| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DescriptorSupport.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   [**NEXT CLASS**](http://docs.google.com/javax/management/modelmbean/InvalidTargetObjectTypeException.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/modelmbean/DescriptorSupport.html)    [**NO FRAMES**](http://docs.google.com/DescriptorSupport.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#lnxbz9) |

## **javax.management.modelmbean**

Class DescriptorSupport

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.management.modelmbean.DescriptorSupport**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html), [Descriptor](http://docs.google.com/javax/management/Descriptor.html)

public class **DescriptorSupport**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Descriptor](http://docs.google.com/javax/management/Descriptor.html)

This class represents the metadata set for a ModelMBean element. A descriptor is part of the ModelMBeanInfo, ModelMBeanNotificationInfo, ModelMBeanAttributeInfo, ModelMBeanConstructorInfo, and ModelMBeanParameterInfo.

A descriptor consists of a collection of fields. Each field is in fieldname=fieldvalue format. Field names are not case sensitive, case will be preserved on field values.

All field names and values are not predefined. New fields can be defined and added by any program. Some fields have been predefined for consistency of implementation and support by the ModelMBeanInfo, ModelMBeanAttributeInfo, ModelMBeanConstructorInfo, ModelMBeanNotificationInfo, ModelMBeanOperationInfo and ModelMBean classes.

The **serialVersionUID** of this class is -6292969195866300415L.

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

| **Constructor Summary** | |
| --- | --- |
| [**DescriptorSupport**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#DescriptorSupport())()            Descriptor default constructor. |
| [**DescriptorSupport**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#DescriptorSupport(javax.management.modelmbean.DescriptorSupport))([DescriptorSupport](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html) inDescr)            Descriptor constructor taking a Descriptor as parameter. |
| [**DescriptorSupport**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#DescriptorSupport(int))(int initNumFields)            Descriptor constructor. |
| [**DescriptorSupport**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#DescriptorSupport(java.lang.String...))([String](http://docs.google.com/java/lang/String.html)... fields)            Constructor taking fields in the *fieldName=fieldValue* format. |
| [**DescriptorSupport**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#DescriptorSupport(java.lang.String))([String](http://docs.google.com/java/lang/String.html) inStr)            Descriptor constructor taking an XML String. |
| [**DescriptorSupport**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#DescriptorSupport(java.lang.String%5B%5D,%20java.lang.Object%5B%5D))([String](http://docs.google.com/java/lang/String.html)[] fieldNames, [Object](http://docs.google.com/java/lang/Object.html)[] fieldValues)            Constructor taking field names and field values. |

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**clone**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#clone())()            Returns a new Descriptor which is a duplicate of the Descriptor. |
| boolean | [**equals**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) o)            Compares this descriptor to the given object. |
| [String](http://docs.google.com/java/lang/String.html)[] | [**getFieldNames**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#getFieldNames())()            Returns all the field names in the descriptor. |
| [String](http://docs.google.com/java/lang/String.html)[] | [**getFields**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#getFields())()            Returns all of the fields contained in this descriptor as a string array. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getFieldValue**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#getFieldValue(java.lang.String))([String](http://docs.google.com/java/lang/String.html) fieldName)            Returns the value for a specific field name, or null if no value is present for that name. |
| [Object](http://docs.google.com/java/lang/Object.html)[] | [**getFieldValues**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#getFieldValues(java.lang.String...))([String](http://docs.google.com/java/lang/String.html)... fieldNames)            Returns all the field values in the descriptor as an array of Objects. |
| int | [**hashCode**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#hashCode())()            Returns the hash code value for this descriptor. |
| boolean | [**isValid**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#isValid())()            Returns true if all of the fields have legal values given their names. |
| void | [**removeField**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#removeField(java.lang.String))([String](http://docs.google.com/java/lang/String.html) fieldName)            Removes a field from the descriptor. |
| void | [**setField**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#setField(java.lang.String,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) fieldName, [Object](http://docs.google.com/java/lang/Object.html) fieldValue)            Sets the value for a specific field name. |
| void | [**setFields**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#setFields(java.lang.String%5B%5D,%20java.lang.Object%5B%5D))([String](http://docs.google.com/java/lang/String.html)[] fieldNames, [Object](http://docs.google.com/java/lang/Object.html)[] fieldValues)            Sets all fields in the field names array to the new value with the same index in the field values array. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#toString())()            Returns a human readable string representing the descriptor. |
| [String](http://docs.google.com/java/lang/String.html) | [**toXMLString**](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#toXMLString())()            Returns an XML String representing the descriptor. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [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** |
| --- |

### DescriptorSupport

public **DescriptorSupport**()

Descriptor default constructor. Default initial descriptor size is 20. It will grow as needed.

Note that the created empty descriptor is not a valid descriptor (the method [isValid](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#isValid()) returns false)

### DescriptorSupport

public **DescriptorSupport**(int initNumFields)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Descriptor constructor. Takes as parameter the initial capacity of the Map that stores the descriptor fields. Capacity will grow as needed.

Note that the created empty descriptor is not a valid descriptor (the method [isValid](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#isValid()) returns false).

**Parameters:**initNumFields - The initial capacity of the Map that stores the descriptor fields. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - for illegal value for initNumFields (<= 0) [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception.

### DescriptorSupport

public **DescriptorSupport**([DescriptorSupport](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html) inDescr)

Descriptor constructor taking a Descriptor as parameter. Creates a new descriptor initialized to the values of the descriptor passed in parameter.

**Parameters:**inDescr - the descriptor to be used to initialize the constructed descriptor. If it is null or contains no descriptor fields, an empty Descriptor will be created.

### DescriptorSupport

public **DescriptorSupport**([String](http://docs.google.com/java/lang/String.html) inStr)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html),  
 [XMLParseException](http://docs.google.com/javax/management/modelmbean/XMLParseException.html)

Descriptor constructor taking an XML String.

The format of the XML string is not defined, but an implementation must ensure that the string returned by [toXMLString()](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#toXMLString()) on an existing descriptor can be used to instantiate an equivalent descriptor using this constructor.

In this implementation, all field values will be created as Strings. If the field values are not Strings, the programmer will have to reset or convert these fields correctly.

**Parameters:**inStr - An XML-formatted string used to populate this Descriptor. The format is not defined, but any implementation must ensure that the string returned by method [toXMLString](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#toXMLString()) on an existing descriptor can be used to instantiate an equivalent descriptor when instantiated using this constructor. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - If the String inStr passed in parameter is null [XMLParseException](http://docs.google.com/javax/management/modelmbean/XMLParseException.html) - XML parsing problem while parsing the input String [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception.

### DescriptorSupport

public **DescriptorSupport**([String](http://docs.google.com/java/lang/String.html)[] fieldNames,  
 [Object](http://docs.google.com/java/lang/Object.html)[] fieldValues)  
 throws [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Constructor taking field names and field values. Neither array can be null.

**Parameters:**fieldNames - String array of field names. No elements of this array can be null.fieldValues - Object array of the corresponding field values. Elements of the array can be null. The fieldValue must be valid for the fieldName (as defined in method [isValid](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#isValid()))

Note: array sizes of parameters should match. If both arrays are empty, then an empty descriptor is created.

**Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - for illegal value for field Names or field Values. The array lengths must be equal. If the descriptor construction fails for any reason, this exception will be thrown.

### DescriptorSupport

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

Constructor taking fields in the *fieldName=fieldValue* format.

**Parameters:**fields - String array with each element containing a field name and value. If this array is null or empty, then the default constructor will be executed. Null strings or empty strings will be ignored.

All field values should be Strings. If the field values are not Strings, the programmer will have to reset or convert these fields correctly.

Note: Each string should be of the form *fieldName=fieldValue*. The field name ends at the first = character; for example if the String is a=b=c then the field name is a and its value is b=c.

**Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - for illegal value for field Names or field Values. The field must contain an "=". "=fieldValue", "fieldName", and "fieldValue" are illegal. FieldName cannot be null. "fieldName=" will cause the value to be null. If the descriptor construction fails for any reason, this exception will be thrown.

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

### getFieldValue

public [Object](http://docs.google.com/java/lang/Object.html) **getFieldValue**([String](http://docs.google.com/java/lang/String.html) fieldName)  
 throws [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

**Description copied from interface:** [**Descriptor**](http://docs.google.com/javax/management/Descriptor.html#getFieldValue(java.lang.String)) Returns the value for a specific field name, or null if no value is present for that name.

**Specified by:**[getFieldValue](http://docs.google.com/javax/management/Descriptor.html#getFieldValue(java.lang.String)) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Parameters:**fieldName - the field name. **Returns:**the corresponding value, or null if the field is not present. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - if the field name is illegal.

### setField

public void **setField**([String](http://docs.google.com/java/lang/String.html) fieldName,  
 [Object](http://docs.google.com/java/lang/Object.html) fieldValue)  
 throws [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

**Description copied from interface:** [**Descriptor**](http://docs.google.com/javax/management/Descriptor.html#setField(java.lang.String,%20java.lang.Object))

Sets the value for a specific field name. This will modify an existing field or add a new field.

The field value will be validated before it is set. If it is not valid, then an exception will be thrown. The meaning of validity is dependent on the descriptor implementation.

**Specified by:**[setField](http://docs.google.com/javax/management/Descriptor.html#setField(java.lang.String,%20java.lang.Object)) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Parameters:**fieldName - The field name to be set. Cannot be null or empty.fieldValue - The field value to be set for the field name. Can be null if that is a valid value for the field. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - if the field name or field value is illegal (wrapped exception is [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html)); or if the descriptor is immutable (wrapped exception is [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html)).

### getFields

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

**Description copied from interface:** [**Descriptor**](http://docs.google.com/javax/management/Descriptor.html#getFields()) Returns all of the fields contained in this descriptor as a string array.

**Specified by:**[getFields](http://docs.google.com/javax/management/Descriptor.html#getFields()) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Returns:**String array of fields in the format *fieldName=fieldValue*

If the value of a field is not a String, then the toString() method will be called on it and the returned value, enclosed in parentheses, used as the value for the field in the returned array. If the value of a field is null, then the value of the field in the returned array will be empty. If the descriptor is empty, you will get an empty array.**See Also:**[Descriptor.setFields(java.lang.String[], java.lang.Object[])](http://docs.google.com/javax/management/Descriptor.html#setFields(java.lang.String%5B%5D,%20java.lang.Object%5B%5D))

### getFieldNames

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

**Description copied from interface:** [**Descriptor**](http://docs.google.com/javax/management/Descriptor.html#getFieldNames()) Returns all the field names in the descriptor.

**Specified by:**[getFieldNames](http://docs.google.com/javax/management/Descriptor.html#getFieldNames()) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Returns:**String array of field names. If the descriptor is empty, you will get an empty array.

### getFieldValues

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

**Description copied from interface:** [**Descriptor**](http://docs.google.com/javax/management/Descriptor.html#getFieldValues(java.lang.String...)) Returns all the field values in the descriptor as an array of Objects. The returned values are in the same order as the fieldNames String array parameter.

**Specified by:**[getFieldValues](http://docs.google.com/javax/management/Descriptor.html#getFieldValues(java.lang.String...)) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Parameters:**fieldNames - String array of the names of the fields that the values should be returned for. If the array is empty then an empty array will be returned. If the array is null then all values will be returned, as if the parameter were the array returned by [Descriptor.getFieldNames()](http://docs.google.com/javax/management/Descriptor.html#getFieldNames()). If a field name in the array does not exist, including the case where it is null or the empty string, then null is returned for the matching array element being returned. **Returns:**Object array of field values. If the list of fieldNames is empty, you will get an empty array.

### setFields

public void **setFields**([String](http://docs.google.com/java/lang/String.html)[] fieldNames,  
 [Object](http://docs.google.com/java/lang/Object.html)[] fieldValues)  
 throws [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

**Description copied from interface:** [**Descriptor**](http://docs.google.com/javax/management/Descriptor.html#setFields(java.lang.String%5B%5D,%20java.lang.Object%5B%5D))

Sets all fields in the field names array to the new value with the same index in the field values array. Array sizes must match.

The field value will be validated before it is set. If it is not valid, then an exception will be thrown. If the arrays are empty, then no change will take effect.

**Specified by:**[setFields](http://docs.google.com/javax/management/Descriptor.html#setFields(java.lang.String%5B%5D,%20java.lang.Object%5B%5D)) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Parameters:**fieldNames - String array of field names. The array and array elements cannot be null.fieldValues - Object array of the corresponding field values. The array cannot be null. Elements of the array can be null. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - if the change fails for any reason. Wrapped exception is [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) if fieldNames or fieldValues is null, or if the arrays are of different lengths, or if there is an illegal value in one of them. Wrapped exception is [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) if the descriptor is immutable, and the call would change its contents.**See Also:**[Descriptor.getFields()](http://docs.google.com/javax/management/Descriptor.html#getFields())

### clone

public [Object](http://docs.google.com/java/lang/Object.html) **clone**()  
 throws [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Returns a new Descriptor which is a duplicate of the Descriptor.

**Specified by:**[clone](http://docs.google.com/javax/management/Descriptor.html#clone()) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html)**Overrides:**[clone](http://docs.google.com/java/lang/Object.html#clone()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a clone of this instance. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - for illegal value for field Names or field Values. If the descriptor construction fails for any reason, this exception will be thrown.**See Also:**[Cloneable](http://docs.google.com/java/lang/Cloneable.html)

### removeField

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

**Description copied from interface:** [**Descriptor**](http://docs.google.com/javax/management/Descriptor.html#removeField(java.lang.String)) Removes a field from the descriptor.

**Specified by:**[removeField](http://docs.google.com/javax/management/Descriptor.html#removeField(java.lang.String)) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Parameters:**fieldName - String name of the field to be removed. If the field name is illegal or the field is not found, no exception is thrown.

### equals

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

Compares this descriptor to the given object. The objects are equal if the given object is also a Descriptor, and if the two Descriptors have the same field names (possibly differing in case) and the same associated values. The respective values for a field in the two Descriptors are equal if the following conditions hold:

* If one value is null then the other must be too.
* If one value is a primitive array then the other must be a primitive array of the same type with the same elements.
* If one value is an object array then the other must be too and [Arrays.deepEquals](http://docs.google.com/java/util/Arrays.html#deepEquals(java.lang.Object%5B%5D,%20java.lang.Object%5B%5D)) must return true.
* Otherwise [Object.equals(Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) must return true.

**Specified by:**[equals](http://docs.google.com/javax/management/Descriptor.html#equals(java.lang.Object)) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html)**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:**o - the object to compare with. **Returns:**true if the objects are the same; false otherwise.**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### hashCode

public int **hashCode**()

Returns the hash code value for this descriptor. The hash code is computed as the sum of the hash codes for each field in the descriptor. The hash code of a field with name n and value v is n.toLowerCase().hashCode() ^ h. Here h is the hash code of v, computed as follows:

* If v is null then h is 0.
* If v is a primitive array then h is computed using the appropriate overloading of java.util.Arrays.hashCode.
* If v is an object array then h is computed using [Arrays.deepHashCode](http://docs.google.com/java/util/Arrays.html#deepHashCode(java.lang.Object%5B%5D)).
* Otherwise h is v.hashCode().

**Specified by:**[hashCode](http://docs.google.com/javax/management/Descriptor.html#hashCode()) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html)**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**A hash code value for this object.**See Also:**[Object.equals(java.lang.Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### isValid

public boolean **isValid**()  
 throws [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Returns true if all of the fields have legal values given their names.

This implementation does not support interoperating with a directory or lookup service. Thus, conforming to the specification, no checking is done on the *"export"* field.

Otherwise this implementation returns false if:

* name and descriptorType fieldNames are not defined, or null, or empty, or not String
* class, role, getMethod, setMethod fieldNames, if defined, are null or not String
* persistPeriod, currencyTimeLimit, lastUpdatedTimeStamp, lastReturnedTimeStamp if defined, are null, or not a Numeric String or not a Numeric Value >= -1
* log fieldName, if defined, is null, or not a Boolean or not a String with value "t", "f", "true", "false". These String values must not be case sensitive.
* visibility fieldName, if defined, is null, or not a Numeric String or a not Numeric Value >= 1 and <= 4
* severity fieldName, if defined, is null, or not a Numeric String or not a Numeric Value >= 0 and <= 6
* persistPolicy fieldName, if defined, is null, or not one of the following strings:  
  "OnUpdate", "OnTimer", "NoMoreOftenThan", "OnUnregister", "Always", "Never". These String values must not be case sensitive.

**Specified by:**[isValid](http://docs.google.com/javax/management/Descriptor.html#isValid()) in interface [Descriptor](http://docs.google.com/javax/management/Descriptor.html) **Returns:**true if the values are legal. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - If the validity checking fails for any reason, this exception will be thrown.

### toXMLString

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

Returns an XML String representing the descriptor.

The format is not defined, but an implementation must ensure that the string returned by this method can be used to build an equivalent descriptor when instantiated using the constructor [DescriptorSupport(String inStr)](http://docs.google.com/javax/management/modelmbean/DescriptorSupport.html#DescriptorSupport(java.lang.String)).

Fields which are not String objects will have toString() called on them to create the value. The value will be enclosed in parentheses. It is not guaranteed that you can reconstruct these objects unless they have been specifically set up to support toString() in a meaningful format and have a matching constructor that accepts a String in the same format.

If the descriptor is empty the following String is returned: <Descriptor></Descriptor>

**Returns:**the XML string. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - for illegal value for field Names or field Values. If the XML formatted string construction fails for any reason, this exception will be thrown.

### toString

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

Returns a human readable string representing the descriptor. The string will be in the format of "fieldName=fieldValue,fieldName2=fieldValue2,..."

If there are no fields in the descriptor, then an empty String is returned.

If a fieldValue is an object then the toString() method is called on it and its returned value is used as the value for the field enclosed in parenthesis.

**Overrides:**[toString](http://docs.google.com/java/lang/Object.html#toString()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a string representation of the object. **Throws:** [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - for illegal value for field Names or field Values. If the descriptor string fails for any reason, this exception will be thrown.

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