/Context.html#rename(java.lang.String,%20java.lang.String)), [unbind](http://docs.google.com/javax/naming/Context.html#unbind(javax.naming.Name)), [unbind](http://docs.google.com/javax/naming/Context.html#unbind(java.lang.String)) |

| **Field Detail** |
| --- |

### CONTROL\_FACTORIES

static final [String](http://docs.google.com/java/lang/String.html) **CONTROL\_FACTORIES**

Constant that holds the name of the environment property for specifying the list of control factories to use. The value of the property should be a colon-separated list of the fully qualified class names of factory classes that will create a control given another control. See ControlFactory.getControlInstance() for details. This property may be specified in the environment, an applet parameter, a system property, or one or more resource files.

The value of this constant is "java.naming.factory.control".

**See Also:**[ControlFactory](http://docs.google.com/javax/naming/ldap/ControlFactory.html), [Context.addToEnvironment(java.lang.String, java.lang.Object)](http://docs.google.com/javax/naming/Context.html#addToEnvironment(java.lang.String,%20java.lang.Object)), [Context.removeFromEnvironment(java.lang.String)](http://docs.google.com/javax/naming/Context.html#removeFromEnvironment(java.lang.String)), [Constant Field Values](http://docs.google.com/constant-values.html#javax.naming.ldap.LdapContext.CONTROL_FACTORIES)

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

### extendedOperation

[ExtendedResponse](http://docs.google.com/javax/naming/ldap/ExtendedResponse.html) **extendedOperation**([ExtendedRequest](http://docs.google.com/javax/naming/ldap/ExtendedRequest.html) request)  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Performs an extended operation. This method is used to support LDAPv3 extended operations.

**Parameters:**request - The non-null request to be performed. **Returns:**The possibly null response of the operation. null means the operation did not generate any response. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If an error occurred while performing the extended operation.

### newInstance

[LdapContext](http://docs.google.com/javax/naming/ldap/LdapContext.html) **newInstance**([Control](http://docs.google.com/javax/naming/ldap/Control.html)[] requestControls)  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Creates a new instance of this context initialized using request controls. This method is a convenience method for creating a new instance of this context for the purposes of multithreaded access. For example, if multiple threads want to use different context request controls, each thread may use this method to get its own copy of this context and set/get context request controls without having to synchronize with other threads.

The new context has the same environment properties and connection request controls as this context. See the class description for details. Implementations might also allow this context and the new context to share the same network connection or other resources if doing so does not impede the independence of either context.

**Parameters:**requestControls - The possibly null request controls to use for the new context. If null, the context is initialized with no request controls. **Returns:**A non-null LdapContext instance. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If an error occurred while creating the new instance.**See Also:**[InitialLdapContext](http://docs.google.com/javax/naming/ldap/InitialLdapContext.html)

### reconnect

void **reconnect**([Control](http://docs.google.com/javax/naming/ldap/Control.html)[] connCtls)  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Reconnects to the LDAP server using the supplied controls and this context's environment.

This method is a way to explicitly initiate an LDAP "bind" operation. For example, you can use this method to set request controls for the LDAP "bind" operation, or to explicitly connect to the server to get response controls returned by the LDAP "bind" operation.

This method sets this context's connCtls to be its new connection request controls. This context's context request controls are not affected. After this method has been invoked, any subsequent implicit reconnections will be done using connCtls. connCtls are also used as connection request controls for new context instances derived from this context. These connection request controls are not affected by setRequestControls().

Service provider implementors should read the "Service Provider" section in the class description for implementation details.

**Parameters:**connCtls - The possibly null controls to use. If null, no controls are used. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If an error occurred while reconnecting.**See Also:**[getConnectControls()](http://docs.google.com/javax/naming/ldap/LdapContext.html#getConnectControls()), [newInstance(javax.naming.ldap.Control[])](http://docs.google.com/javax/naming/ldap/LdapContext.html#newInstance(javax.naming.ldap.Control%5B%5D))

### getConnectControls

[Control](http://docs.google.com/javax/naming/ldap/Control.html)[] **getConnectControls**()  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Retrieves the connection request controls in effect for this context. The controls are owned by the JNDI implementation and are immutable. Neither the array nor the controls may be modified by the caller.

**Returns:**A possibly-null array of controls. null means no connect controls have been set for this context. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If an error occurred while getting the request controls.

### setRequestControls

void **setRequestControls**([Control](http://docs.google.com/javax/naming/ldap/Control.html)[] requestControls)  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Sets the request controls for methods subsequently invoked on this context. The request controls are owned by the JNDI implementation and are immutable. Neither the array nor the controls may be modified by the caller.

This removes any previous request controls and adds requestControls for use by subsequent methods invoked on this context. This method does not affect this context's connection request controls.

Note that requestControls will be in effect until the next invocation of setRequestControls(). You need to explicitly invoke setRequestControls() with null or an empty array to clear the controls if you don't want them to affect the context methods any more. To check what request controls are in effect for this context, use getRequestControls().

**Parameters:**requestControls - The possibly null controls to use. If null, no controls are used. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If an error occurred while setting the request controls.**See Also:**[getRequestControls()](http://docs.google.com/javax/naming/ldap/LdapContext.html#getRequestControls())

### getRequestControls

[Control](http://docs.google.com/javax/naming/ldap/Control.html)[] **getRequestControls**()  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Retrieves the request controls in effect for this context. The request controls are owned by the JNDI implementation and are immutable. Neither the array nor the controls may be modified by the caller.

**Returns:**A possibly-null array of controls. null means no request controls have been set for this context. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If an error occurred while getting the request controls.**See Also:**[setRequestControls(javax.naming.ldap.Control[])](http://docs.google.com/javax/naming/ldap/LdapContext.html#setRequestControls(javax.naming.ldap.Control%5B%5D))

### getResponseControls

[Control](http://docs.google.com/javax/naming/ldap/Control.html)[] **getResponseControls**()  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Retrieves the response controls produced as a result of the last method invoked on this context. The response controls are owned by the JNDI implementation and are immutable. Neither the array nor the controls may be modified by the caller.

These response controls might have been generated by a successful or failed operation.

When a context method that may return response controls is invoked, response controls from the previous method invocation are cleared. getResponseControls() returns all of the response controls generated by LDAP operations used by the context method in the order received from the LDAP server. Invoking getResponseControls() does not clear the response controls. You can call it many times (and get back the same controls) until the next context method that may return controls is invoked.

**Returns:**A possibly null array of controls. If null, the previous method invoked on this context did not produce any controls. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If an error occurred while getting the response controls.

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

[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).