| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MultipleDocumentHandling.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/print/attribute/standard/MediaTray.html)   [**NEXT CLASS**](http://docs.google.com/javax/print/attribute/standard/NumberOfDocuments.html) | [**FRAMES**](http://docs.google.com/index.html?javax/print/attribute/standard/MultipleDocumentHandling.html)    [**NO FRAMES**](http://docs.google.com/MultipleDocumentHandling.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#1t3h5sf) | [CONSTR](#4d34og8) | [METHOD](#2s8eyo1) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#2jxsxqh) | [METHOD](#3j2qqm3) |

## **javax.print.attribute.standard**

Class MultipleDocumentHandling

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.print.attribute.EnumSyntax](http://docs.google.com/javax/print/attribute/EnumSyntax.html)  
 **javax.print.attribute.standard.MultipleDocumentHandling**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html), [Attribute](http://docs.google.com/javax/print/attribute/Attribute.html), [PrintJobAttribute](http://docs.google.com/javax/print/attribute/PrintJobAttribute.html), [PrintRequestAttribute](http://docs.google.com/javax/print/attribute/PrintRequestAttribute.html)

public class **MultipleDocumentHandling**extends [EnumSyntax](http://docs.google.com/javax/print/attribute/EnumSyntax.html)implements [PrintRequestAttribute](http://docs.google.com/javax/print/attribute/PrintRequestAttribute.html), [PrintJobAttribute](http://docs.google.com/javax/print/attribute/PrintJobAttribute.html)

Class MultipleDocumentHandling is a printing attribute class, an enumeration, that controls finishing operations and the placement of one or more print-stream pages into impressions and onto media sheets. When the value of the [Copies](http://docs.google.com/javax/print/attribute/standard/Copies.html) attribute exceeds 1, MultipleDocumentHandling also controls the order in which the copies that result from processing the documents are produced. This attribute is relevant only for a multidoc print job consisting of two or more individual docs.

Briefly, MultipleDocumentHandling determines the relationship between the multiple input (electronic) documents fed into a multidoc print job and the output (physical) document or documents produced by the multidoc print job. There are two possibilities:

* The multiple input documents are combined into a single output document. Finishing operations ([Finishings](http://docs.google.com/javax/print/attribute/standard/Finishings.html)), are performed on this single output document. The [Copies](http://docs.google.com/javax/print/attribute/standard/Copies.html) attribute tells how many copies of this single output document to produce. The MultipleDocumentHandling values SINGLE\_DOCUMENT and SINGLE\_DOCUMENT\_NEW\_SHEET specify two variations of this possibility.
* The multiple input documents remain separate output documents. Finishing operations ([Finishings](http://docs.google.com/javax/print/attribute/standard/Finishings.html)), are performed on each output document separately. The [Copies](http://docs.google.com/javax/print/attribute/standard/Copies.html) attribute tells how many copies of each separate output document to produce. The MultipleDocumentHandling values SEPARATE\_DOCUMENTS\_UNCOLLATED\_COPIES and SEPARATE\_DOCUMENTS\_COLLATED\_COPIES specify two variations of this possibility.

In the detailed explanations below, if "a" represents an instance of document data, then the result of processing the data in document "a" is a sequence of media sheets represented by "a(\*)".

The standard MultipleDocumentHandling values are:

* [**SINGLE\_DOCUMENT**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SINGLE_DOCUMENT). If a print job has multiple documents -- say, the document data is called a and b -- then the result of processing all the document data (a and then b) must be treated as a single sequence of media sheets for finishing operations; that is, finishing would be performed on the concatenation of the sequences a(\*),b(\*). The printer must not force the data in each document instance to be formatted onto a new print-stream page, nor to start a new impression on a new media sheet. If more than one copy is made, the ordering of the sets of media sheets resulting from processing the document data must be a(\*),b(\*),a(\*),b(\*),..., and the printer object must force each copy (a(\*),b(\*)) to start on a new media sheet.

* [**SEPARATE\_DOCUMENTS\_UNCOLLATED\_COPIES**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SEPARATE_DOCUMENTS_UNCOLLATED_COPIES). If a print job has multiple documents -- say, the document data is called a and b -- then the result of processing the data in each document instance must be treated as a single sequence of media sheets for finishing operations; that is, the sets a(\*) and b(\*) would each be finished separately. The printer must force each copy of the result of processing the data in a single document to start on a new media sheet. If more than one copy is made, the ordering of the sets of media sheets resulting from processing the document data must be a(\*),a(\*),...,b(\*),b(\*)....

* [**SEPARATE\_DOCUMENTS\_COLLATED\_COPIES**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SEPARATE_DOCUMENTS_COLLATED_COPIES). If a print job has multiple documents -- say, the document data is called a and b -- then the result of processing the data in each document instance must be treated as a single sequence of media sheets for finishing operations; that is, the sets a(\*) and b(\*) would each be finished separately. The printer must force each copy of the result of processing the data in a single document to start on a new media sheet. If more than one copy is made, the ordering of the sets of media sheets resulting from processing the document data must be a(\*),b(\*),a(\*),b(\*),....

* [**SINGLE\_DOCUMENT\_NEW\_SHEET**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SINGLE_DOCUMENT_NEW_SHEET). Same as SINGLE\_DOCUMENT, except that the printer must ensure that the first impression of each document instance in the job is placed on a new media sheet. This value allows multiple documents to be stapled together with a single staple where each document starts on a new sheet.

SINGLE\_DOCUMENT is the same as SEPARATE\_DOCUMENTS\_COLLATED\_COPIES with respect to ordering of print-stream pages, but not media sheet generation, since SINGLE\_DOCUMENT will put the first page of the next document on the back side of a sheet if an odd number of pages have been produced so far for the job, while SEPARATE\_DOCUMENTS\_COLLATED\_COPIES always forces the next document or document copy on to a new sheet.

In addition, if a [Finishings](http://docs.google.com/javax/print/attribute/standard/Finishings.html) attribute of [STAPLE](http://docs.google.com/javax/print/attribute/standard/Finishings.html#STAPLE) is specified, then:

* With SINGLE\_DOCUMENT, documents a and b are stapled together as a single document with no regard to new sheets.
* With SINGLE\_DOCUMENT\_NEW\_SHEET, documents a and b are stapled together as a single document, but document b starts on a new sheet.
* With SEPARATE\_DOCUMENTS\_UNCOLLATED\_COPIES and SEPARATE\_DOCUMENTS\_COLLATED\_COPIES, documents a and b are stapled separately.

*Note:* None of these values provide means to produce uncollated sheets within a document, i.e., where multiple copies of sheet *n* are produced before sheet *n*+1 of the same document. To specify that, see the [SheetCollate](http://docs.google.com/javax/print/attribute/standard/SheetCollate.html) attribute.

**IPP Compatibility:** The category name returned by getName() is the IPP attribute name. The enumeration's integer value is the IPP enum value. The toString() method returns the IPP string representation of the attribute value.

**See Also:**[Copies](http://docs.google.com/javax/print/attribute/standard/Copies.html), [Finishings](http://docs.google.com/javax/print/attribute/standard/Finishings.html), [NumberUp](http://docs.google.com/javax/print/attribute/standard/NumberUp.html), [PageRanges](http://docs.google.com/javax/print/attribute/standard/PageRanges.html), [SheetCollate](http://docs.google.com/javax/print/attribute/standard/SheetCollate.html), [Sides](http://docs.google.com/javax/print/attribute/standard/Sides.html), [Serialized Form](http://docs.google.com/serialized-form.html#javax.print.attribute.standard.MultipleDocumentHandling)

| **Field Summary** | |
| --- | --- |
| static [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) | [**SEPARATE\_DOCUMENTS\_COLLATED\_COPIES**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SEPARATE_DOCUMENTS_COLLATED_COPIES)            Separate documents collated copies -- see above for [further information](#tyjcwt). |
| static [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) | [**SEPARATE\_DOCUMENTS\_UNCOLLATED\_COPIES**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SEPARATE_DOCUMENTS_UNCOLLATED_COPIES)            Separate documents uncollated copies -- see above for [further information](#2et92p0). |
| static [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) | [**SINGLE\_DOCUMENT**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SINGLE_DOCUMENT)            Single document -- see above for [further information](#3znysh7). |
| static [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) | [**SINGLE\_DOCUMENT\_NEW\_SHEET**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#SINGLE_DOCUMENT_NEW_SHEET)            Single document new sheet -- see above for [further information](#3dy6vkm). |

| **Constructor Summary** | |
| --- | --- |
| protected | [**MultipleDocumentHandling**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#MultipleDocumentHandling(int))(int value)            Construct a new multiple document handling enumeration value with the given integer value. |

| **Method Summary** | |
| --- | --- |
| [Class](http://docs.google.com/java/lang/Class.html)<? extends [Attribute](http://docs.google.com/javax/print/attribute/Attribute.html)> | [**getCategory**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#getCategory())()            Get the printing attribute class which is to be used as the "category" for this printing attribute value. |
| protected  [EnumSyntax](http://docs.google.com/javax/print/attribute/EnumSyntax.html)[] | [**getEnumValueTable**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#getEnumValueTable())()            Returns the enumeration value table for class MultipleDocumentHandling. |
| [String](http://docs.google.com/java/lang/String.html) | [**getName**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#getName())()            Get the name of the category of which this attribute value is an instance. |
| protected  [String](http://docs.google.com/java/lang/String.html)[] | [**getStringTable**](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html#getStringTable())()            Returns the string table for class MultipleDocumentHandling. |

| **Methods inherited from class javax.print.attribute.**[**EnumSyntax**](http://docs.google.com/javax/print/attribute/EnumSyntax.html) |
| --- |
| [clone](http://docs.google.com/javax/print/attribute/EnumSyntax.html#clone()), [getOffset](http://docs.google.com/javax/print/attribute/EnumSyntax.html#getOffset()), [getValue](http://docs.google.com/javax/print/attribute/EnumSyntax.html#getValue()), [hashCode](http://docs.google.com/javax/print/attribute/EnumSyntax.html#hashCode()), [readResolve](http://docs.google.com/javax/print/attribute/EnumSyntax.html#readResolve()), [toString](http://docs.google.com/javax/print/attribute/EnumSyntax.html#toString()) |

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

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

### SINGLE\_DOCUMENT

public static final [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) **SINGLE\_DOCUMENT**

Single document -- see above for [further information](#3znysh7).

### SEPARATE\_DOCUMENTS\_UNCOLLATED\_COPIES

public static final [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) **SEPARATE\_DOCUMENTS\_UNCOLLATED\_COPIES**

Separate documents uncollated copies -- see above for [further information](#2et92p0).

### SEPARATE\_DOCUMENTS\_COLLATED\_COPIES

public static final [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) **SEPARATE\_DOCUMENTS\_COLLATED\_COPIES**

Separate documents collated copies -- see above for [further information](#tyjcwt).

### SINGLE\_DOCUMENT\_NEW\_SHEET

public static final [MultipleDocumentHandling](http://docs.google.com/javax/print/attribute/standard/MultipleDocumentHandling.html) **SINGLE\_DOCUMENT\_NEW\_SHEET**

Single document new sheet -- see above for [further information](#3dy6vkm).

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

### MultipleDocumentHandling

protected **MultipleDocumentHandling**(int value)

Construct a new multiple document handling enumeration value with the given integer value.

**Parameters:**value - Integer value.

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

### getStringTable

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

Returns the string table for class MultipleDocumentHandling.

**Overrides:**[getStringTable](http://docs.google.com/javax/print/attribute/EnumSyntax.html#getStringTable()) in class [EnumSyntax](http://docs.google.com/javax/print/attribute/EnumSyntax.html) **Returns:**the string table

### getEnumValueTable

protected [EnumSyntax](http://docs.google.com/javax/print/attribute/EnumSyntax.html)[] **getEnumValueTable**()

Returns the enumeration value table for class MultipleDocumentHandling.

**Overrides:**[getEnumValueTable](http://docs.google.com/javax/print/attribute/EnumSyntax.html#getEnumValueTable()) in class [EnumSyntax](http://docs.google.com/javax/print/attribute/EnumSyntax.html) **Returns:**the value table

### getCategory

public final [Class](http://docs.google.com/java/lang/Class.html)<? extends [Attribute](http://docs.google.com/javax/print/attribute/Attribute.html)> **getCategory**()

Get the printing attribute class which is to be used as the "category" for this printing attribute value.

For class MultipleDocumentHandling and any vendor-defined subclasses, the category is class MultipleDocumentHandling itself.

**Specified by:**[getCategory](http://docs.google.com/javax/print/attribute/Attribute.html#getCategory()) in interface [Attribute](http://docs.google.com/javax/print/attribute/Attribute.html) **Returns:**Printing attribute class (category), an instance of class [java.lang.Class](http://docs.google.com/java/lang/Class.html).

### getName

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

Get the name of the category of which this attribute value is an instance.

For class MultipleDocumentHandling and any vendor-defined subclasses, the category name is "multiple-document-handling".

**Specified by:**[getName](http://docs.google.com/javax/print/attribute/Attribute.html#getName()) in interface [Attribute](http://docs.google.com/javax/print/attribute/Attribute.html) **Returns:**Attribute category name.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MultipleDocumentHandling.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/print/attribute/standard/MediaTray.html)   [**NEXT CLASS**](http://docs.google.com/javax/print/attribute/standard/NumberOfDocuments.html) | [**FRAMES**](http://docs.google.com/index.html?javax/print/attribute/standard/MultipleDocumentHandling.html)    [**NO FRAMES**](http://docs.google.com/MultipleDocumentHandling.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#1t3h5sf) | [CONSTR](#4d34og8) | [METHOD](#2s8eyo1) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#2jxsxqh) | [METHOD](#3j2qqm3) |

[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).