| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ReferralException.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/Referenceable.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/ServiceUnavailableException.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/ReferralException.html)    [**NO FRAMES**](http://docs.google.com/ReferralException.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [METHOD](#lnxbz9) |

## **javax.naming**

Class ReferralException

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.lang.Throwable](http://docs.google.com/java/lang/Throwable.html)  
 [java.lang.Exception](http://docs.google.com/java/lang/Exception.html)  
 [javax.naming.NamingException](http://docs.google.com/javax/naming/NamingException.html)  
 **javax.naming.ReferralException**

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

public abstract class **ReferralException**extends [NamingException](http://docs.google.com/javax/naming/NamingException.html)

This abstract class is used to represent a referral exception, which is generated in response to a *referral* such as that returned by LDAP v3 servers.

A service provider provides a subclass of ReferralException by providing implementations for getReferralInfo() and getReferralContext() (and appropriate constructors and/or corresponding "set" methods).

The following code sample shows how ReferralException can be used.

while (true) {  
 try {  
 bindings = ctx.listBindings(name);  
 while (bindings.hasMore()) {  
 b = bindings.next();  
 ...  
 }  
 break;  
 } catch (ReferralException e) {  
 ctx = e.getReferralContext();  
 }  
 }

ReferralException is an abstract class. Concrete implementations determine its synchronization and serialization properties.

An environment parameter passed to the getReferralContext() method is owned by the caller. The service provider will not modify the object or keep a reference to it, but may keep a reference to a clone of it.

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

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class javax.naming.**[**NamingException**](http://docs.google.com/javax/naming/NamingException.html) |
| --- |
| [remainingName](http://docs.google.com/javax/naming/NamingException.html#remainingName), [resolvedName](http://docs.google.com/javax/naming/NamingException.html#resolvedName), [resolvedObj](http://docs.google.com/javax/naming/NamingException.html#resolvedObj), [rootException](http://docs.google.com/javax/naming/NamingException.html#rootException) |

| **Constructor Summary** | |
| --- | --- |
| protected | [**ReferralException**](http://docs.google.com/javax/naming/ReferralException.html#ReferralException())()            Constructs a new instance of ReferralException. |
| protected | [**ReferralException**](http://docs.google.com/javax/naming/ReferralException.html#ReferralException(java.lang.String))([String](http://docs.google.com/java/lang/String.html) explanation)            Constructs a new instance of ReferralException using the explanation supplied. |

| **Method Summary** | |
| --- | --- |
| abstract  [Context](http://docs.google.com/javax/naming/Context.html) | [**getReferralContext**](http://docs.google.com/javax/naming/ReferralException.html#getReferralContext())()            Retrieves the context at which to continue the method. |
| abstract  [Context](http://docs.google.com/javax/naming/Context.html) | [**getReferralContext**](http://docs.google.com/javax/naming/ReferralException.html#getReferralContext(java.util.Hashtable))([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> env)            Retrieves the context at which to continue the method using environment properties. |
| abstract  [Object](http://docs.google.com/java/lang/Object.html) | [**getReferralInfo**](http://docs.google.com/javax/naming/ReferralException.html#getReferralInfo())()            Retrieves information (such as URLs) related to this referral. |
| abstract  void | [**retryReferral**](http://docs.google.com/javax/naming/ReferralException.html#retryReferral())()            Retries the referral currently being processed. |
| abstract  boolean | [**skipReferral**](http://docs.google.com/javax/naming/ReferralException.html#skipReferral())()            Discards the referral about to be processed. |

| **Methods inherited from class javax.naming.**[**NamingException**](http://docs.google.com/javax/naming/NamingException.html) |
| --- |
| [appendRemainingComponent](http://docs.google.com/javax/naming/NamingException.html#appendRemainingComponent(java.lang.String)), [appendRemainingName](http://docs.google.com/javax/naming/NamingException.html#appendRemainingName(javax.naming.Name)), [getCause](http://docs.google.com/javax/naming/NamingException.html#getCause()), [getExplanation](http://docs.google.com/javax/naming/NamingException.html#getExplanation()), [getRemainingName](http://docs.google.com/javax/naming/NamingException.html#getRemainingName()), [getResolvedName](http://docs.google.com/javax/naming/NamingException.html#getResolvedName()), [getResolvedObj](http://docs.google.com/javax/naming/NamingException.html#getResolvedObj()), [getRootCause](http://docs.google.com/javax/naming/NamingException.html#getRootCause()), [initCause](http://docs.google.com/javax/naming/NamingException.html#initCause(java.lang.Throwable)), [setRemainingName](http://docs.google.com/javax/naming/NamingException.html#setRemainingName(javax.naming.Name)), [setResolvedName](http://docs.google.com/javax/naming/NamingException.html#setResolvedName(javax.naming.Name)), [setResolvedObj](http://docs.google.com/javax/naming/NamingException.html#setResolvedObj(java.lang.Object)), [setRootCause](http://docs.google.com/javax/naming/NamingException.html#setRootCause(java.lang.Throwable)), [toString](http://docs.google.com/javax/naming/NamingException.html#toString()), [toString](http://docs.google.com/javax/naming/NamingException.html#toString(boolean)) |

| **Methods inherited from class java.lang.**[**Throwable**](http://docs.google.com/java/lang/Throwable.html) |
| --- |
| [fillInStackTrace](http://docs.google.com/java/lang/Throwable.html#fillInStackTrace()), [getLocalizedMessage](http://docs.google.com/java/lang/Throwable.html#getLocalizedMessage()), [getMessage](http://docs.google.com/java/lang/Throwable.html#getMessage()), [getStackTrace](http://docs.google.com/java/lang/Throwable.html#getStackTrace()), [printStackTrace](http://docs.google.com/java/lang/Throwable.html#printStackTrace()), [printStackTrace](http://docs.google.com/java/lang/Throwable.html#printStackTrace(java.io.PrintStream)), [printStackTrace](http://docs.google.com/java/lang/Throwable.html#printStackTrace(java.io.PrintWriter)), [setStackTrace](http://docs.google.com/java/lang/Throwable.html#setStackTrace(java.lang.StackTraceElement%5B%5D)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### ReferralException

protected **ReferralException**([String](http://docs.google.com/java/lang/String.html) explanation)

Constructs a new instance of ReferralException using the explanation supplied. All other fields are set to null.

**Parameters:**explanation - Additional detail about this exception. Can be null.**See Also:**[Throwable.getMessage()](http://docs.google.com/java/lang/Throwable.html#getMessage())

### ReferralException

protected **ReferralException**()

Constructs a new instance of ReferralException. All fields are set to null.

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

### getReferralInfo

public abstract [Object](http://docs.google.com/java/lang/Object.html) **getReferralInfo**()

Retrieves information (such as URLs) related to this referral. The program may examine or display this information to the user to determine whether to continue with the referral, or to determine additional information needs to be supplied in order to continue with the referral.

**Returns:**Non-null referral information related to this referral.

### getReferralContext

public abstract [Context](http://docs.google.com/javax/naming/Context.html) **getReferralContext**()  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Retrieves the context at which to continue the method. Regardless of whether a referral is encountered directly during a context operation, or indirectly, for example, during a search enumeration, the referral exception should provide a context at which to continue the operation. The referral context is created using the environment properties of the context that threw the ReferralException.

To continue the operation, the client program should re-invoke the method using the same arguments as the original invocation.

**Returns:**The non-null context at which to continue the method. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If a naming exception was encountered. Call either retryReferral() or skipReferral() to continue processing referrals.

### getReferralContext

public abstract [Context](http://docs.google.com/javax/naming/Context.html) **getReferralContext**([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> env)  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Retrieves the context at which to continue the method using environment properties. Regardless of whether a referral is encountered directly during a context operation, or indirectly, for example, during a search enumeration, the referral exception should provide a context at which to continue the operation.

The referral context is created using env as its environment properties. This method should be used instead of the no-arg overloaded form when the caller needs to use different environment properties for the referral context. It might need to do this, for example, when it needs to supply different authentication information to the referred server in order to create the referral context.

To continue the operation, the client program should re-invoke the method using the same arguments as the original invocation.

**Parameters:**env - The possibly null environment to use when retrieving the referral context. If null, no environment properties will be used. **Returns:**The non-null context at which to continue the method. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If a naming exception was encountered. Call either retryReferral() or skipReferral() to continue processing referrals.

### skipReferral

public abstract boolean **skipReferral**()

Discards the referral about to be processed. A call to this method should be followed by a call to getReferralContext to allow the processing of other referrals to continue. The following code fragment shows a typical usage pattern.

} catch (ReferralException e) {  
 if (!shallIFollow(e.getReferralInfo())) {  
 if (!e.skipReferral()) {  
 return;  
 }  
 }  
 ctx = e.getReferralContext();  
 }

**Returns:**true If more referral processing is pending; false otherwise.

### retryReferral

public abstract void **retryReferral**()

Retries the referral currently being processed. A call to this method should be followed by a call to getReferralContext to allow the current referral to be retried. The following code fragment shows a typical usage pattern.

} catch (ReferralException e) {  
 while (true) {  
 try {  
 ctx = e.getReferralContext(env);  
 break;  
 } catch (NamingException ne) {  
 if (! shallIRetry()) {  
 return;  
 }  
 // modify environment properties (env), if necessary  
 e.retryReferral();  
 }  
 }  
 }

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ReferralException.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/Referenceable.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/ServiceUnavailableException.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/ReferralException.html)    [**NO FRAMES**](http://docs.google.com/ReferralException.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [METHOD](#lnxbz9) |

[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).