| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AudioFileWriter.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/sound/sampled/spi/AudioFileReader.html)   [**NEXT CLASS**](http://docs.google.com/javax/sound/sampled/spi/FormatConversionProvider.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sound/sampled/spi/AudioFileWriter.html)    [**NO FRAMES**](http://docs.google.com/AudioFileWriter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **javax.sound.sampled.spi**

Class AudioFileWriter

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.sound.sampled.spi.AudioFileWriter**

public abstract class **AudioFileWriter**extends [Object](http://docs.google.com/java/lang/Object.html)

Provider for audio file writing services. Classes providing concrete implementations can write one or more types of audio file from an audio stream.

**Since:** 1.3

| **Constructor Summary** | |
| --- | --- |
| [**AudioFileWriter**](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#AudioFileWriter())() |

| **Method Summary** | |
| --- | --- |
| abstract  [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html)[] | [**getAudioFileTypes**](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#getAudioFileTypes())()            Obtains the file types for which file writing support is provided by this audio file writer. |
| abstract  [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html)[] | [**getAudioFileTypes**](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#getAudioFileTypes(javax.sound.sampled.AudioInputStream))([AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream)            Obtains the file types that this audio file writer can write from the audio input stream specified. |
| boolean | [**isFileTypeSupported**](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type))([AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType)            Indicates whether file writing support for the specified file type is provided by this audio file writer. |
| boolean | [**isFileTypeSupported**](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type,%20javax.sound.sampled.AudioInputStream))([AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType, [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream)            Indicates whether an audio file of the type specified can be written from the audio input stream indicated. |
| abstract  int | [**write**](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#write(javax.sound.sampled.AudioInputStream,%20javax.sound.sampled.AudioFileFormat.Type,%20java.io.File))([AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream, [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType, [File](http://docs.google.com/java/io/File.html) out)            Writes a stream of bytes representing an audio file of the file format indicated to the external file provided. |
| abstract  int | [**write**](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#write(javax.sound.sampled.AudioInputStream,%20javax.sound.sampled.AudioFileFormat.Type,%20java.io.OutputStream))([AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream, [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType, [OutputStream](http://docs.google.com/java/io/OutputStream.html) out)            Writes a stream of bytes representing an audio file of the file type indicated to the output stream provided. |

| **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()), [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()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [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** |
| --- |

### AudioFileWriter

public **AudioFileWriter**()

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

### getAudioFileTypes

public abstract [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html)[] **getAudioFileTypes**()

Obtains the file types for which file writing support is provided by this audio file writer.

**Returns:**array of file types. If no file types are supported, an array of length 0 is returned.

### isFileTypeSupported

public boolean **isFileTypeSupported**([AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType)

Indicates whether file writing support for the specified file type is provided by this audio file writer.

**Parameters:**fileType - the file type for which write capabilities are queried **Returns:**true if the file type is supported, otherwise false

### getAudioFileTypes

public abstract [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html)[] **getAudioFileTypes**([AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream)

Obtains the file types that this audio file writer can write from the audio input stream specified.

**Parameters:**stream - the audio input stream for which audio file type support is queried **Returns:**array of file types. If no file types are supported, an array of length 0 is returned.

### isFileTypeSupported

public boolean **isFileTypeSupported**([AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType,  
 [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream)

Indicates whether an audio file of the type specified can be written from the audio input stream indicated.

**Parameters:**fileType - file type for which write capabilities are queriedstream - for which file writing support is queried **Returns:**true if the file type is supported for this audio input stream, otherwise false

### write

public abstract int **write**([AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream,  
 [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType,  
 [OutputStream](http://docs.google.com/java/io/OutputStream.html) out)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a stream of bytes representing an audio file of the file type indicated to the output stream provided. Some file types require that the length be written into the file header, and cannot be written from start to finish unless the length is known in advance. An attempt to write such a file type will fail with an IOException if the length in the audio file format is [AudioSystem.NOT\_SPECIFIED](http://docs.google.com/javax/sound/sampled/AudioSystem.html#NOT_SPECIFIED).

**Parameters:**stream - the audio input stream containing audio data to be written to the output streamfileType - file type to be written to the output streamout - stream to which the file data should be written **Returns:**the number of bytes written to the output stream **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O exception occurs [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the file type is not supported by the system**See Also:**[isFileTypeSupported(AudioFileFormat.Type, AudioInputStream)](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type,%20javax.sound.sampled.AudioInputStream)), [getAudioFileTypes()](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#getAudioFileTypes())

### write

public abstract int **write**([AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream,  
 [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html) fileType,  
 [File](http://docs.google.com/java/io/File.html) out)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a stream of bytes representing an audio file of the file format indicated to the external file provided.

**Parameters:**stream - the audio input stream containing audio data to be written to the filefileType - file type to be written to the fileout - external file to which the file data should be written **Returns:**the number of bytes written to the file **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O exception occurs [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the file format is not supported by the system**See Also:**[isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type)](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type)), [getAudioFileTypes()](http://docs.google.com/javax/sound/sampled/spi/AudioFileWriter.html#getAudioFileTypes())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AudioFileWriter.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/sound/sampled/spi/AudioFileReader.html)   [**NEXT CLASS**](http://docs.google.com/javax/sound/sampled/spi/FormatConversionProvider.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sound/sampled/spi/AudioFileWriter.html)    [**NO FRAMES**](http://docs.google.com/AudioFileWriter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).