

Chapter 14: Multimedia

- Audio Files
- Playing Audio
- Running Audio on a Separate Thread
- Displaying Images
- Displaying a Sequence of Images
- Using `MediaTracker`



Audio Files

- Audio is stored in files. There are several formats of audio files. JDK 1.2 can play several audio file formats, including .wav and .au files.



Playing Audio

➤ `play(URL url, String filename);`

Plays the audio clip after it is given the URL and the file name that is relative to the URL. Nothing happens if the audio file cannot be found.

➤ `play(getCodeBase(), "soundfile.au");`

Plays the sound file `soundfile.au`, located in the applet's directory.

➤ `play(getDocumentBase(),
"soundfile.au");`

Plays the sound file `soundfile.au`, located in the HTML file's directory.



Using Audio Clips

```
➤ public AudioClip getAudioClip(URL url);  
➤ public AudioClip getAudioClip(URL url,  
String name);
```

Either method creates an audio clip. Specify `String` name to use a relative URL address.

```
➤ public abstract void play()  
➤ public abstract void loop()  
➤ public abstract void stop()
```

Use these methods to start the clip, play it repeatedly, and stop the clip, respectively.



Example 14.1

Incorporating Sound in Applets

- Objective: Display a running clock and play sound files to announce the time at every minute.

[ClockAppletWithAudio](#)

Run Applet Viewer



Clock

Defined
In Example 13.5

CurrentTimeApplet

Defined
In Example
12.1

ClockWithAudio

-ClockAppletWithAudio applet

+paintComponent(g)

1

1

ClockAppletWithAudio

-ClockWithAudio clock
-hourAudio[]
-minuteAudio
-amAudio
-pmAudio

+init()
+createClcok()
+announceTime(s, m, h)
+start()
+stop()



Running Audio on a Separate Thread

- 👉 Objective: Avoid the conflict between painting the clock and announcing time in Example 14.1 by running the tasks on separate threads.

[ClockAppletWithAudioOnSeparateThread](#)

Run Applet Viewer



Displaying Images

Two methods are available for displaying images:

- Use the `getImage ()` method to retrieve image files and create `Image` objects.
- Paint the images on the viewing area using the `drawImage ()` method.



Example 14.3: Displaying Images in an Applet

👉 Objective: Display images in applets

DisplayImageApplet

Run Applet Viewer



Example 14.4: Using Image and Audio in Applets and in Applications

➔ Objective: Display images and playing audio in applets and in applications.

ResourceLocatorDemo

Run as an Application

Run as an Applet



Example 14.5: Using Image Animation

- ➔ Objective: Simulate a movie by displaying a sequence of images in a control loop.

ImageAnimation

Run Applet Viewer

Note: Images may take several seconds to load.



Example 14.6: Using MediaTracker

- Objective: Use the `MediaTracker` class to load all the images before displaying them in a sequence.

[ImageAnimationUsingMediaTracker](#)

Run Applet Viewer

Note: Images may take several seconds to load.

