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

## **javax.naming.ldap**

Class Rdn

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

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Comparable](http://docs.google.com/java/lang/Comparable.html)<[Object](http://docs.google.com/java/lang/Object.html)>

public class **Rdn**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html), [Comparable](http://docs.google.com/java/lang/Comparable.html)<[Object](http://docs.google.com/java/lang/Object.html)>

This class represents a relative distinguished name, or RDN, which is a component of a distinguished name as specified by [RFC 2253](http://ietf.org/rfc/rfc2253.txt). An example of an RDN is "OU=Sales+CN=J.Smith". In this example, the RDN consist of multiple attribute type/value pairs. The RDN is parsed as described in the class description for [LdapName](http://docs.google.com/javax/naming/ldap/LdapName.html).

The Rdn class represents an RDN as attribute type/value mappings, which can be viewed using [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html). In addition, it contains convenience methods that allow easy retrieval of type and value when the Rdn consist of a single type/value pair, which is how it appears in a typical usage. It also contains helper methods that allow escaping of the unformatted attribute value and unescaping of the value formatted according to the escaping syntax defined in RFC2253. For methods that take or return attribute value as an Object, the value is either a String (in unescaped form) or a byte array.

Rdn will properly parse all valid RDNs, but does not attempt to detect all possible violations when parsing invalid RDNs. It is "generous" in accepting invalid RDNs. The "validity" of a name is determined ultimately when it is supplied to an LDAP server, which may accept or reject the name based on factors such as its schema information and interoperability considerations.

The following code example shows how to construct an Rdn using the constructor that takes type and value as arguments:

Rdn rdn = new Rdn("cn", "Juicy, Fruit");  
 System.out.println(rdn.toString());

The last line will print cn=Juicy\, Fruit. The [unescapeValue()](http://docs.google.com/javax/naming/ldap/Rdn.html#unescapeValue(java.lang.String)) method can be used to unescape the escaped comma resulting in the original value "Juicy, Fruit". The [escapeValue()](http://docs.google.com/javax/naming/ldap/Rdn.html#escapeValue(java.lang.Object)) method adds the escape back preceding the comma.

This class can be instantiated by a string representation of the RDN defined in RFC 2253 as shown in the following code example:

Rdn rdn = new Rdn("cn=Juicy\\, Fruit");  
 System.out.println(rdn.toString());

The last line will print cn=Juicy\, Fruit.

Concurrent multithreaded read-only access of an instance of Rdn need not be synchronized.

Unless otherwise noted, the behavior of passing a null argument to a constructor or method in this class will cause NullPointerException to be thrown.

**Since:** 1.5 **See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#javax.naming.ldap.Rdn)

| **Constructor Summary** | |
| --- | --- |
| [**Rdn**](http://docs.google.com/javax/naming/ldap/Rdn.html#Rdn(javax.naming.directory.Attributes))([Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) attrSet)            Constructs an Rdn from the given attribute set. |
| [**Rdn**](http://docs.google.com/javax/naming/ldap/Rdn.html#Rdn(javax.naming.ldap.Rdn))([Rdn](http://docs.google.com/javax/naming/ldap/Rdn.html) rdn)            Constructs an Rdn from the given rdn. |
| [**Rdn**](http://docs.google.com/javax/naming/ldap/Rdn.html#Rdn(java.lang.String))([String](http://docs.google.com/java/lang/String.html) rdnString)            Constructs an Rdn from the given string. |
| [**Rdn**](http://docs.google.com/javax/naming/ldap/Rdn.html#Rdn(java.lang.String,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) type, [Object](http://docs.google.com/java/lang/Object.html) value)            Constructs an Rdn from the given attribute type and value. |

| **Method Summary** | |
| --- | --- |
| int | [**compareTo**](http://docs.google.com/javax/naming/ldap/Rdn.html#compareTo(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Compares this Rdn with the specified Object for order. |
| boolean | [**equals**](http://docs.google.com/javax/naming/ldap/Rdn.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Compares the specified Object with this Rdn for equality. |
| static [String](http://docs.google.com/java/lang/String.html) | [**escapeValue**](http://docs.google.com/javax/naming/ldap/Rdn.html#escapeValue(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) val)            Given the value of an attribute, returns a string escaped according to the rules specified in [RFC 2253](http://ietf.org/rfc/rfc2253.txt). |
| [String](http://docs.google.com/java/lang/String.html) | [**getType**](http://docs.google.com/javax/naming/ldap/Rdn.html#getType())()            Retrieves one of this Rdn's type. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getValue**](http://docs.google.com/javax/naming/ldap/Rdn.html#getValue())()            Retrieves one of this Rdn's value. |
| int | [**hashCode**](http://docs.google.com/javax/naming/ldap/Rdn.html#hashCode())()            Returns the hash code of this RDN. |
| int | [**size**](http://docs.google.com/javax/naming/ldap/Rdn.html#size())()            Retrieves the number of attribute type/value pairs in this Rdn. |
| [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) | [**toAttributes**](http://docs.google.com/javax/naming/ldap/Rdn.html#toAttributes())()            Retrieves the [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) view of the type/value mappings contained in this Rdn. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/naming/ldap/Rdn.html#toString())()            Returns this Rdn as a string represented in a format defined by [RFC 2253](http://ietf.org//rfc/rfc2253.txt) and described in the class description for [LdapName](http://docs.google.com/javax/naming/ldap/LdapName.html). |
| static [Object](http://docs.google.com/java/lang/Object.html) | [**unescapeValue**](http://docs.google.com/javax/naming/ldap/Rdn.html#unescapeValue(java.lang.String))([String](http://docs.google.com/java/lang/String.html) val)            Given an attribute value string formated according to the rules specified in [RFC 2253](http://ietf.org//rfc/rfc2253.txt), returns the unformated value. |

| **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()), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [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** |
| --- |

### Rdn

public **Rdn**([Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) attrSet)  
 throws [InvalidNameException](http://docs.google.com/javax/naming/InvalidNameException.html)

Constructs an Rdn from the given attribute set. See [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html).

The string attribute values are not interpretted as [RFC 2253](http://ietf.org/rfc/rfc2253.txt) formatted RDN strings. That is, the values are used literally (not parsed) and assumed to be unescaped.

**Parameters:**attrSet - The non-null and non-empty attributes containing type/value mappings. **Throws:** [InvalidNameException](http://docs.google.com/javax/naming/InvalidNameException.html) - If contents of attrSet cannot be used to construct a valid RDN.

### Rdn

public **Rdn**([String](http://docs.google.com/java/lang/String.html) rdnString)  
 throws [InvalidNameException](http://docs.google.com/javax/naming/InvalidNameException.html)

Constructs an Rdn from the given string. This constructor takes a string formatted according to the rules defined in [RFC 2253](http://ietf.org//rfc/rfc2253.txt) and described in the class description for [LdapName](http://docs.google.com/javax/naming/ldap/LdapName.html).

**Parameters:**rdnString - The non-null and non-empty RFC2253 formatted string. **Throws:** [InvalidNameException](http://docs.google.com/javax/naming/InvalidNameException.html) - If a syntax error occurs during parsing of the rdnString.

### Rdn

public **Rdn**([Rdn](http://docs.google.com/javax/naming/ldap/Rdn.html) rdn)

Constructs an Rdn from the given rdn. The contents of the rdn are simply copied into the newly created Rdn.

**Parameters:**rdn - The non-null Rdn to be copied.

### Rdn

public **Rdn**([String](http://docs.google.com/java/lang/String.html) type,  
 [Object](http://docs.google.com/java/lang/Object.html) value)  
 throws [InvalidNameException](http://docs.google.com/javax/naming/InvalidNameException.html)

Constructs an Rdn from the given attribute type and value. The string attribute values are not interpretted as [RFC 2253](http://ietf.org/rfc/rfc2253.txt) formatted RDN strings. That is, the values are used literally (not parsed) and assumed to be unescaped.

**Parameters:**type - The non-null and non-empty string attribute type.value - The non-null and non-empty attribute value. **Throws:** [InvalidNameException](http://docs.google.com/javax/naming/InvalidNameException.html) - If type/value cannot be used to construct a valid RDN.**See Also:**[toString()](http://docs.google.com/javax/naming/ldap/Rdn.html#toString())

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

### getValue

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

Retrieves one of this Rdn's value. This is a convenience method for obtaining the value, when the RDN contains a single type and value mapping, which is the common RDN usage.

For a multi-valued RDN, this method returns value corresponding to the type returned by [getType()](http://docs.google.com/javax/naming/ldap/Rdn.html#getType()) method.

**Returns:**The non-null attribute value.

### getType

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

Retrieves one of this Rdn's type. This is a convenience method for obtaining the type, when the RDN contains a single type and value mapping, which is the common RDN usage.

For a multi-valued RDN, the type/value pairs have no specific order defined on them. In that case, this method returns type of one of the type/value pairs. The [getValue()](http://docs.google.com/javax/naming/ldap/Rdn.html#getValue()) method returns the value corresponding to the type returned by this method.

**Returns:**The non-null attribute type.

### toString

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

Returns this Rdn as a string represented in a format defined by [RFC 2253](http://ietf.org//rfc/rfc2253.txt) and described in the class description for [LdapName](http://docs.google.com/javax/naming/ldap/LdapName.html).

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

### compareTo

public int **compareTo**([Object](http://docs.google.com/java/lang/Object.html) obj)

Compares this Rdn with the specified Object for order. Returns a negative integer, zero, or a positive integer as this Rdn is less than, equal to, or greater than the given Object.

If obj is null or not an instance of Rdn, ClassCastException is thrown.

The attribute type and value pairs of the RDNs are lined up against each other and compared lexicographically. The order of components in multi-valued Rdns (such as "ou=Sales+cn=Bob") is not significant.

**Specified by:**[compareTo](http://docs.google.com/java/lang/Comparable.html#compareTo(T)) in interface [Comparable](http://docs.google.com/java/lang/Comparable.html)<[Object](http://docs.google.com/java/lang/Object.html)> **Parameters:**obj - The non-null object to compare against. **Returns:**A negative integer, zero, or a positive integer as this Rdn is less than, equal to, or greater than the given Object. **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if obj is null or not a Rdn.

### equals

public boolean **equals**([Object](http://docs.google.com/java/lang/Object.html) obj)

Compares the specified Object with this Rdn for equality. Returns true if the given object is also a Rdn and the two Rdns represent the same attribute type and value mappings. The order of components in multi-valued Rdns (such as "ou=Sales+cn=Bob") is not significant.

Type and value equalilty matching is done as below:

* The types are compared for equality with their case ignored.
* String values with different but equivalent usage of quoting, escaping, or UTF8-hex-encoding are considered equal. The case of the values is ignored during the comparison.

If obj is null or not an instance of Rdn, false is returned.

**Overrides:**[equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) in class [Object](http://docs.google.com/java/lang/Object.html) **Parameters:**obj - object to be compared for equality with this Rdn. **Returns:**true if the specified object is equal to this Rdn.**See Also:**[hashCode()](http://docs.google.com/javax/naming/ldap/Rdn.html#hashCode())

### hashCode

public int **hashCode**()

Returns the hash code of this RDN. Two RDNs that are equal (according to the equals method) will have the same hash code.

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**An int representing the hash code of this Rdn.**See Also:**[equals(java.lang.Object)](http://docs.google.com/javax/naming/ldap/Rdn.html#equals(java.lang.Object))

### toAttributes

public [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) **toAttributes**()

Retrieves the [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) view of the type/value mappings contained in this Rdn.

**Returns:**The non-null attributes containing the type/value mappings of this Rdn.

### size

public int **size**()

Retrieves the number of attribute type/value pairs in this Rdn.

**Returns:**The non-negative number of type/value pairs in this Rdn.

### escapeValue

public static [String](http://docs.google.com/java/lang/String.html) **escapeValue**([Object](http://docs.google.com/java/lang/Object.html) val)

Given the value of an attribute, returns a string escaped according to the rules specified in [RFC 2253](http://ietf.org/rfc/rfc2253.txt).

For example, if the val is "Sue, Grabbit and Runn", the escaped value returned by this method is "Sue\, Grabbit and Runn".

A string value is represented as a String and binary value as a byte array.

**Parameters:**val - The non-null object to be escaped. **Returns:**Escaped string value. **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if val is is not a String or byte array.

### unescapeValue

public static [Object](http://docs.google.com/java/lang/Object.html) **unescapeValue**([String](http://docs.google.com/java/lang/String.html) val)

Given an attribute value string formated according to the rules specified in [RFC 2253](http://ietf.org//rfc/rfc2253.txt), returns the unformated value. Escapes and quotes are stripped away, and hex-encoded UTF-8 is converted to equivalent UTF-16 characters. Returns a string value as a String, and a binary value as a byte array.

Legal and illegal values are defined in RFC 2253. This method is generous in accepting the values and does not catch all illegal values. Therefore, passing in an illegal value might not necessarily trigger an IllegalArgumentException.

**Parameters:**val - The non-null string to be unescaped. **Returns:**Unescaped value. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - When an Illegal value is provided.

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