| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CompositeData.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/management/openmbean/ArrayType.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/openmbean/CompositeDataInvocationHandler.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/openmbean/CompositeData.html)    [**NO FRAMES**](http://docs.google.com/CompositeData.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **javax.management.openmbean**

Interface CompositeData

**All Known Implementing Classes:** [CompositeDataSupport](http://docs.google.com/javax/management/openmbean/CompositeDataSupport.html)

public interface **CompositeData**

The CompositeData interface specifies the behavior of a specific type of complex *open data* objects which represent *composite data* structures.

**Since:** 1.5

| **Method Summary** | |
| --- | --- |
| boolean | [**containsKey**](http://docs.google.com/javax/management/openmbean/CompositeData.html#containsKey(java.lang.String))([String](http://docs.google.com/java/lang/String.html) key)            Returns true if and only if this CompositeData instance contains an item whose name is key. |
| boolean | [**containsValue**](http://docs.google.com/javax/management/openmbean/CompositeData.html#containsValue(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) value)            Returns true if and only if this CompositeData instance contains an item whose value is value. |
| boolean | [**equals**](http://docs.google.com/javax/management/openmbean/CompositeData.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Compares the specified obj parameter with this CompositeData instance for equality. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**get**](http://docs.google.com/javax/management/openmbean/CompositeData.html#get(java.lang.String))([String](http://docs.google.com/java/lang/String.html) key)            Returns the value of the item whose name is key. |
| [Object](http://docs.google.com/java/lang/Object.html)[] | [**getAll**](http://docs.google.com/javax/management/openmbean/CompositeData.html#getAll(java.lang.String%5B%5D))([String](http://docs.google.com/java/lang/String.html)[] keys)            Returns an array of the values of the items whose names are specified by keys, in the same order as keys. |
| [CompositeType](http://docs.google.com/javax/management/openmbean/CompositeType.html) | [**getCompositeType**](http://docs.google.com/javax/management/openmbean/CompositeData.html#getCompositeType())()            Returns the *composite type*  of this *composite data* instance. |
| int | [**hashCode**](http://docs.google.com/javax/management/openmbean/CompositeData.html#hashCode())()            Returns the hash code value for this CompositeData instance. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/management/openmbean/CompositeData.html#toString())()            Returns a string representation of this CompositeData instance. |
| [Collection](http://docs.google.com/java/util/Collection.html)<?> | [**values**](http://docs.google.com/javax/management/openmbean/CompositeData.html#values())()            Returns an unmodifiable Collection view of the item values contained in this CompositeData instance. |

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

### getCompositeType

[CompositeType](http://docs.google.com/javax/management/openmbean/CompositeType.html) **getCompositeType**()

Returns the *composite type*  of this *composite data* instance.

**Returns:**the type of this CompositeData.

### get

[Object](http://docs.google.com/java/lang/Object.html) **get**([String](http://docs.google.com/java/lang/String.html) key)

Returns the value of the item whose name is key.

**Parameters:**key - the name of the item. **Returns:**the value associated with this key. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if key is a null or empty String. [InvalidKeyException](http://docs.google.com/javax/management/openmbean/InvalidKeyException.html) - if key is not an existing item name for this CompositeData instance.

### getAll

[Object](http://docs.google.com/java/lang/Object.html)[] **getAll**([String](http://docs.google.com/java/lang/String.html)[] keys)

Returns an array of the values of the items whose names are specified by keys, in the same order as keys.

**Parameters:**keys - the names of the items. **Returns:**the values corresponding to the keys. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if an element in keys is a null or empty String. [InvalidKeyException](http://docs.google.com/javax/management/openmbean/InvalidKeyException.html) - if an element in keys is not an existing item name for this CompositeData instance.

### containsKey

boolean **containsKey**([String](http://docs.google.com/java/lang/String.html) key)

Returns true if and only if this CompositeData instance contains an item whose name is key. If key is a null or empty String, this method simply returns false.

**Parameters:**key - the key to be tested. **Returns:**true if this CompositeData contains the key.

### containsValue

boolean **containsValue**([Object](http://docs.google.com/java/lang/Object.html) value)

Returns true if and only if this CompositeData instance contains an item whose value is value.

**Parameters:**value - the value to be tested. **Returns:**true if this CompositeData contains the value.

### values

[Collection](http://docs.google.com/java/util/Collection.html)<?> **values**()

Returns an unmodifiable Collection view of the item values contained in this CompositeData instance. The returned collection's iterator will return the values in the ascending lexicographic order of the corresponding item names.

**Returns:**the values.

### equals

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

Compares the specified obj parameter with this CompositeData instance for equality.

Returns true if and only if all of the following statements are true:

* obj is non null,
* obj also implements the CompositeData interface,
* their composite types are equal
* their contents, i.e. (name, value) pairs are equal. If a value contained in the content is an array, the value comparison is done as if by calling the [deepEquals](http://docs.google.com/java/util/Arrays.html#deepEquals(java.lang.Object%5B%5D,%20java.lang.Object%5B%5D)) method for arrays of object reference types or the appropriate overloading of Arrays.equals(e1,e2) for arrays of primitive types

This ensures that this equals method works properly for obj parameters which are different implementations of the CompositeData interface, with the restrictions mentioned in the [equals](http://docs.google.com/java/util/Collection.html#equals(java.lang.Object)) method of the java.util.Collection interface.

**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 - the object to be compared for equality with this CompositeData instance. **Returns:**true if the specified object is equal to this CompositeData instance.**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### hashCode

int **hashCode**()

Returns the hash code value for this CompositeData instance.

The hash code of a CompositeData instance is the sum of the hash codes of all elements of information used in equals comparisons (ie: its *composite type* and all the item values).

This ensures that t1.equals(t2) implies that t1.hashCode()==t2.hashCode() for any two CompositeData instances t1 and t2, as required by the general contract of the method [Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()).

Each item value's hash code is added to the returned hash code. If an item value is an array, its hash code is obtained as if by calling the [deepHashCode](http://docs.google.com/java/util/Arrays.html#deepHashCode(java.lang.Object%5B%5D)) method for arrays of object reference types or the appropriate overloading of Arrays.hashCode(e) for arrays of primitive types.

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**the hash code value for this CompositeData instance**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)

### toString

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

Returns a string representation of this CompositeData instance.

The string representation consists of the name of the implementing class, the string representation of the composite type of this instance, and the string representation of the contents (ie list the itemName=itemValue mappings).

**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 this CompositeData instance

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CompositeData.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/management/openmbean/ArrayType.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/openmbean/CompositeDataInvocationHandler.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/openmbean/CompositeData.html)    [**NO FRAMES**](http://docs.google.com/CompositeData.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).