9 Integrating Audio and Video



Objectives

You will have mastered the material in this chapter when you can:

- Describe the benefits and limitations of multimedia in websites
- Identify audio formats
- Identify video formats
- Describe a plug-in
- Understand codecs
- Understand and create audio elements

- Understand and create video elements
- Identify common audio attributes
- Identify common video attributes
- Understand the source element
- Test audio elements
- Test video elements

9 Integrating Audio and Video

Introduction

A website provides several creative opportunities to capture the attention of your audience. Many websites use multimedia (digital media such as audio, video, and animation) to enrich the user experience and provide interactivity. For example, websites include videos to share news updates, promote products and services, or demonstrate how to use a product. Videos add visual appeal and can pack a huge amount of information into a few seconds. According to a recent study by Forrester Research, consumers prefer videos to text, find them more memorable, and stay on websites that include videos longer than websites that do not.

Audio on webpages can set a mood or tone and further entice a user to make a purchase. Audio can also provide information more quickly or effectively than text, such as a guided tour that describes the features of a product or testimonials from customers that support a product or service.

In this chapter, you will learn how to use HTML5 to embed audio and video on an HTML webpage. You will learn how to use the **audio** element and its attributes to add audio to a webpage. Similarly, you will learn how to use the **video** element and its attributes to add a video to webpage. You will also learn about the various audio and video file formats and which formats to use for the web. Finally, you will learn how to style the video using CSS.

Project — Add Audio and Video to a Webpage

In Chapter 9, you enhance the Forward Fitness Club website by adding audio to the home page to improve the user experience. Additionally, you insert a video in the About Us page to engage the user and promote the club.

This chapter provides an introduction to using multimedia in web development, focusing on two forms of multimedia: audio and video. The finished webpages contain relevant audio and video clips that provide valuable multimedia content for visitors of the Forward Fitness Club website.

The project in this chapter improves a website by integrating HTML5 audio and video elements. To perform these tasks, you first copy the media files to your media folder. Next, you add an audio element to a webpage. Then, you add a video element to a webpage. In the style sheet, you add style rules for the video element for tablet and desktop viewports. Figure 9–1 shows the home page with the embedded audio and controls. Figure 9–2 shows the video on the About Us page.



Figure 9-1



Figure 9-2

Roadmap

In this chapter, you will learn how to create the webpages shown in Figures 9–1 and 9–2. The following roadmap identifies general activities you will perform as you progress through this chapter:

- 1. INSERT the AUDIO ELEMENT.
- 2. **INSERT** the **VIDEO ELEMENT**.
- 3. **STYLE** the **VIDEO ELEMENT**.

At the beginning of step instructions throughout the chapter, you will see an abbreviated form of this roadmap. The abbreviated roadmap uses colors to indicate chapter progress: gray means the chapter is beyond that activity; blue means the task being shown is covered in that activity; and black means that activity is yet to be covered. For example, the following abbreviated roadmap indicates the chapter would be showing a task in the 2 INSERT VIDEO ELEMENT activity.

1 INSERT AUDIO ELEMENT | 2 INSERT VIDEO ELEMENT

Use the abbreviated roadmap as a progress guide while you read or step through the instructions in this chapter.

Using Multimedia

The popularity of the web is due in part to its ability to display webpages that include graphic images, audio, and video. These additions can boost the visual appeal of a website and make the browsing experience more enjoyable and interactive. Sometimes, however, multimedia can distract from the website message. When deciding whether to include multimedia, remember the purpose of the website. If the multimedia content enhances that purpose, it should be included. If the multimedia content distracts from the purpose of the website, then you should reconsider using it.

Multimedia is the combination of text, images, sound, and video to express an idea or convey a message. Because most people have broadband Internet connections that can transfer data quickly, multimedia webpages that include large graphics, audio, and video are common. Figure 9–3 shows an example of multimedia used on the PBS website.



Many websites use videos to advertise products, to entertain visitors, or to provide instruction. For example, you may be able to review a medical procedure online before having the procedure done yourself. You can view clips of movies or hear segments of audio recordings from webpages that provide content in those formats. **Podcasts**, a series of audio or video clips that are released in a sequence, are popular in home, academic, and corporate settings. Figure 9–4 shows an example of a website, Podcast Alley, with several podcasts on current technology topics.



Figure 9-4

Creating Multimedia Files

You can obtain multimedia files by creating them yourself or by finding files that are already available. If you create your own multimedia files, you do not have to be concerned with copyright or licensing agreements. On the other hand, the multimedia files might not have the same quality and effectiveness as professional content. You can create your own audio files using a microphone and software designed to edit digital files, such as Audacity, a free, open-source audio editor, and Adobe Audition, which can be purchased as part of the Adobe Creative Cloud. If you do use any portion of files that have been professionally developed, such as those from an online multimedia provider, be certain that you understand and follow the copyright and licensing requirements.

For video files, you can use a digital camcorder, a digital camera, or even a smartphone to create clips that can be included on a webpage. For the Windows operating system, a popular, free video editing application is Windows Movie Maker (Figure 9–5). This application is available for download from the Microsoft website as part of the Windows Live Essentials software package. MAGIX Movie Edit Pro software and training is available for a fee, with an active online community for additional support. Corel VideoStudio Pro has simple and more advanced menu system options that novice movie editors and professionals alike can use comfortably. CyberLink PowerDirector is another software option that allows you to create professional-quality videos.

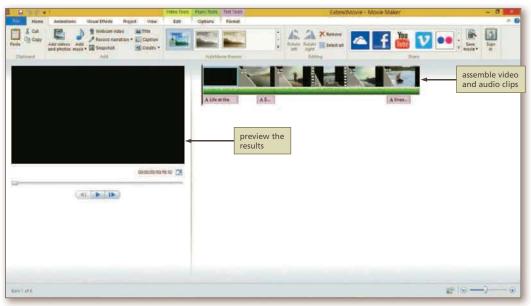


Figure 9-5

For Mac users, iMovie, part of the iLife software suite, is a low-cost option. The iMovie app is also available for download to the iPad from the App Store. Final Cut Pro X is a video editor for the Mac that offers professional-level editing.

Adobe Premiere Pro is another highly rated application for movie editing that can be used on either the Mac or Windows operating systems. As with many multimedia software products, Adobe Premiere Pro lets novice users create professional-quality video. Figure 9–6 shows the Adobe Premiere Pro Creative Cloud application.

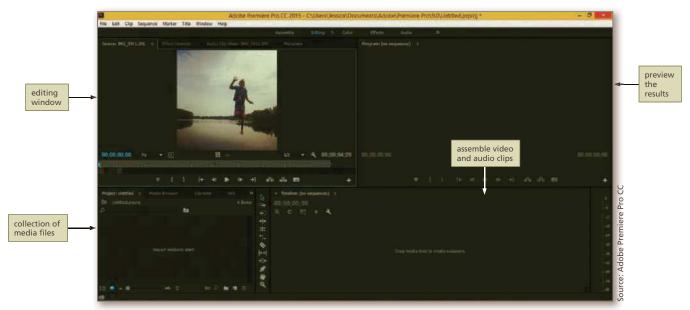


Figure 9-6

Many audio- and video-editing applications offer a free trial period. You might try downloading a free trial of the software, if available, and using it to create or edit a movie. You could also try a few different programs to see how they work. Most multimedia software operates in a similar fashion and has a user-friendly help utility that guides you through the process of editing. Most products also provide templates and effects that enhance the creative aspects of presenting multimedia.

To find multimedia resources on the web, search for "public domain audio or video." These resources are available free of copyright restrictions. Websites that provide stock photos often provide music, sound, and video files for a fee. Before using any content, be sure to understand and follow the copyright license agreements that accompany any multimedia content that you find on the web.

Can I use any images, audio, or video I find on the Internet to create a multimedia file?

In general, no, you cannot use any file you find. However, as a student, you are permitted to use some for academic purposes. This is known as Fair Use. For more information about copyright laws, visit copyright.gov.

Embedded vs. External Multimedia

Embedding media is similar to inserting inline images. The embedded media files appear within the webpage along with the audio or video player controls. Visitors use the controls to play or stop the media. Because the media file is embedded directly into the webpage, you can complement the audio or video clip resource with surrounding text or graphical images. Before HTML5, the <code>object</code> element was used to insert embedded content, including multimedia. HTML5 introduces two new elements, <code>audio</code> and <code>video</code>, to use as an alternative to the <code>object</code> element. Figure 9–7 shows an example of how the U.S. Department of Defense uses the <code>video</code> element on its webpage.



Figure 9-7

To access external media files, website visitors click a link. They can then decide whether to link directly to the external source or download the file. Unlike embedded media, the external media is displayed out of context with the webpage that contains the link. Using external links is a common web development practice. For example, sites that provide many video resources, such as YouTube, use external media files and the <code>embed</code> element, which defines a container for an external application or interactive content (also called a plug-in). Figure 9–8 shows an example of a White House video provided on YouTube through an external link.



Figure 9-8

Media Players and Plug-Ins

As you know, browsers can display text and the graphics formats discussed in earlier chapters without using external software. To play an audio or video file, however, some browsers need the help of an application called a media player or a plug-in. A **media player** is computer software that plays multimedia files. Most software media players support an array of media formats, including audio and video files. This chapter uses an .mp3 audio file and an .mp4 video file. The Internet Explorer browser can play files in both formats. Windows Media Player, included as part of Windows, also plays both audio and video. The Mac operating system comes with QuickTime Player for playing movies, while iTunes can be downloaded for Windows or Mac OS to play a variety of media formats.

A **plug-in** (also called an add-in or add-on) is extra software added to the browser (or other program) to provide a capability that is not inherent to the browser. In other words, for an embedded media file to work in a browser, the website visitor needs to have the correct plug-in. Most browsers have a variety of plug-ins installed, but website visitors can also download and install necessary plug-ins from the browser manufacturers. Common plug-ins are Windows Media Player, Apple QuickTime Player, Adobe Flash, Microsoft Silverlight, and Adobe Acrobat. Figure 9–9 shows an example of plug-ins installed on the Internet Explorer 11 browser.

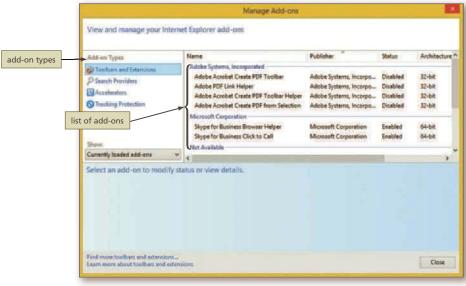


Figure 9-9

As you will see in the following section, the various audio and video formats can be played on a variety of players. For embedded multimedia, use a format supported by multiple players.

BTW

Plug-ins and Add-ons

Investigate the plug-ins available for the version of browser that you use. (In Internet Explorer, display the Command bar, click Tools on the Command bar, and then select Manage add-ons.) Search the web for information about the listed add-ons and determine if you can download other add-ons that you might need.

HTML5 and Multimedia

HTML5 introduces built-in media support via the audio and video elements. Using these elements, web developers can easily embed media into HTML documents. This chapter teaches you how to use the new HTML5 audio and video elements. Additional multimedia types discussed in this chapter include Flash, Java applets, and the HTML object element.

Will the ActiveX controls and security notification appear if I use a browser other than Internet Explorer?

No. ActiveX controls are displayed only in the Internet Explorer web browser.



Flash, or Adobe Flash, has been used within websites for nearly 20 years. Using Flash, you can create animations or movie files. Flash files have the .swf file extension and require the browser to have a Flash plug-in. Incorporating Flash animation was popular on websites before using smartphones and tablets became widespread. Some websites were developed exclusively with Flash. However, many of today's websites use Flash sparingly, if at all, partly because the iPhone and iPad, which run the iOS operating system, do not support Flash. Instead, many web developers have embraced JavaScript to incorporate additional interactivity within their websites. Figure 9–10 shows an example of an interactive Flash video used on the National Parks Service website.

Java Applets

A **Java applet** is a small program created with Java, a programming language. A Java applet can be embedded within a webpage, though the browser must have Java installed and enabled. Java applets were very popular in the late 1990s; however, their use in today's modern websites is in decline.



Figure 9-10

Object Element

You use the HTML <code>object</code> element to embed multimedia objects such as Flash files, Java applets, PDF files, and ActiveX controls. Use the <code>param</code> element to define parameters for plug-ins embedded with an <code>object</code> element. The following is an example of the <code>object</code> element.

```
<object data="audio.wav">
  <param name="autoplay" value="true">
</object>
```

In this example, an audio file named audio.wav is embedded on the webpage. The autoplay parameter is set to true, meaning the audio starts playing when the webpage opens.

Integrating Audio

Adding audio to a webpage can help set a desired mood or tone. It can also be a distraction to some users, so you must first determine if the audio is appropriate for the webpage and its audience. Audio integrated within a webpage should have a distinct purpose and should provide added value or instruction. You should also consider the time it takes for the browser to load the audio file. If your audience includes users with dial-up connections rather than broadband access, they will likely experience a delay, especially if the audio files are large. Keep in mind that using audio or video on a mobile device requires a data plan if the device is not connected to a wireless network. One popular way that websites use audio is to provide links to music files that visitors can play or download. Figure 9–11 shows list of music files to listen to on the Noisetrade.com website.

BTW

Parameters

Many other object parameters are available that are not discussed in this chapter. Each ActiveX control has particular parameters. Review online other parameters that might be effective for you to use in your web development.



Figure 9-11

Audio File Formats

Table 9–1 lists the most common audio file formats for webpages. The HTML5 audio element supports only three audio file formats: .mp3, .ogg, and .wav. These supported formats are indicated with an asterisk in Table 9–1. However, you can use audio converter software to convert files from one audio format to a supported format. Audio files for the web often use file compression techniques to reduce the size of the file, though they can also diminish the sound quality. Uncompressed audio formats listed in Table 9–1 include .aiff, .au, and .wav.

File Format	File Extension	Description
AIFF	.aiff	Standard audio file format developed by Apple As an uncompressed and lossless format, it uses more disk space than the MP3 format
AU	.au	Standard audio file format used by Sun, Unix, and Java Can be compressed
MIDI	.mid .rmi	Musical Instrument Digital Interface (MIDI) Limited to electronic musical instruments (such as synthesizers) and other electronic equipment Files can be much smaller than in other formats
MP3*	.mp3	One of the most popular formats for music storage Compresses files to approximately one-tenth the size of uncompressed files
MP4	.mp4	Based on the QuickTime format; used for audio and video Creates quicker, faster, high-quality media Not supported by Windows Media Player
Ogg*	.ogg	Maintained by Xiph.Org Foundation Designed to provide for efficient streaming and high-quality digital multimedia Can be used with audio element
RealAudio	.ra .ram	Designed for streaming audio over the Internet; declining use Sound quality not as good as other formats
WAV*	.WaV	Standard audio format for Windows Commonly used for storing uncompressed CD-quality sound files All RiGompression is available to reduce tile sizeblicated, in whole or in part. WCN 02-200-203

File Compression and Codecs

When embedding an audio or video file onto a webpage, you should keep the file size small and compress the file when necessary. A **codec** is a compression technology used to compress images, audio, and video files. The word codec is short for code/ decode because it consists of an encoder, which compresses the file, and a decoder, which decompresses the file. An audio editor uses a codec to reduce the size of an audio file, while maintaining enough quality to play the audio file on a computer. Although codecs improve the page load time, compressing a media file too much results in a loss of sound quality. On the other hand, if you do not compress a media file enough, the file may take too long to download into a browser. Figure 9–12 shows an example of audio converter software that uses a codec to compress files.

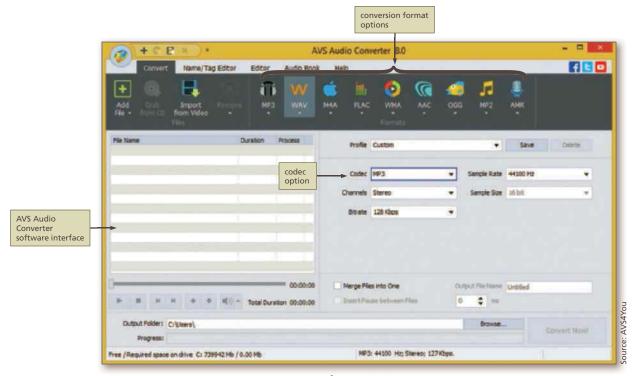


Figure 9-12

HMTL5 audio Element

One of the most long-awaited features in HTML5 is the **audio** element, which you use to define sound, such as music or other audio streams. Table 9–2 shows the attributes and values that can be used with the **audio** element.

The following sample code shows how to use the audio element to insert an audio file named music.mp3. Because browsers that do not support the audio element ignore the <audio> tag, you should insert text content between the <audio> and </audio> tags to alert users, as the paragraph element does in the following code.

```
<audio controls="controls" autoplay="autoplay">
    <source src="music.mp3" type="audio/mp3">
    Your browser does not support the HTML5 audio element.
</audio>
```

In this example, an audio file named music.mp3 is embedded on the webpage. The first line shows the beginning audio tag and includes two attributes: controls and autoplay. The controls attribute specifies to display the audio controls, such

Table 9–2 Audio Element Attributes				
Attribute	Value	Description		
autoplay	autoplay	Specifies that the audio will automatically start playing		
controls	controls	Specifies that audio controls should be displayed (such as a play/pause button)		
loop	loop	Specifies that the audio will start playing every time it is finished		
muted	muted	Specifies that the audio should be muted		
preload	auto metadata none	Specifies whether and how the audio should be loaded when the page loads		
src	URL	Specifies the URL of the audio file		

as play/pause and volume control. The autoplay attribute specifies to start playing the audio file once the page is loaded in the browser. The second line uses the source tag with two attributes, src and type. The src attribute specifies the location of the audio file. The type attribute specifies the type of audio file, in this instance, MP3. The third line is a paragraph element with fallback text to let the user know that their browser does not support the audio element. The sentence, "Your browser does not support the HTML5 audio element" appears in a legacy browser that does not support HTML5. The fourth line includes the ending audio tag.

If you use the object element instead of the audio element, you must include the data attribute to specify the source file. Only one data attribute can be specified within one object element. In addition, you need to include the param element in the object element to specify additional parameters, such as autoplay. However, many of the attributes for the object element and the parameter element are not supported in HTML5.

The **controls** attribute adds audio controls, such as play, pause, and volume. It can be set up in any of the three following ways:

```
<audio controls="controls">
<audio controls>
<audio controls="">
Similarly, the autoplay attribute can be set up in any of these ways:
<audio autoplay>
<audio autoplay="autoplay">
<audio autoplay="">
<audio
```

To enable the audio element to work in all browsers, use source elements inside the audio element. The source elements can link to the same or different audio files as appropriate. Including the source elements accommodates visitors who use any of the five major browsers. The browser will use the first recognized format. The following is an example of an audio element with multiple source elements:

BTW Audio Clip

Many web design sites discuss the use of audio clips in web development. Search for ideas of how you can most effectively use an audio clip. In this example, if the browser cannot play MP3 files, it checks to determine whether it can play Ogg files. If it can, it plays the music.ogg file. If it cannot play Ogg files, it checks to determine whether it can play WAV files. If it can, it plays the music. wav file. Otherwise, it does not play any audio. If the browser cannot play the audio, then it does not support HTML5. Table 9–3 lists the three audio file formats supported by the audio element and identifies whether each file format is also supported by the five major browsers. While most of these browsers support more than the audio file formats listed in Table 9–3, at this time, the audio element only supports three.

Other attributes include the preload attribute, which tells the browser to begin downloading the audio file immediately when it encounters the audio element. However, if you use the autoplay attribute, which also begins downloading immediately, you do not need to also use the preload attribute. The loop attribute restarts the audio immediately once it has finished playing.

Table 9–3 Audio File Browser Support						
Audio File Format	Internet Explorer	Google Chrome	Mozilla Firefox	Apple Safari	Opera	
MP3	•	•	•	•	•	
Ogg		•	•		•	
Wav		•	•	•	•	

To Add Audio to the Home Page

1 INSERT AUDIO ELEMENT | 2 INSERT VIDEO ELEMENT

Insert an **audio** element on the home page of the Forward Fitness Center website to add sound to the page. *Why?* Set an energetic tone with a short audio file to further enhance the user's interest. Include three file sources to accommodate all browsers and include attributes to autoplay the audio file and to display controls on the webpage. The following steps add audio to the home page.

- 1
- Copy the media files from the chapter09\fitness\media folder provided with the Data Files for Students to the fitness\media folder.
- Open index.html in your text editor and update the comment with today's date.

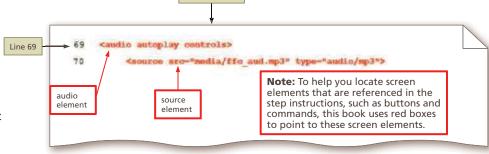


Figure 9-13

- Tap or click at the end of Line 67,
 and then press the ENTER key twice to insert new Lines 68 and 69.
- On Line 69, type <audio autoplay controls> to insert a starting audio tag.
- Press the ENTER key to insert a new Line 70, increase the indent, and then type source src="media/
 ffc aud.mp3" type="audio/mp3"> to insert a source element (Figure 9–13).
- ▼ | What is the purpose of the autoplay and controls attributes?
- The autoplay attribute specifies to automatically play the audio file in the browser. The controls attribute specifies to display the audio controls within the browser, which allows the user to play or pause the music and adjust the volume.

Why did I begin inserting the code on Line 69?

The code appears at the end of the main element, so the audio controls will appear at the end of the page, before the footer.



- Press the ENTER key to 69 <audio autoplay controls> insert a new Line 71, and 70 <source src="media/ffc_aud.mp3" type="audio/mp3"> then type <source e sro-"media/ffc aud.ogg" type-"audio/ogg Line 71 71 src="media/ source elements 72 e sro-"media/ffc aud.wav" type-"audio/wav"> ffc aud.ogg" 73 Your browser does not support the audio tag. paragraph element type="audio/ogg"> 74 closing audio tag to insert a source element.
- Press the ENTER key to insert a new Line 72, and then type <source src=" media/ffc aud.wav" type="audio/wav"> to insert a source element.
- Press the ENTER key to insert a new Line 73, and then type Your browser does not support the audio tag. to insert a paragraph element.
- Press the ENTER key to insert a new Line 74, decrease the indent, and then type </audio> to insert a closing audio tag (Figure 9–14).
- Why do I need to include three source elements?
- To accommodate all major browsers, you specify three audio source files, MP3, Ogg, and WAV. These are the three file formats supported by the audio element.

Will the webpage display the paragraph element?

The webpage displays the paragraph element only if the browser does not support the audio element.



- The loop attribute

 automatically replays the audio file after it is finished. To see how this works, add the loop attribute to the audio tag, save your changes, and then refresh index.html in your browser.
- Remove the controls attribute from the audio element to view the page without controls, save your changes, and then refresh index.html in your browser.
- Remove the loop attribute, add the controls attribute to the audio tag, and then save your changes.
- A message appears at the bottom of my window, "Internet Explorer restricted this webpage from running scripts or ActiveX controls." How should I proceed?

Tap or click the Allow blocked content button to play the audio file and display the audio controls.

My controls do not look like the controls in Figure 9–15. Why?

Figure 9–15 shows the controls in Internet Explorer 11. If you are using a different browser, your controls will look different.

Break Point: If you want to take a break, this is a good place to do so. You can exit the text editor now. To resume at a later time, run your text editor, open the file called about.html, and continue following the steps from this location forward.

Integrating Video

Adding video to a webpage can increase traffic to your website if the video is informative and meaningful. As with audio files, consider the time it takes for the browser to load the video file. Non-broadband users will likely experience a delay, especially if the video files are large, and mobile device users might incur costs on a data plan. For these reasons, include only those videos that create a rich experience for the user.

Video File Formats

Table 9–4 lists the most commonly used video file formats for the web. The HTML5 **video** element supports only three of these video file formats: .mp4, .ogg, and .webm. The supported formats are indicated with an asterisk in Table 9–4. If you want to use an unsupported video file format with the **video** element, use video converter software to convert the file to a supported format. The **video** element does not work in legacy browsers, so consider your audience before using the **video** element. Most video file formats for the web use codec compression techniques to reduce the file size.

Table 9–4 Common Video File Formats			
File Format	File Extension	Description	
AVI	.avi	Audio / Video Interleaved Developed by Microsoft to use with Windows Can contain both audio and video data	
Flash	.swf	Small Web Format Can contain audio, video, or animations Requires Adobe Flash Player	
Flash Video	.flv .f4v	Developed by Adobe Format of choice for embedded video on the web Used by YouTube and Hulu	
MPEG	.mpg .mpeg .mp3	Moving Picture Experts Group Can be highly compressed, resulting in small file size Supported by most major browsers	
MP4*	.mp4	Based on QuickTime format; used for audio and video Creates quicker, faster, high-quality media Not supported by Windows Media Player	
Ogg*	.ogg	Maintained by Xiph.org Foundation Designed for efficient streaming and high-quality digital multimedia Can be used with HTML5 video element	
QuickTime	.mov	Developed by Apple for both Windows and Mac operating systems File compression can result in smaller file size Requires QuickTime Player or Adobe Flash Player, which are easily downloaded	
RealVideo	.rm .rv	Proprietary video format developed by RealNetworks Requires RealPlayer	
WebM*	.webm	Developed by Google Royalty-free, open format Can be used with HTML5 <video> tag</video>	
Windows Media	.wmv	Developed by Microsoft Originally designed for Internet streaming applications Requires Windows Media Player or RealPlayer	

HTML5 video Element

Until the new HTML5 video element was introduced, browsers did not have a standard for playing video on a webpage. Most videos were shown through a plug-in (such as Adobe Flash). The problem with plug-ins is that browsers may have different plug-ins or only recognize specific video formats. The HTML5 video element solves this problem and provides a standard way to play web videos in today's modern browsers.

You can use three HTML elements to incorporate video: embed, object, and video. You would have to add all three elements and use different video formats to make sure your video can play in all browsers and browser versions (Internet Explorer, Chrome, Firefox, Safari, and Opera) and on all hardware (PC, Mac, Android Tablet, Android Smartphone, iPad, and iPhone).

The purpose of the **embed** element is to embed multimedia elements in HTML pages. However, the **embed** element is unknown to HTML4 and is not validated correctly. In addition, the **embed** element causes other problems: if the browser does not support Flash, the video specified in the **embed** element will not play; iOS devices such as the iPad and iPhone do not display Flash videos; and if you convert the video to another format, it still does not play in all browsers.

You can also use the <code>object</code> element to embed multimedia elements in HTML pages. Although the code with an <code>object</code> element is validated correctly, the <code>object</code> element causes other problems: if the browser does not support Flash, your video will not play; iOS devices such as the iPad and iPhone do not display Flash videos; and if you convert the video to another format, it still does not play in all browsers.

Using the video Element

Use the **video** element to specify a video, such as a movie clip or other video stream, on a webpage. Table 9–5 lists the attributes and values that can be used with the **video** element.

Table 9–5 Video Element Attributes				
Attribute	Value	Description		
autoplay	autoplay	Specifies that the video will start playing as soon as it is ready		
controls	controls	Specifies that video controls should be displayed (such as a play/pause button)		
height	pixels	Sets the height of the video player		
loop	loop	Specifies that the video will start playing every time it is finished		
muted	muted	Specifies that the audio output of the video should be muted		
poster	URL	Specifies an image to be shown while the video is downloading, or until the user clicks the play button		
preload	auto metadata none	Specifies whether and how the video should be loaded when the page loads		
src	URL	Specifies the URL of the video file		
width	pixels	Sets the width of the video player		

The following sample code shows how to use the **video** element to insert a video file named advertisement.mp4 in a webpage. Because browsers that do not support the **video** element ignore the <video> tag, you should insert text content

between the <video> and </video> tags to alert users, as the paragraph element does in the following code.

```
<video width="320" height="240" controls="controls">
    <source src="advertisement.mp4" type="video/mp4">
     Your browser does not support the HTML5 video element.
</video>
```

In this example, the code sets the dimensions of the video (320 pixels by 240 pixels) and displays the playback controls. The browser plays the advertisement.mp4 file, which is an MP4 video file. If the browser does not support the **video** element, the user is alerted.

It is a good idea to include width and height attributes when you are not designing a responsive website. If you set the width and height, the space required for the video is reserved when the page is loaded. However, without these attributes, the browser does not know the size of the video and cannot reserve the appropriate space for it. The effect is that the page layout will change when the video loads. If you are designing a responsive website, do not include width and height attributes. Instead, use CSS to set the height to auto and the max-width attribute to 100%, as this will allow the video to automatically resize to fit the user's viewport.

As with the audio element, the controls attribute adds video controls including play, pause, and volume. It can be set up in any of the three following ways:

```
<video controls="controls">
<video controls>
<video controls="">
```

You can use the **src** attribute in the **video** element itself, as shown in the sample below:

```
<video controls autoplay src="advertisement.mp4">
  Your browser does not support the HTML5 video element.
</video>
```

Also, as you did with the audio element, you should use the source elements inside the video element. The video element allows multiple source elements, which can link to different video files. Including the source elements accommodates visitors who use any of the five major browsers. The browser will use the first recognized format. The following is an example of a video element with multiple source elements:

```
<video controls="controls" autoplay="autoplay">
    <source src="advertisement.mp4" type="video/mp4">
    <source src="advertisement.ogg" type="video/ogg">
    <source src="advertisement.webm" type="video/webm">
    Your browser does not support the HTML5 video element.
</video>
```

In this example, if the browser cannot play MP4 files, it checks to determine whether it can play Ogg files. If it cannot play the advertisement.ogg file, it checks to determine whether it can play WebM files. If it cannot play MP4, Ogg, or WebM files, then the browser does not support the HTML5 video element. Table 9–6 lists the three video file formats supported by the video element and identifies whether or not each file format is supported by the five major browsers. While most of these browsers support more than the video file formats listed in Table 9–6, at this time, the

BTW Aligning Videos on a Webpage

The video element is merely a container to hold a video. To center a video on a page, wrap the video element within a div element and then style the div element.

video element only supports three.

Table 9–6 Video File Browser Support						
Video File Format	Internet Explorer	Google Chrome	Mozilla Firefox	Apple Safari	Opera	
MP4						
Ogg						
WebM						

BTW

Video Clips

Video clips can have a large file size depending on the length and the quality of the clip. Search for rulesof-thumb for file sizes when video clips are incorporated into a website.

To Add Video to the About Us Page

1 INSERT AUDIO ELEMENT | 2 INSERT VIDEO ELEMENT 3 STYLE VIDEO ELEMENT

Next, insert a video on the About Us page to highlight its services. Why? You can use a video to promote the benefits of joining the Forward Fitness Club. Include three file sources to accommodate all browsers. Also include attributes to automatically play the video file and to display controls on the webpage. The following steps add video to the About Us page.



- Open about.html in your text editor and update the comment with today's date.
- Tap or click at the end of Line 31, and then press the ENTER key twice to insert new Lines 32 and 33.
- On Line 33, increase the indent, and then type <div class="video"> to insert a div tag.

insert a source element.

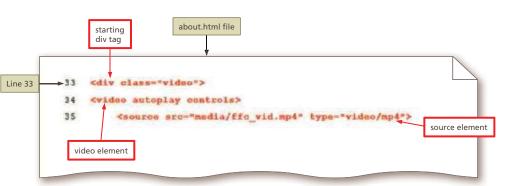


Figure 9-16

- Press the ENTER key to insert a new Line 34, and then type <video autoplay controls> to insert a starting video tag.
- Press the ENTER key to insert a new Line 35, increase the indent, and then type <source src="media/ffc" vid.mp4" type="video/mp4"> to insert a source element (Figure 9–16).
- ◀ | Why am I creating a new div element with a class value of video?
- In future steps, you will style the video. In order to center the video on the webpage, wrap it within a div element and include a class value of video.

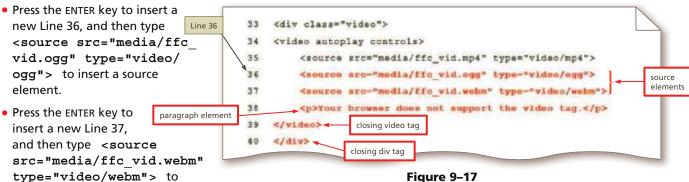


Figure 9-17

- Press the ENTER key to insert a new Line 38, and then type Your browser does not support the video tag. to insert a paragraph element.
- Press the ENTER key to insert a new Line 39, decrease the indent, and then type </wideo> to insert a closing audio tag.
- Press the ENTER key to insert a new Line 40, and then type set / div zoot to insert a closing div tag (Eigure 9–17).

Why do I need to include three source elements?

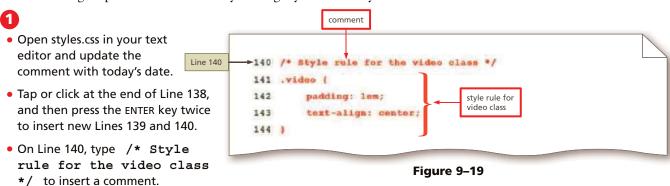
To accommodate all major browsers, you specify the video source files in three formats: MP4, Ogg, and WebM. The video element supports these three file formats.



To Style the Video

1 INSERT AUDIO ELEMENT | 2 INSERT VIDEO ELEMENT

You have added a video to the About Us page, but need to format the video so it is appealing on the webpage. Start by centering the video, and then apply borders and padding to further improve the video's appearance on the page. Wby? Formatting videos enhances their visual appeal to the user. Also create a style rule for the video element to accommodate responsive design by specifying height and max-width attribute values. The following steps format the video by adding styles to the styles.css file.



- Press the ENTER key to insert a new Line 141, and then type .video { to insert a new selector.
- Press the ENTER key to insert a new Line 142, increase the indent, and then type <code>padding: lem;</code> to insert a new property and value.
- Press the ENTER key to insert a new Line 143, and then type text-align: center;
 to insert a new property and value.
- Press the ENTER key to insert a new Line 144, decrease the indent, and then type } to insert a closing brace (Figure 9–19).

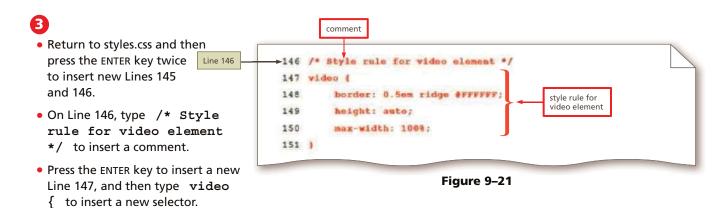


 Save your changes and refresh about.html in your browser (Figure 9–20).

> video element is center aligned



Figure 9-20



- Press the ENTER key to insert a new Line 148, increase the indent, and then type border: 0.5em ridge #FFFFFF; to insert a new property and value.
- Press the ENTER key to insert a new Line 149, and then type height: auto; to insert a new property and value.
- Press the ENTER key to insert a new Line 150, and then type max-width: 100%; to insert a new property and value.
- Press the ENTER key to insert a new Line 151, decrease the indent, and then type } to insert a closing brace (Figure 9–21).
- What is the result of the ridge value specified with the border property? The ridge value defines a 3-D ridged border.



- Save your changes and refresh about. html in your browser (Figure 9-22).
- Why does the border color appear white and gray? The color combination is the result of specifying a ridge border.



Figure 9-22

 Adjust your browser to a tablet viewport (Figure 9-23).



Figure 9-23

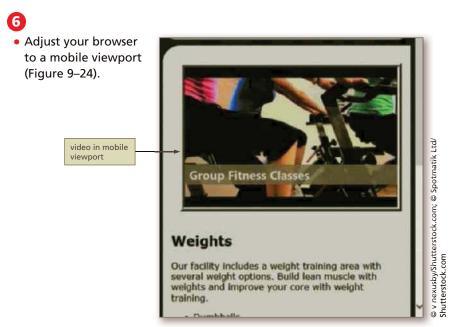


Figure 9-24

To Validate the Style Sheet

Always run your files through W3C's validator to check the document for errors. If the document has any errors, validating gives you a chance to identify and correct them. Validation is also an effective troubleshooting tool during the development process and adds a valuable level of professionalism to your work. The following steps validate a CSS document.

- 1 Open your browser and type http://jigsaw.w3.org/css-validator/ in the address bar to display the W3C CSS Validation Service page.
- 2 Tap or click the By file upload tab to display the Validate by file upload information.
- 3 Tap or click the Browse button to display the Choose File to Upload dialog box.
- Navigate to your css folder to find the styles.css file.
- Tap or click the styles.css document to select it.
- 6 Tap or click the Open button to upload the selected file to the W3C CSS validator.
- 7 Tap or click the Check button to send the document through the validator and display the validation results page.
- 8 If necessary, correct any errors, save your changes, and run through the validator again to revalidate the page.

To Validate the HTML Files

Every time you create a new webpage or modify an existing webpage, run it through W3C's validator to check the document for errors. If any errors exist, you need to correct them. Validation is also an effective troubleshooting tool during the development process and adds a valuable level of professionalism to your work. The following steps validate an HTML document.

- 1 Open your browser and type http://validator.w3.org/ in the address bar to display the W3C validator page.
- Tap or click the Validate by File Upload tab to display the Validate by File Upload information.
- 3 Tap or click the Browse button to display the Choose File to Upload dialog box.
- 4 Navigate to your fitness folder to find the index.html file.
- 5 Tap or click the index.html document to select it.
- 6 Tap or click the Open button to upload it to the W3C validator.
- 7 Tap or click the Check button to send the document through the validator and display the validation results page.
- 8 If necessary, correct any errors, save your changes, and run through the validator again to revalidate the page.
- Follow these steps to validate the about.html page and correct any errors.

Chapter Summary

In this chapter, you learned how to integrate audio and video elements within a website. You used HTML5 audio element to insert three audio source elements and specified that it automatically play and display controls. You also learned how to use the HTML5 video element and insert a video onto a webpage and coded for cross-browser compatibility. Finally, you used CSS styles to format a video for all viewports. The items listed below include all the new skills you have learned in this chapter, with the tasks grouped by activity.

Integrating Audio

Add Audio to the Home Page (HTML 456)

Integrating Video

Add Video to the About Us Page (HTML 451) Style the Video (HTML 452)



How will you use multimedia?

Use these guidelines as you complete the assignments in this chapter and create your own webpages outside of this class.

- 1. Determine if audio has a defined purpose for your website.
 - a) Determine if the audio should automatically play when the page loads.
 - b) Determine whether to show or hide the audio controls.
- 2. Determine if a video can enhance your website.
 - a) Determine the ideal placement of your video to make it stand out.
 - b) Determine how to style the video for mobile, tablet, and desktop viewports.
- 3. Determine how to best obtain multimedia for your website.
 - a) Identify tools and software necessary to create your own video.
 - b) Identify public domain content available for use.
 - c) Learn how to use multimedia and converter software.
- 4. Determine if other forms of multimedia are appropriate for your website.
 - a) Identify the pros and cons of using Adobe Flash.
 - b) Determine which HTML elements are best for including your multimedia.

How should you submit solutions to questions in the assignments identified with a ** symbol? Every assignment in this book contains one or more questions identified with a ** symbol. These questions require you to think beyond the assigned presentation. Present your solutions to the questions in the format required by your instructor. Possible formats may include one or more of these options: create a document that contains the answer; present your answer to the class; discuss your answer in a group; record the answer as audio or video using a webcam, smartphone, or portable media player; or post answers on a blog, wiki, or website.

Apply Your Knowledge

Reinforce the skills and apply the concepts you learned in this chapter.

Adding Audio to a Webpage

Instructions: In this exercise, you will use your text editor to add audio to a webpage. Insert an audio element and three source elements. Work with the apply09.html file in the apply folder and audio files in the apply\media folder from the Data Files for Students. The completed webpage is shown in Figure 9–25. You will also use professional web development practices to indent, space, comment, and validate your code.



Figure 9-25

Perform the following tasks:

- 1. Open apply09.html in your text editor, review the page, add a title, modify the comment at the top of the page to include your name and today's date, and replace "Student's Name" with your name in the footer element.
- 2. Insert an audio element below the figure element.
- 3. Set the audio element to play automatically and to repeat continuously.
- 4. Insert three **source** elements within the **audio** element and use the three audio files contained within the apply\media folder.
- 5. Include a paragraph element within the audio element that advises legacy browser users that their browser does not support the audio element.
- 6. Save all of your changes and open the apply09.html in your browser.
- 7. Validate your HTML document using the W3C validator found at validator.w3.org and fix any errors that are identified.

Apply Your Knowledge continued

- 8. Submit the files in a format specified by your instructor.
- 9. In this exercise, you specified three source files, one of which was an Ogg file. Use your browser to research free resources available to convert an audio file to the Ogg format. Summarize your findings and provide a link to at least three different resources.

Extend Your Knowledge

Extend the skills you learned in this chapter and experiment with new skills. You may need to use additional resources to complete the assignment.

Using Multimedia Software

Instructions: In this exercise, you download and use multimedia software. If you use a Windows PC, download Windows Movie Maker. Use multimedia software to create a short video. If you use a Mac and have iMovie on your system, use iMovie to complete this project.

Perform the following tasks:

- 1. If you use a Windows PC, download and install Windows Movie Maker at windows.microsoft. com/en-us/windows-live/movie-maker.
- 2. Research how to use the software you installed.
- 3. Use the software to create a short video (30 seconds or less).
- 4. Your video should include, at a minimum, images, sound, and captions or titles.
- 5. Save the movie in the MP4 file format.
- 6. View your video in your preferred video player.
- 7. Submit the video in a format specified by your instructor.
- 8. In this exercise, you learned how to use multimedia software. Describe the next steps you would take to convert the video file you created to accommodate all major browsers.

Analyze, Correct, Improve

Analyze a website, correct all errors, and improve it.

Improving a Webpage with Video and Audio

Instructions: Work with the analyze09.html file in the analyze folder, the analyzestyles09.css file in the analyze\css folder, and the media files in the analyze\media folder from the Data Files for Students. The analyze09.html webpage needs multimedia to advertise its services. Insert a video element with appropriate source files to accommodate all major browsers. Set the video to play automatically, continuously, display controls, and specify width and height. Next, insert an audio element that continuously plays and does not display controls. Finally, style the video element. Use Figure 9–26 as a guide to correct these files.

1. Correct

- a. Open the analyze09.html file in your text editor from the Data Files for Students. Modify the comment at the top of the page to include your name and today's date and then link the file to analyzestyles09.css.
- b. Open the analyzestyles09.css file in your text editor from the Data Files for Students and then modify and correct the comment at the top of the document to include your name and today's date.
- c. Review the video element within the analyze09.html file and add appropriate attributes to specify that the video plays automatically when the page loads, displays controls, and plays continuously. Also specify a width and height for the video. (*Hint:* Review the video's properties.)

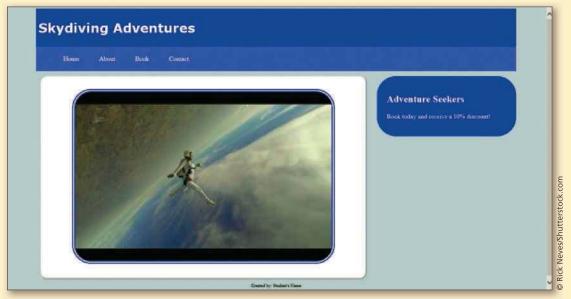


Figure 9-26

- d. Modify the video element to enable the video to play with all major, modern browsers and provide fallback text for legacy browsers that do not support the video element.
- e. Wrap the video element in a div element with a class value of video.

2. Improve

- a. In the analyze09.html file, add an audio element below the div element you just created.
- b. Specify three audio source files and provide fallback text for legacy browsers that do not support the audio element.
- c. Set the audio to play automatically and continuously.
- d. In the analyzestyles09.css file, create a style rule for the video class to center the div element and to apply 1em of padding. Include a comment for the style rule.
- e. Create a style rule for the video element and specify a border with rounded corners. Include a comment for the style rule.
- f. Save all of your changes and open the analyze09.html in your browser.
- g. Validate your CSS file using the W3C validator found at http://jigsaw.w3.org/css-validator/and fix any errors that are identified.
- h. Validate your HTML webpage using the W3C validator found at validator.w3.org and fix any errors that are identified.
- i. \(\mathhbar{3}\) Identify the steps you would take to accommodate legacy browsers when including a Flash video on a webpage.

In the Labs

Labs 1 and 2, which increase in difficulty, require you to create webpages based on what you learned in the chapter; Lab 3 requires you to dive deeper into a topic covered in the chapter.

Lab 1: Adding Video to the New Harvest Food Bank Website

Problem: You volunteer at a local food bank called New Harvest Food Bank that collects community food donations and provides food and other services to those in need. You have already created a responsive website but now need to add a video to the home page to attract volunteers. You also need to format the video on the page. Create and style the video as shown in Figure 9–27.

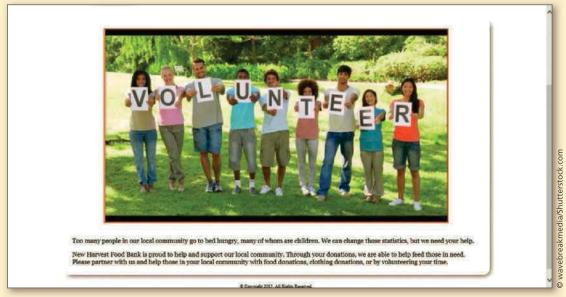


Figure 9-27

Instructions: Perform the following tasks:

- 1. Save the lab1\media folder from the Data Files for Students to your lab1 folder.
- 2. Open your text editor and then open the index.html document in the lab1 folder and update the comment at the top of the page to include today's date.
- 3. In the main element of the index.html document, insert a video element with autoplay and controls attributes.
- 4. Specify the following three source files within the **video** element and insert the appropriate **type** attribute and value for each source.

```
src="media/volunteer.mp4"
src="media/volunteer.ogg"
src="media/volunteer.webm"
```

- 5. Create a paragraph element below the source elements with text that states Your browser does not support the video element.
- 6. Wrap the video element in a **div** element and include a **class="video"** attribute and value.

7. In the fbstyles.css file, update the comment with today's date, and then create the following new style rules above the media queries and include comments for each:

```
.video {
text-align: center;
padding: 1em;
}

video {
border: 0.2em solid #FF6600;
height: auto;
max-width: 100%;
}
```

- 8. Save all files, open the index.html page within a browser, and then view the page, as shown in Figure 9–27.
- 9. Validate your HTML code and fix any errors.
- 10. Validate your CSS code and fix any errors.
- 11. Submit your assignment in the format specified by your instructor.
- 12. In this assignment, you wrapped a **video** element within a **div** element. Could the **video** element be wrapped within an HTML5 semantic element, such as an article, aside, or a section? Why or why not?

Lab 2: Adding Audio to the Website for Steve's Screen Services

Problem: You work for a screening company called Steve's Screen Services that specializes in screening, cleaning, and repairing screened patios. You have already created a responsive website, but now need to enhance the Gallery page with audio. You add and style an audio element for the Gallery page. The webpage with the audio element is shown in Figure 9–28.



Figure 9-28

In the Labs continued

Instructions: Perform the following tasks:

- 1. Save the lab2\media folder from the Data Files for Students to your lab2 folder.
- 2. Open the gallery.html document and update the comment at the top of the page to include today's date.
- 3. Insert an **audio** element below the **figure** element (within the **article** element) and set the element to automatically play the sound and to display the controls.
- 4. Insert three **source** elements and reference each audio file within the media folder. Insert the appropriate **type** attribute and value for each source.
- 5. Create a paragraph element with a fallback message to legacy browser users, notifying them that their browser does not support the audio element.
- 6. Wrap the audio element within a div element with the attribute and value class="sound".
- 7. In the screenstyles.css file, modify the comment at the top of the page to include today's date.
- 8. In the screenstyles.css file, create a new style rule for the **sound** class above the media queries that specifies a center alignment. Include a comment for this new style rule.
- 9. Save all files and open the gallery.html page within a browser to view the page, as shown in Figure 9–28.
- 10. Validate your HTML code and fix any errors.
- 11. Validate your CSS code and fix any errors.
- 12. Submit your assignment in the format specified by your instructor.
- 13. Research the Internet to locate free resources for audio files and provide a summary of your findings. Include at least three resource links.

Lab 3: Expand Your World

Exploring Multimedia File Formats and Plug-ins

Problem: In this chapter you worked with many types of file formats and learned about different kinds of multimedia. In this exercise, you will research to learn more about plug-ins.

Instructions:

- 1. Search the Internet for information about plug-ins.
- 2. Identify why plug-ins were created and how they are used by a browser.
- 3. Why are plug-ins losing their popularity?
- 4. What plug-ins are installed on your desktop browser?
- 5. What is Microsoft Silverlight?
- 6. Submit your answers in the format specified by your instructor.

Consider This: Your Turn

Apply your creative thinking and problem-solving skills to design and implement a solution.

1. Improving Your Personal Portfolio Website Design

Personal

Part 1: In Chapter 8, you added a table to your portfolio page. In this exercise, you update the page as follows:

- 1. Open the external style sheet you created in Chapter 8 and save it with the name portfoliostyles09.css in the styles folder within the your_turn_1 folder.
- 2. Add your name and the current date as a comment to the first line of the portfoliostyles09.css file.
- 3. In your HTML editor, open the files you created for your portfolio website in Chapter 8 and make sure your name and the current date are added as a comment to the first line of all files.
 - a. Update the HTML files to modify the link to the external style sheet to reference portfoliostyles09.css.
 - b. Insert an audio element on the portfolio page.
 - c. Determine which attributes to include for the audio element.
 - d. Include at least two source files. Audio files for this exercise are available in the your_turn_1\media folder in the Data Files for Students.
- 4. Save and test your files.
- 5. Validate and correct your HTML and CSS files, and submit your assignment in the format specified by your instructor.

Part 2: 🛞

Discuss the reasons you should or should not include audio on your portfolio webpage.

2. Improving the WebDevPros Website

Professional

Part 1: In Chapter 8, you added a table and a form for webDevPros. In this exercise, you update the site as follows:

- 1. Open the external style sheet you created in Chapter 8 and save it with the name webdevstyles09.css in the styles folder within the your_turn_2 folder.
- 2. Add your name and the current date as a comment to the first line of the webdevstyles09.css file.
- 3. Open the files you created for webDevPros in Chapter 8 in your HTML editor and make sure your name and the current date are added as a comment to the first line of all files.
- 4. Update the HTML files to:
 - a. Integrate a video element on the home page.
 - b. Set the video to play automatically and to display controls.
 - c. Set a width and height for the video element.
 - d. Specify two source elements. Video files are provided for this exercise in the your_turn_2\ media folder in the Data Files for Students.
 - e. Include fallback text in case the user's browser does not support the video element.
 - f. Wrap the video element within a div element with a class named ad.

Consider This: Your Turn continued

- 5. Update the CSS file to:
 - a. Style the div element to center the video on the page and to apply a small amount of padding.
 - b. Style the **video** element for responsive design and include a border, and rounded corners.
 - c. Add comments to note all changes and updates.
- 6. Save and test your files.
- 7. Validate and correct your HTML and CSS files, and submit your assignment in the format specified by your instructor.

Part 2: **%** In this exercise, you integrated two video file formats, MP4 and WebM. Search the Internet for an Ogg converter, create an Ogg file format, and add it as a third source file to the **video** element. Identify the website source that you used to create the Ogg file.

3. Improving the Dog Hall of Fame Website

Research and Collaboration

Part 1: In Chapter 8, you added a form to the Dog Hall of Fame website. In this exercise, you create and style a video for the website. Do the following activities as a group:

- 1. Open the external style sheet you created in Chapter 8 and save it with the name dogstyles09.css in the styles folder within the your_turn_3 folder.
- 2. Add your names and the current date as a comment to the first line of the dogstyles09.css file.
- 3. Open the files you created for the Dog Hall of Fame in Chapter 8 in your HTML editor and make sure your names and the current date are added as a comment to the first line of all files.
- 4. Modify the link to the external style sheet to reference dogstyles09.css.
- 5. Insert a video element on the nominations page. At a minimum, update the HTML page to do the following:
 - a. Integrate a **video** element on the page.
 - b. Set the video to play automatically and to display controls.
 - c. Set a width and height for the video element.
 - d. Specify three source elements. Video files are provided for this exercise in the your_turn_3\ media folder in the Data Files for Students.
 - e. Include fallback text in case the user's browser does not support the video element.
 - f. Wrap the video element within a div element with a class named dog.
- 6. Update the CSS file to:
 - a. Style the **div** element to center the video on the page and to apply a small amount of padding.
 - b. Style the **video** element for responsive design and include a border, rounded corners, and a box shadow effect.
 - c. Add comments to note all changes and updates.
- 7. Save and test your files.
- 8. Validate and correct your HTML and CSS files, and submit your assignment in the format specified by your instructor.

Part 2:
As a group, discuss the benefits of uploading a video file to YouTube and then linking the YouTube video to a website. List the steps you would take to accomplish this task.