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

## **javax.sound.sampled**

Class AudioSystem

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

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

The AudioSystem class acts as the entry point to the sampled-audio system resources. This class lets you query and access the mixers that are installed on the system. AudioSystem includes a number of methods for converting audio data between different formats, and for translating between audio files and streams. It also provides a method for obtaining a [Line](http://docs.google.com/javax/sound/sampled/Line.html) directly from the AudioSystem without dealing explicitly with mixers.

Properties can be used to specify the default mixer for specific line types. Both system properties and a properties file are considered. In the Sun reference implementation, the properties file is "lib/sound.properties" in the JRE directory. If a property exists both as a system property and in the properties file, the system property takes precedence. If none is specified, a suitable default is chosen among the available devices. The syntax of the properties file is specified in [Properties.load](http://docs.google.com/java/util/Properties.html#load(java.io.InputStream)). The following table lists the available property keys and which methods consider them:

| Property Key | Interface | Affected Method(s) |
| --- | --- | --- |
| javax.sound.sampled.Clip | [Clip](http://docs.google.com/javax/sound/sampled/Clip.html) | [getLine(javax.sound.sampled.Line.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getLine(javax.sound.sampled.Line.Info)), [getClip()](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getClip()) |
| javax.sound.sampled.Port | [Port](http://docs.google.com/javax/sound/sampled/Port.html) | [getLine(javax.sound.sampled.Line.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getLine(javax.sound.sampled.Line.Info)) |
| javax.sound.sampled.SourceDataLine | [SourceDataLine](http://docs.google.com/javax/sound/sampled/SourceDataLine.html) | [getLine(javax.sound.sampled.Line.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getLine(javax.sound.sampled.Line.Info)), [getSourceDataLine(javax.sound.sampled.AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getSourceDataLine(javax.sound.sampled.AudioFormat)) |
| javax.sound.sampled.TargetDataLine | [TargetDataLine](http://docs.google.com/javax/sound/sampled/TargetDataLine.html) | [getLine(javax.sound.sampled.Line.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getLine(javax.sound.sampled.Line.Info)), [getTargetDataLine(javax.sound.sampled.AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetDataLine(javax.sound.sampled.AudioFormat)) |

The property value consists of the provider class name and the mixer name, separated by the hash mark ("#"). The provider class name is the fully-qualified name of a concrete [mixer provider](http://docs.google.com/javax/sound/sampled/spi/MixerProvider.html) class. The mixer name is matched against the String returned by the getName method of Mixer.Info. Either the class name, or the mixer name may be omitted. If only the class name is specified, the trailing hash mark is optional.

If the provider class is specified, and it can be successully retrieved from the installed providers, the list of Mixer.Info objects is retrieved from the provider. Otherwise, or when these mixers do not provide a subsequent match, the list is retrieved from [getMixerInfo()](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getMixerInfo()) to contain all available Mixer.Info objects.

If a mixer name is specified, the resulting list of Mixer.Info objects is searched: the first one with a matching name, and whose Mixer provides the respective line interface, will be returned. If no matching Mixer.Info object is found, or the mixer name is not specified, the first mixer from the resulting list, which provides the respective line interface, will be returned. For example, the property javax.sound.sampled.Clip with a value "com.sun.media.sound.MixerProvider#SunClip" will have the following consequences when getLine is called requesting a Clip instance: if the class com.sun.media.sound.MixerProvider exists in the list of installed mixer providers, the first Clip from the first mixer with name "SunClip" will be returned. If it cannot be found, the first Clip from the first mixer of the specified provider will be returned, regardless of name. If there is none, the first Clip from the first Mixer with name "SunClip" in the list of all mixers (as returned by getMixerInfo) will be returned, or, if not found, the first Clip of the first Mixerthat can be found in the list of all mixers is returned. If that fails, too, an IllegalArgumentException is thrown.

**Since:** 1.3 **See Also:**[AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html), [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html), [Mixer](http://docs.google.com/javax/sound/sampled/Mixer.html), [Line](http://docs.google.com/javax/sound/sampled/Line.html), [Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html)

| **Field Summary** | |
| --- | --- |
| static int | [**NOT\_SPECIFIED**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#NOT_SPECIFIED)            An integer that stands for an unknown numeric value. |

| **Method Summary** | |
| --- | --- |
| static [AudioFileFormat](http://docs.google.com/javax/sound/sampled/AudioFileFormat.html) | [**getAudioFileFormat**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioFileFormat(java.io.File))([File](http://docs.google.com/java/io/File.html) file)            Obtains the audio file format of the specified File. |
| static [AudioFileFormat](http://docs.google.com/javax/sound/sampled/AudioFileFormat.html) | [**getAudioFileFormat**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioFileFormat(java.io.InputStream))([InputStream](http://docs.google.com/java/io/InputStream.html) stream)            Obtains the audio file format of the provided input stream. |
| static [AudioFileFormat](http://docs.google.com/javax/sound/sampled/AudioFileFormat.html) | [**getAudioFileFormat**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioFileFormat(java.net.URL))([URL](http://docs.google.com/java/net/URL.html) url)            Obtains the audio file format of the specified URL. |
| static [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html)[] | [**getAudioFileTypes**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioFileTypes())()            Obtains the file types for which file writing support is provided by the system. |
| static [AudioFileFormat.Type](http://docs.google.com/javax/sound/sampled/AudioFileFormat.Type.html)[] | [**getAudioFileTypes**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioFileTypes(javax.sound.sampled.AudioInputStream))([AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) stream)            Obtains the file types that the system can write from the audio input stream specified. |
| static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) | [**getAudioInputStream**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioInputStream(javax.sound.sampled.AudioFormat.Encoding,%20javax.sound.sampled.AudioInputStream))([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) targetEncoding, [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) sourceStream)            Obtains an audio input stream of the indicated encoding, by converting the provided audio input stream. |
| static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) | [**getAudioInputStream**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioInputStream(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.AudioInputStream))([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) targetFormat, [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) sourceStream)            Obtains an audio input stream of the indicated format, by converting the provided audio input stream. |
| static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) | [**getAudioInputStream**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioInputStream(java.io.File))([File](http://docs.google.com/java/io/File.html) file)            Obtains an audio input stream from the provided File. |
| static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) | [**getAudioInputStream**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioInputStream(java.io.InputStream))([InputStream](http://docs.google.com/java/io/InputStream.html) stream)            Obtains an audio input stream from the provided input stream. |
| static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) | [**getAudioInputStream**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioInputStream(java.net.URL))([URL](http://docs.google.com/java/net/URL.html) url)            Obtains an audio input stream from the URL provided. |
| static [Clip](http://docs.google.com/javax/sound/sampled/Clip.html) | [**getClip**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getClip())()            Obtains a clip that can be used for playing back an audio file or an audio stream. |
| static [Clip](http://docs.google.com/javax/sound/sampled/Clip.html) | [**getClip**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getClip(javax.sound.sampled.Mixer.Info))([Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) mixerInfo)            Obtains a clip from the specified mixer that can be used for playing back an audio file or an audio stream. |
| static [Line](http://docs.google.com/javax/sound/sampled/Line.html) | [**getLine**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getLine(javax.sound.sampled.Line.Info))([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)            Obtains a line that matches the description in the specified Line.Info object. |
| static [Mixer](http://docs.google.com/javax/sound/sampled/Mixer.html) | [**getMixer**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getMixer(javax.sound.sampled.Mixer.Info))([Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) info)            Obtains the requested audio mixer. |
| static [Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html)[] | [**getMixerInfo**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getMixerInfo())()            Obtains an array of mixer info objects that represents the set of audio mixers that are currently installed on the system. |
| static [SourceDataLine](http://docs.google.com/javax/sound/sampled/SourceDataLine.html) | [**getSourceDataLine**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getSourceDataLine(javax.sound.sampled.AudioFormat))([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format)            Obtains a source data line that can be used for playing back audio data in the format specified by the AudioFormat object. |
| static [SourceDataLine](http://docs.google.com/javax/sound/sampled/SourceDataLine.html) | [**getSourceDataLine**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getSourceDataLine(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.Mixer.Info))([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format, [Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) mixerinfo)            Obtains a source data line that can be used for playing back audio data in the format specified by the AudioFormat object, provided by the mixer specified by the Mixer.Info object. |
| static [Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html)[] | [**getSourceLineInfo**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getSourceLineInfo(javax.sound.sampled.Line.Info))([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)            Obtains information about all source lines of a particular type that are supported by the installed mixers. |
| static [TargetDataLine](http://docs.google.com/javax/sound/sampled/TargetDataLine.html) | [**getTargetDataLine**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetDataLine(javax.sound.sampled.AudioFormat))([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format)            Obtains a target data line that can be used for recording audio data in the format specified by the AudioFormat object. |
| static [TargetDataLine](http://docs.google.com/javax/sound/sampled/TargetDataLine.html) | [**getTargetDataLine**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetDataLine(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.Mixer.Info))([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format, [Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) mixerinfo)            Obtains a target data line that can be used for recording audio data in the format specified by the AudioFormat object, provided by the mixer specified by the Mixer.Info object. |
| static [AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html)[] | [**getTargetEncodings**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetEncodings(javax.sound.sampled.AudioFormat.Encoding))([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) sourceEncoding)            Obtains the encodings that the system can obtain from an audio input stream with the specified encoding using the set of installed format converters. |
| static [AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html)[] | [**getTargetEncodings**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetEncodings(javax.sound.sampled.AudioFormat))([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)            Obtains the encodings that the system can obtain from an audio input stream with the specified format using the set of installed format converters. |
| static [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html)[] | [**getTargetFormats**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetFormats(javax.sound.sampled.AudioFormat.Encoding,%20javax.sound.sampled.AudioFormat))([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) targetEncoding, [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)            Obtains the formats that have a particular encoding and that the system can obtain from a stream of the specified format using the set of installed format converters. |
| static [Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html)[] | [**getTargetLineInfo**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetLineInfo(javax.sound.sampled.Line.Info))([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)            Obtains information about all target lines of a particular type that are supported by the installed mixers. |
| static boolean | [**isConversionSupported**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#isConversionSupported(javax.sound.sampled.AudioFormat.Encoding,%20javax.sound.sampled.AudioFormat))([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) targetEncoding, [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)            Indicates whether an audio input stream of the specified encoding can be obtained from an audio input stream that has the specified format. |
| static boolean | [**isConversionSupported**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#isConversionSupported(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.AudioFormat))([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) targetFormat, [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)            Indicates whether an audio input stream of a specified format can be obtained from an audio input stream of another specified format. |
| static boolean | [**isFileTypeSupported**](http://docs.google.com/javax/sound/sampled/AudioSystem.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 the system. |
| static boolean | [**isFileTypeSupported**](http://docs.google.com/javax/sound/sampled/AudioSystem.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 specified file type can be written from the indicated audio input stream. |
| static boolean | [**isLineSupported**](http://docs.google.com/javax/sound/sampled/AudioSystem.html#isLineSupported(javax.sound.sampled.Line.Info))([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)            Indicates whether the system supports any lines that match the specified Line.Info object. |
| static int | [**write**](http://docs.google.com/javax/sound/sampled/AudioSystem.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 specified file type to the external file provided. |
| static int | [**write**](http://docs.google.com/javax/sound/sampled/AudioSystem.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 specified file type 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)) |

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

### NOT\_SPECIFIED

public static final int **NOT\_SPECIFIED**

An integer that stands for an unknown numeric value. This value is appropriate only for signed quantities that do not normally take negative values. Examples include file sizes, frame sizes, buffer sizes, and sample rates. A number of Java Sound constructors accept a value of NOT\_SPECIFIED for such parameters. Other methods may also accept or return this value, as documented.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.sound.sampled.AudioSystem.NOT_SPECIFIED)

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

### getMixerInfo

public static [Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html)[] **getMixerInfo**()

Obtains an array of mixer info objects that represents the set of audio mixers that are currently installed on the system.

**Returns:**an array of info objects for the currently installed mixers. If no mixers are available on the system, an array of length 0 is returned.**See Also:**[getMixer(javax.sound.sampled.Mixer.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getMixer(javax.sound.sampled.Mixer.Info))

### getMixer

public static [Mixer](http://docs.google.com/javax/sound/sampled/Mixer.html) **getMixer**([Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) info)

Obtains the requested audio mixer.

**Parameters:**info - a Mixer.Info object representing the desired mixer, or null for the system default mixer **Returns:**the requested mixer **Throws:** [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if the requested mixer is unavailable because of security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the info object does not represent a mixer installed on the system**See Also:**[getMixerInfo()](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getMixerInfo())

### getSourceLineInfo

public static [Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html)[] **getSourceLineInfo**([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)

Obtains information about all source lines of a particular type that are supported by the installed mixers.

**Parameters:**info - a Line.Info object that specifies the kind of lines about which information is requested **Returns:**an array of Line.Info objects describing source lines matching the type requested. If no matching source lines are supported, an array of length 0 is returned.**See Also:**[Mixer.getSourceLineInfo(Line.Info)](http://docs.google.com/javax/sound/sampled/Mixer.html#getSourceLineInfo(javax.sound.sampled.Line.Info))

### getTargetLineInfo

public static [Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html)[] **getTargetLineInfo**([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)

Obtains information about all target lines of a particular type that are supported by the installed mixers.

**Parameters:**info - a Line.Info object that specifies the kind of lines about which information is requested **Returns:**an array of Line.Info objects describing target lines matching the type requested. If no matching target lines are supported, an array of length 0 is returned.**See Also:**[Mixer.getTargetLineInfo(Line.Info)](http://docs.google.com/javax/sound/sampled/Mixer.html#getTargetLineInfo(javax.sound.sampled.Line.Info))

### isLineSupported

public static boolean **isLineSupported**([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)

Indicates whether the system supports any lines that match the specified Line.Info object. A line is supported if any installed mixer supports it.

**Parameters:**info - a Line.Info object describing the line for which support is queried **Returns:**true if at least one matching line is supported, otherwise false**See Also:**[Mixer.isLineSupported(Line.Info)](http://docs.google.com/javax/sound/sampled/Mixer.html#isLineSupported(javax.sound.sampled.Line.Info))

### getLine

public static [Line](http://docs.google.com/javax/sound/sampled/Line.html) **getLine**([Line.Info](http://docs.google.com/javax/sound/sampled/Line.Info.html) info)  
 throws [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html)

Obtains a line that matches the description in the specified Line.Info object.

If a DataLine is requested, and info is an instance of DataLine.Info specifying at least one fully qualified audio format, the last one will be used as the default format of the returned DataLine.

If system properties javax.sound.sampled.Clip, javax.sound.sampled.Port, javax.sound.sampled.SourceDataLine and javax.sound.sampled.TargetDataLine are defined or they are defined in the file "sound.properties", they are used to retrieve default lines. For details, refer to the [class description](http://docs.google.com/javax/sound/sampled/AudioSystem.html). If the respective property is not set, or the mixer requested in the property is not installed or does not provide the requested line, all installed mixers are queried for the requested line type. A Line will be returned from the first mixer providing the requested line type.

**Parameters:**info - a Line.Info object describing the desired kind of line **Returns:**a line of the requested kind **Throws:** [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html) - if a matching line is not available due to resource restrictions [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a matching line is not available due to security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the system does not support at least one line matching the specified Line.Info object through any installed mixer

### getClip

public static [Clip](http://docs.google.com/javax/sound/sampled/Clip.html) **getClip**()  
 throws [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html)

Obtains a clip that can be used for playing back an audio file or an audio stream. The returned clip will be provided by the default system mixer, or, if not possible, by any other mixer installed in the system that supports a Clip object.

The returned clip must be opened with the open(AudioFormat) or open(AudioInputStream) method.

This is a high-level method that uses getMixer and getLine internally.

If the system property javax.sound.sampled.Clip is defined or it is defined in the file "sound.properties", it is used to retrieve the default clip. For details, refer to the [class description](http://docs.google.com/javax/sound/sampled/AudioSystem.html).

**Returns:**the desired clip object **Throws:** [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html) - if a clip object is not available due to resource restrictions [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a clip object is not available due to security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the system does not support at least one clip instance through any installed mixer**Since:** 1.5 **See Also:**[getClip(Mixer.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getClip(javax.sound.sampled.Mixer.Info))

### getClip

public static [Clip](http://docs.google.com/javax/sound/sampled/Clip.html) **getClip**([Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) mixerInfo)  
 throws [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html)

Obtains a clip from the specified mixer that can be used for playing back an audio file or an audio stream.

The returned clip must be opened with the open(AudioFormat) or open(AudioInputStream) method.

This is a high-level method that uses getMixer and getLine internally.

**Parameters:**mixerInfo - a Mixer.Info object representing the desired mixer, or null for the system default mixer **Returns:**a clip object from the specified mixer **Throws:** [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html) - if a clip is not available from this mixer due to resource restrictions [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a clip is not available from this mixer due to security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the system does not support at least one clip through the specified mixer**Since:** 1.5 **See Also:**[getClip()](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getClip())

### getSourceDataLine

public static [SourceDataLine](http://docs.google.com/javax/sound/sampled/SourceDataLine.html) **getSourceDataLine**([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format)  
 throws [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html)

Obtains a source data line that can be used for playing back audio data in the format specified by the AudioFormat object. The returned line will be provided by the default system mixer, or, if not possible, by any other mixer installed in the system that supports a matching SourceDataLine object.

The returned line should be opened with the open(AudioFormat) or open(AudioFormat, int) method.

This is a high-level method that uses getMixer and getLine internally.

The returned SourceDataLine's default audio format will be initialized with format.

If the system property javax.sound.sampled.SourceDataLine is defined or it is defined in the file "sound.properties", it is used to retrieve the default source data line. For details, refer to the [class description](http://docs.google.com/javax/sound/sampled/AudioSystem.html).

**Parameters:**format - an AudioFormat object specifying the supported audio format of the returned line, or null for any audio format **Returns:**the desired SourceDataLine object **Throws:** [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html) - if a matching source data line is not available due to resource restrictions [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a matching source data line is not available due to security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the system does not support at least one source data line supporting the specified audio format through any installed mixer**Since:** 1.5 **See Also:**[getSourceDataLine(AudioFormat, Mixer.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getSourceDataLine(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.Mixer.Info))

### getSourceDataLine

public static [SourceDataLine](http://docs.google.com/javax/sound/sampled/SourceDataLine.html) **getSourceDataLine**([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format,  
 [Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) mixerinfo)  
 throws [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html)

Obtains a source data line that can be used for playing back audio data in the format specified by the AudioFormat object, provided by the mixer specified by the Mixer.Info object.

The returned line should be opened with the open(AudioFormat) or open(AudioFormat, int) method.

This is a high-level method that uses getMixer and getLine internally.

The returned SourceDataLine's default audio format will be initialized with format.

**Parameters:**format - an AudioFormat object specifying the supported audio format of the returned line, or null for any audio formatmixerinfo - a Mixer.Info object representing the desired mixer, or null for the system default mixer **Returns:**the desired SourceDataLine object **Throws:** [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html) - if a matching source data line is not available from the specified mixer due to resource restrictions [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a matching source data line is not available from the specified mixer due to security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the specified mixer does not support at least one source data line supporting the specified audio format**Since:** 1.5 **See Also:**[getSourceDataLine(AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getSourceDataLine(javax.sound.sampled.AudioFormat))

### getTargetDataLine

public static [TargetDataLine](http://docs.google.com/javax/sound/sampled/TargetDataLine.html) **getTargetDataLine**([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format)  
 throws [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html)

Obtains a target data line that can be used for recording audio data in the format specified by the AudioFormat object. The returned line will be provided by the default system mixer, or, if not possible, by any other mixer installed in the system that supports a matching TargetDataLine object.

The returned line should be opened with the open(AudioFormat) or open(AudioFormat, int) method.

This is a high-level method that uses getMixer and getLine internally.

The returned TargetDataLine's default audio format will be initialized with format.

**Parameters:**format - an AudioFormat object specifying the supported audio format of the returned line, or null for any audio format **Returns:**the desired TargetDataLine object **Throws:** [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html) - if a matching target data line is not available due to resource restrictions [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a matching target data line is not available due to security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the system does not support at least one target data line supporting the specified audio format through any installed mixer**Since:** 1.5 **See Also:**[getTargetDataLine(AudioFormat, Mixer.Info)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetDataLine(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.Mixer.Info)), [AudioPermission](http://docs.google.com/javax/sound/sampled/AudioPermission.html)

### getTargetDataLine

public static [TargetDataLine](http://docs.google.com/javax/sound/sampled/TargetDataLine.html) **getTargetDataLine**([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) format,  
 [Mixer.Info](http://docs.google.com/javax/sound/sampled/Mixer.Info.html) mixerinfo)  
 throws [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html)

Obtains a target data line that can be used for recording audio data in the format specified by the AudioFormat object, provided by the mixer specified by the Mixer.Info object.

The returned line should be opened with the open(AudioFormat) or open(AudioFormat, int) method.

This is a high-level method that uses getMixer and getLine internally.

The returned TargetDataLine's default audio format will be initialized with format.

If the system property javax.sound.sampled.TargetDataLine is defined or it is defined in the file "sound.properties", it is used to retrieve the default target data line. For details, refer to the [class description](http://docs.google.com/javax/sound/sampled/AudioSystem.html).

**Parameters:**format - an AudioFormat object specifying the supported audio format of the returned line, or null for any audio formatmixerinfo - a Mixer.Info object representing the desired mixer, or null for the system default mixer **Returns:**the desired TargetDataLine object **Throws:** [LineUnavailableException](http://docs.google.com/javax/sound/sampled/LineUnavailableException.html) - if a matching target data line is not available from the specified mixer due to resource restrictions [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a matching target data line is not available from the specified mixer due to security restrictions [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the specified mixer does not support at least one target data line supporting the specified audio format**Since:** 1.5 **See Also:**[getTargetDataLine(AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetDataLine(javax.sound.sampled.AudioFormat)), [AudioPermission](http://docs.google.com/javax/sound/sampled/AudioPermission.html)

### getTargetEncodings

public static [AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html)[] **getTargetEncodings**([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) sourceEncoding)

Obtains the encodings that the system can obtain from an audio input stream with the specified encoding using the set of installed format converters.

**Parameters:**sourceEncoding - the encoding for which conversion support is queried **Returns:**array of encodings. If sourceEncodingis not supported, an array of length 0 is returned. Otherwise, the array will have a length of at least 1, representing sourceEncoding (no conversion).

### getTargetEncodings

public static [AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html)[] **getTargetEncodings**([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)

Obtains the encodings that the system can obtain from an audio input stream with the specified format using the set of installed format converters.

**Parameters:**sourceFormat - the audio format for which conversion is queried **Returns:**array of encodings. If sourceFormatis not supported, an array of length 0 is returned. Otherwise, the array will have a length of at least 1, representing the encoding of sourceFormat (no conversion).

### isConversionSupported

public static boolean **isConversionSupported**([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) targetEncoding,  
 [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)

Indicates whether an audio input stream of the specified encoding can be obtained from an audio input stream that has the specified format.

**Parameters:**targetEncoding - the desired encoding after conversionsourceFormat - the audio format before conversion **Returns:**true if the conversion is supported, otherwise false

### getAudioInputStream

public static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) **getAudioInputStream**([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) targetEncoding,  
 [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) sourceStream)

Obtains an audio input stream of the indicated encoding, by converting the provided audio input stream.

**Parameters:**targetEncoding - the desired encoding after conversionsourceStream - the stream to be converted **Returns:**an audio input stream of the indicated encoding **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the conversion is not supported**See Also:**[getTargetEncodings(AudioFormat.Encoding)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetEncodings(javax.sound.sampled.AudioFormat.Encoding)), [getTargetEncodings(AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetEncodings(javax.sound.sampled.AudioFormat)), [isConversionSupported(AudioFormat.Encoding, AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#isConversionSupported(javax.sound.sampled.AudioFormat.Encoding,%20javax.sound.sampled.AudioFormat)), [getAudioInputStream(AudioFormat, AudioInputStream)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioInputStream(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.AudioInputStream))

### getTargetFormats

public static [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html)[] **getTargetFormats**([AudioFormat.Encoding](http://docs.google.com/javax/sound/sampled/AudioFormat.Encoding.html) targetEncoding,  
 [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)

Obtains the formats that have a particular encoding and that the system can obtain from a stream of the specified format using the set of installed format converters.

**Parameters:**targetEncoding - the desired encoding after conversionsourceFormat - the audio format before conversion **Returns:**array of formats. If no formats of the specified encoding are supported, an array of length 0 is returned.

### isConversionSupported

public static boolean **isConversionSupported**([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) targetFormat,  
 [AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) sourceFormat)

Indicates whether an audio input stream of a specified format can be obtained from an audio input stream of another specified format.

**Parameters:**targetFormat - the desired audio format after conversionsourceFormat - the audio format before conversion **Returns:**true if the conversion is supported, otherwise false

### getAudioInputStream

public static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) **getAudioInputStream**([AudioFormat](http://docs.google.com/javax/sound/sampled/AudioFormat.html) targetFormat,  
 [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) sourceStream)

Obtains an audio input stream of the indicated format, by converting the provided audio input stream.

**Parameters:**targetFormat - the desired audio format after conversionsourceStream - the stream to be converted **Returns:**an audio input stream of the indicated format **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the conversion is not supported #see #getTargetEncodings(AudioFormat)**See Also:**[getTargetFormats(AudioFormat.Encoding, AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getTargetFormats(javax.sound.sampled.AudioFormat.Encoding,%20javax.sound.sampled.AudioFormat)), [isConversionSupported(AudioFormat, AudioFormat)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#isConversionSupported(javax.sound.sampled.AudioFormat,%20javax.sound.sampled.AudioFormat)), [getAudioInputStream(AudioFormat.Encoding, AudioInputStream)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioInputStream(javax.sound.sampled.AudioFormat.Encoding,%20javax.sound.sampled.AudioInputStream))

### getAudioFileFormat

public static [AudioFileFormat](http://docs.google.com/javax/sound/sampled/AudioFileFormat.html) **getAudioFileFormat**([InputStream](http://docs.google.com/java/io/InputStream.html) stream)  
 throws [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Obtains the audio file format of the provided input stream. The stream must point to valid audio file data. The implementation of this method may require multiple parsers to examine the stream to determine whether they support it. These parsers must be able to mark the stream, read enough data to determine whether they support the stream, and, if not, reset the stream's read pointer to its original position. If the input stream does not support these operations, this method may fail with an IOException.

**Parameters:**stream - the input stream from which file format information should be extracted **Returns:**an AudioFileFormat object describing the stream's audio file format **Throws:** [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html) - if the stream does not point to valid audio file data recognized by the system [IOException](http://docs.google.com/java/io/IOException.html) - if an input/output exception occurs**See Also:**[InputStream.markSupported()](http://docs.google.com/java/io/InputStream.html#markSupported()), [InputStream.mark(int)](http://docs.google.com/java/io/InputStream.html#mark(int))

### getAudioFileFormat

public static [AudioFileFormat](http://docs.google.com/javax/sound/sampled/AudioFileFormat.html) **getAudioFileFormat**([URL](http://docs.google.com/java/net/URL.html) url)  
 throws [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Obtains the audio file format of the specified URL. The URL must point to valid audio file data.

**Parameters:**url - the URL from which file format information should be extracted **Returns:**an AudioFileFormat object describing the audio file format **Throws:** [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html) - if the URL does not point to valid audio file data recognized by the system [IOException](http://docs.google.com/java/io/IOException.html) - if an input/output exception occurs

### getAudioFileFormat

public static [AudioFileFormat](http://docs.google.com/javax/sound/sampled/AudioFileFormat.html) **getAudioFileFormat**([File](http://docs.google.com/java/io/File.html) file)  
 throws [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Obtains the audio file format of the specified File. The File must point to valid audio file data.

**Parameters:**file - the File from which file format information should be extracted **Returns:**an AudioFileFormat object describing the audio file format **Throws:** [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html) - if the File does not point to valid audio file data recognized by the system [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O exception occurs

### getAudioInputStream

public static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) **getAudioInputStream**([InputStream](http://docs.google.com/java/io/InputStream.html) stream)  
 throws [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Obtains an audio input stream from the provided input stream. The stream must point to valid audio file data. The implementation of this method may require multiple parsers to examine the stream to determine whether they support it. These parsers must be able to mark the stream, read enough data to determine whether they support the stream, and, if not, reset the stream's read pointer to its original position. If the input stream does not support these operation, this method may fail with an IOException.

**Parameters:**stream - the input stream from which the AudioInputStream should be constructed **Returns:**an AudioInputStream object based on the audio file data contained in the input stream. **Throws:** [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html) - if the stream does not point to valid audio file data recognized by the system [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O exception occurs**See Also:**[InputStream.markSupported()](http://docs.google.com/java/io/InputStream.html#markSupported()), [InputStream.mark(int)](http://docs.google.com/java/io/InputStream.html#mark(int))

### getAudioInputStream

public static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) **getAudioInputStream**([URL](http://docs.google.com/java/net/URL.html) url)  
 throws [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Obtains an audio input stream from the URL provided. The URL must point to valid audio file data.

**Parameters:**url - the URL for which the AudioInputStream should be constructed **Returns:**an AudioInputStream object based on the audio file data pointed to by the URL **Throws:** [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html) - if the URL does not point to valid audio file data recognized by the system [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O exception occurs

### getAudioInputStream

public static [AudioInputStream](http://docs.google.com/javax/sound/sampled/AudioInputStream.html) **getAudioInputStream**([File](http://docs.google.com/java/io/File.html) file)  
 throws [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Obtains an audio input stream from the provided File. The File must point to valid audio file data.

**Parameters:**file - the File for which the AudioInputStream should be constructed **Returns:**an AudioInputStream object based on the audio file data pointed to by the File **Throws:** [UnsupportedAudioFileException](http://docs.google.com/javax/sound/sampled/UnsupportedAudioFileException.html) - if the File does not point to valid audio file data recognized by the system [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O exception occurs

### getAudioFileTypes

public static [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 the system.

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

### isFileTypeSupported

public static 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 the system.

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

### getAudioFileTypes

public static [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 the system 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 static 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 specified file type can be written from the indicated audio input stream.

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

### write

public static 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 specified file type to the output stream provided. Some file types require that the length be written into the file header; such files cannot be written from start to finish unless the length is known in advance. An attempt to write a file of such a type will fail with an IOException if the length in the audio file type is AudioSystem.NOT\_SPECIFIED.

**Parameters:**stream - the audio input stream containing audio data to be written to the filefileType - the kind of audio file to writeout - the 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 input/output exception occurs [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the file type is not supported by the system**See Also:**[isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type)), [getAudioFileTypes()](http://docs.google.com/javax/sound/sampled/AudioSystem.html#getAudioFileTypes())

### write

public static 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 specified file type to the external file provided.

**Parameters:**stream - the audio input stream containing audio data to be written to the filefileType - the kind of audio file to writeout - the 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 type is not supported by the system**See Also:**[isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type)](http://docs.google.com/javax/sound/sampled/AudioSystem.html#isFileTypeSupported(javax.sound.sampled.AudioFileFormat.Type)), [getAudioFileTypes()](http://docs.google.com/javax/sound/sampled/AudioSystem.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/AudioSystem.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/AudioPermission.html)   [**NEXT CLASS**](http://docs.google.com/javax/sound/sampled/BooleanControl.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sound/sampled/AudioSystem.html)    [**NO FRAMES**](http://docs.google.com/AudioSystem.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#3dy6vkm) | CONSTR | [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).