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

## **javax.print**

Interface Doc

**All Known Implementing Classes:** [SimpleDoc](http://docs.google.com/javax/print/SimpleDoc.html)

public interface **Doc**

Interface Doc specifies the interface for an object that supplies one piece of print data for a Print Job. "Doc" is a short, easy-to-pronounce term that means "a piece of print data." The client passes to the Print Job an object that implements interface Doc, and the Print Job calls methods on that object to obtain the print data. The Doc interface lets a Print Job:

* Determine the format, or "doc flavor" (class [DocFlavor](http://docs.google.com/javax/print/DocFlavor.html)), in which the print data is available. A doc flavor designates the print data format (a MIME type) and the representation class of the object from which the print data comes.
* Obtain the print data representation object, which is an instance of the doc flavor's representation class. The Print Job can then obtain the actual print data from the representation object.
* Obtain the printing attributes that specify additional characteristics of the doc or that specify processing instructions to be applied to the doc. Printing attributes are defined in package [javax.print.attribute](http://docs.google.com/javax/print/attribute/package-summary.html). The doc returns its printing attributes stored in an [javax.print.attribute.DocAttributeSet](http://docs.google.com/javax/print/attribute/DocAttributeSet.html).

Each method in an implementation of interface Doc is permitted always to return the same object each time the method is called. This has implications for a Print Job or other caller of a doc object whose print data representation object "consumes" the print data as the caller obtains the print data, such as a print data representation object which is a stream. Once the Print Job has called [getPrintData()](http://docs.google.com/javax/print/Doc.html#getPrintData()) and obtained the stream, any further calls to [getPrintData()](http://docs.google.com/javax/print/Doc.html#getPrintData()) will return the same stream object upon which reading may already be in progress, *not* a new stream object that will re-read the print data from the beginning. Specifying a doc object to behave this way simplifies the implementation of doc objects, and is justified on the grounds that a particular doc is intended to convey print data only to one Print Job, not to several different Print Jobs. (To convey the same print data to several different Print Jobs, you have to create several different doc objects on top of the same print data source.)

Interface Doc affords considerable implementation flexibility. The print data might already be in existence when the doc object is constructed. In this case the objects returned by the doc's methods can be supplied to the doc's constructor, be stored in the doc ahead of time, and simply be returned when called for. Alternatively, the print data might not exist yet when the doc object is constructed. In this case the doc object might provide a "lazy" implementation that generates the print data representation object (and/or the print data) only when the Print Job calls for it (when the Print Job calls the [getPrintData()](http://docs.google.com/javax/print/Doc.html#getPrintData()) method).

There is no restriction on the number of client threads that may be simultaneously accessing the same doc. Therefore, all implementations of interface Doc must be designed to be multiple thread safe.

However there can only be one consumer of the print data obtained from a Doc.

If print data is obtained from the client as a stream, by calling Doc's getReaderForText() or getStreamForBytes() methods, or because the print data source is already an InputStream or Reader, then the print service should always close these streams for the client on all job completion conditions. With the following caveat. If the print data is itself a stream, the service will always close it. If the print data is otherwise something that can be requested as a stream, the service will only close the stream if it has obtained the stream before terminating. That is, just because a print service might request data as a stream does not mean that it will, with the implications that Doc implementors which rely on the service to close them should create such streams only in response to a request from the service.

| **Method Summary** | |
| --- | --- |
| [DocAttributeSet](http://docs.google.com/javax/print/attribute/DocAttributeSet.html) | [**getAttributes**](http://docs.google.com/javax/print/Doc.html#getAttributes())()            Obtains the set of printing attributes for this doc object. |
| [DocFlavor](http://docs.google.com/javax/print/DocFlavor.html) | [**getDocFlavor**](http://docs.google.com/javax/print/Doc.html#getDocFlavor())()            Determines the doc flavor in which this doc object will supply its piece of print data. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getPrintData**](http://docs.google.com/javax/print/Doc.html#getPrintData())()            Obtains the print data representation object that contains this doc object's piece of print data in the format corresponding to the supported doc flavor. |
| [Reader](http://docs.google.com/java/io/Reader.html) | [**getReaderForText**](http://docs.google.com/javax/print/Doc.html#getReaderForText())()            Obtains a reader for extracting character print data from this doc. |
| [InputStream](http://docs.google.com/java/io/InputStream.html) | [**getStreamForBytes**](http://docs.google.com/javax/print/Doc.html#getStreamForBytes())()            Obtains an input stream for extracting byte print data from this doc. |

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

### getDocFlavor

[DocFlavor](http://docs.google.com/javax/print/DocFlavor.html) **getDocFlavor**()

Determines the doc flavor in which this doc object will supply its piece of print data.

**Returns:**Doc flavor.

### getPrintData

[Object](http://docs.google.com/java/lang/Object.html) **getPrintData**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Obtains the print data representation object that contains this doc object's piece of print data in the format corresponding to the supported doc flavor. The getPrintData() method returns an instance of the representation class whose name is given by [getDocFlavor()](http://docs.google.com/javax/print/Doc.html#getDocFlavor()).[getRepresentationClassName()](http://docs.google.com/javax/print/DocFlavor.html#getRepresentationClassName()), and the return value can be cast from class Object to that representation class.

**Returns:**Print data representation object. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - Thrown if the representation class is a stream and there was an I/O error while constructing the stream.

### getAttributes

[DocAttributeSet](http://docs.google.com/javax/print/attribute/DocAttributeSet.html) **getAttributes**()

Obtains the set of printing attributes for this doc object. If the returned attribute set includes an instance of a particular attribute *X,* the printer must use that attribute value for this doc, overriding any value of attribute *X* in the job's attribute set. If the returned attribute set does not include an instance of a particular attribute *X* or if null is returned, the printer must consult the job's attribute set to obtain the value for attribute *X,* and if not found there, the printer must use an implementation-dependent default value. The returned attribute set is unmodifiable.

**Returns:**Unmodifiable set of printing attributes for this doc, or null to obtain all attribute values from the job's attribute set.

### getReaderForText

[Reader](http://docs.google.com/java/io/Reader.html) **getReaderForText**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Obtains a reader for extracting character print data from this doc. The Doc implementation is required to support this method if the DocFlavor has one of the following print data representation classes, and return null otherwise:

* char[]
* java.lang.String
* java.io.Reader

The doc's print data representation object is used to construct and return a Reader for reading the print data as a stream of characters from the print data representation object. However, if the print data representation object is itself a Reader, then the print data representation object is simply returned.

**Returns:**Reader for reading the print data characters from this doc. If a reader cannot be provided because this doc does not meet the criteria stated above, null is returned. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - Thrown if there was an I/O error while creating the reader.

### getStreamForBytes

[InputStream](http://docs.google.com/java/io/InputStream.html) **getStreamForBytes**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Obtains an input stream for extracting byte print data from this doc. The Doc implementation is required to support this method if the DocFlavor has one of the following print data representation classes, and return null otherwise:

* byte[]
* java.io.InputStream

This doc's print data representation object is obtained, then an input stream for reading the print data from the print data representation object as a stream of bytes is created and returned. However, if the print data representation object is itself an input stream, then the print data representation object is simply returned.

**Returns:**Input stream for reading the print data bytes from this doc. If an input stream cannot be provided because this doc does not meet the criteria stated above, null is returned. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - Thrown if there was an I/O error while creating the input stream.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Doc.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/CancelablePrintJob.html)   [**NEXT CLASS**](http://docs.google.com/javax/print/DocFlavor.html) | [**FRAMES**](http://docs.google.com/index.html?javax/print/Doc.html)    [**NO FRAMES**](http://docs.google.com/Doc.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).