Multimedia Notes:

Midi Files:

-Musical Instrument Digital Interface (MIDI) protocol

-files are small

-not a specification for sampled digital audio; it is a bank of digitized sounds and control information for replaying the file (similar to electronic synthesizer)

Wav Files:  
-proprietary to Microsoft and IBM (windows-based pcs)

Mp3 and Mp4 Files:

MP3 files are CD quality songs that are compressed down from huge files to much smaller files without any noticeable decrease in sound quality.

MP3 stands for MPEG 1 (Motion Picture Experts Group) Layer 3.

MP3 squishes down sound files 10-15 times smaller than a parent file. And yet that tiny .mp3 file retains most of the perceived audio fidelity by stripping out the waveforms that the human ear doesn't process.

MP4 is short for Moving Picture Expert Group-4.

considered the gold standard for all types of streaming and broadcast applications online

You should generally indicate the size of the mp3 or mp4 file on your web page, next to the link for it.

Embeding audio-clips:

Embedding an audio file on a web page means that the viewer's sound controller will be loaded on the page, at the time that the browser loads the page.

Midi, wav, and mp3 files can all be embedded on a web page.

Will not validate for XHTML:

<embed src="beethoven.mid" autostart="false" loop="false" width="20%" height="20" />

autostart="false" - means that the midi file will NOT start playing automatically right after the page is loaded. This is generally a good idea. The viewer must click the start button on the controller for the file to play.

loop="false" - means that the midi file will not keep playing continuously after the viewer clicks start.

width="20%" - means that the controller will occupy 20% of the viewer's screen width.

heigth="20" -means the controller will be 20 pixels high.

Will validate for XHTML:

You can also embed an audio file using just the **object tag**. The object tag **will validate** for XHTML. See the first example on this [page](http://fog.ccsf.edu/~srubin/h4Object.html). Here is the coding:

<object classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B"codebase="http://www.apple.com/qtactivex/qtplugin.cab" width="200" height="16">

<param name="src" value="jungle.mp3" />

<param name="controller" value="true" />

<param name="autoplay" value="false" />

<param name="autostart" value="0" /> <param name="pluginspage"value="http://www.apple.com/quicktime/download/" />

<object type="audio/x-mpeg" data="jungle.mp3" width="200" height="16"> <param name="src" value="jungle.mp3"/>

<param name="controller" value="true" />

<param name="autoplay" value="false" />

<param name="autostart" value="0" /> <param name="pluginurl"value="http://www.apple.com/quicktime/download/" />

</object>

</object>

**Note:** classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" is necessary for the audio file to play in an IE controller. IE needs a non-standard value to the valid classid attribute, an identifier to load an associated activeX.

1. If you embed an audio file with the embed tag, I suggest that you also provide a link to it. Here is an example of a[linked midi file](http://fog.ccsf.edu/~srubin/beethoven.mid). This will bring up an audio controller outside of the web page, and the midi file will play automatically.

3. For HTML5, the classid tag won't validate. This example's coding will validate for HTML5.

<object type="audio/x-mpeg" data="jungle.mp3" width="200" height="16"> <param name="src" value="jungle.mp3">  
<param name="controller" value="true">  
<param name="autoplay" value="false">  
<param name="autostart" value="0">   
<param name="pluginurl" value="http://www.apple.com/quicktime/download/">  
</object>

4. For HTML5, you can use the **audio tag**, for example:

<audio controls="controls">  
<source src="jungle.mp3" type="audio/mpeg" />  
<source src="jungle.ogg" type="audio/ogg" />  
Your browser does not support this audio  
</audio>

I needed to convert jungle.mp3 into an ogg file for use with this tag. The OGG Converter (allows a Free Trial, but costs 19.95 to buy it).  
  
Converts MP3, WAV, WMA, OGG from one to another  
Supports Resampling of MP3, WAV, WMA, OGG Vorbis

You can also download a [free audio converter](http://www.freemake.com/free_audio_converter/).

5. For HTML5, the embed tag will not validate. You should use the audio tag or the object tag.

Mpeg Files:

MPEG, which stands for Moving Picture Experts Group, is the name of family of standards used for coding audio-visual information (e.g., movies, video, music) in a digital compressed format. The major advantage of MPEG compared to other video and audio coding formats is that MPEG files are much smaller for the same quality. This is because MPEG uses very sophisticated compression techniques. MPEG file extensions can be .mpg, .mpeg, or .mpe.

MP4 is a multimedia container format standard specified as a part of MPEG-4. It is most commonly used to store digital video and digital audio streams, especially those defined by MPEG, but can also be used to store other data such assubtitles and still images.  MPEG-4 allows streaming over the Internet. A separate hint track is used to include streaming information in the file. The only official filename extension for MPEG-4 files is**.mp4**.

Avi Files:

AVI stands for Audio Video Interface. AVI is used in Windows operating systems to provide sound and video.

The file extension for AVI files is .AVI.

Quicktime Files:

-Apple software for creating, playing and streaming digital media for Mac OS and Windows.

Ext: .qt or .mov

WMV Files:

-Windows Media Video (WMV) is a generic name for the set of video codec technologies developed by Microsoft.

Video version 9 codec and submitted it to SMPTE for standardization.

The video stream is often combined with an audio stream of Windows Media Audio and encapsulated in Advanced Systems Format files, carrying the .wmv or .asf file extensions. WMV files are played by players such as MPlayer or Windows Media Player, the latter being only available for Microsoft Windows and Macintosh systems.

Embedding Video Clips:

Embedding a video file on a web page means that the viewer's associated video controller will be loaded onto the page, at the time that the browser loads the page.

<embed src="julia.qt" height="140" width="160" autostart="false" loop="false" controller="true" />

autostart="false" means that the quicktime file will NOT start playing automatically right after the page is loaded. This is generally a good idea. The viewer must click the start button on the video controller for the file to play.

height="140" - means the controller will be 140 pixels high.

width="160" - means that the controller will be 160 pixels wide.

loop="false" - means that the quicktime file will not keep playing continuously after the viewer clicks start.

controller="true" - This adds user controls to the movie.

 A very common error in the failure to display the controller for a video file is not declaring the width and height to be large enough.

 If you are going to embed a video file I suggest that you also link to it.

 You can embed a video file using just the object tag. The object tag will validate for XHTML

 For HTML5, you can use the video tag, for example:

<video width="200" height="150" controls="controls">  
<source src="chimp.mpeg" type="video/mpeg">  
<source src="chimp.ogg" type="video/ogg">  
Your browser does not support the video tag.  
</video>

I needed to convert chimp.mpeg into an ogg file for use with this tag. The OGG Converter (allows a Free Trial, but costs 19.95 to buy it).  
  
Converts MP3, WAV, WMA, OGG from one to another  
Supports Resampling of MP3, WAV, WMA, OGG Vorbis

 The HTML5 <video> element also has methods, properties, and events.

There are methods for playing, pausing, and loading, for example. There are properties (e.g. duration, volume, seeking) that you can read or set. There are also DOM events that can notify you, for example, when the <video> element begins to play, is paused, is ended, etc.

 For HTML5, the embed tag will not validate. You should use the video tag or the object tag. If you use the object tag, you must use the type attribute.  
  
For example:  
  
<object height="136" width="155" type="video/qt">  
<param name="src" value="julia.qt" />  
<param name="controller" value="true" />  
<param name="autostart" value="false" />  
  
<object type="video/quicktime" data="julia.qt" height="136" width="155" class="qt">  
<param name="controller" value="true" />  
<param name="autostart" value="false" />  
Error text.  
</object>

You can also use the Quick Time Object (QTObject) for embedding video files. It works cross browser and cross platform.

Java vs JavaScript

Actually, the 2 languages have almost nothing in common except for the name. Although Java is technically an interpreted programming language, it is coded in a similar fashion to C++, with seperate header and class files, compiled together prior to execution.

Java has been generating a lot of excitment because of its unique ability to run the same program on IBM, Mac, and Unix computers.

A Java applet is a program written in the JavaTM programming language that can be included in an HTML page, much in the same way an image is included. When you use a Java technology-enabled browser to view a page that contains an applet, the applet's code is transferred to your system and executed by the browser's Java Virtual Machine (JVM).

Java Applet Tag

For a java applet to work on a web page, it must already have been compiled. A compiled java applet normally has a file extension of class.

Here is an example of a simple applet tag:

<applet code="myApplet.class" width="100" height="140">

</applet>

Here's a more complex example of an APPLET tag:

<applet code="flame.class" width="250" height="110">

<param name="text" value="Bungle+in+the+Jungle" />

<param name="link" value="http://javaboutique.com" />

</applet>

This tells the viewer or browser to load the applet whose compiled code is flame.class. The first param statement sets the applet's "text" attribute (which customizes the text this applet displays) to be "Bungle+in+the+Jungle". A second param statement will allow the applet to also act as a link, if one clicks the text at the bottom of the applet. If the page is viewed by a browser that can't execute Java applets, then the browser will ignore the APPLET and PARAM tags, displaying only the HTML between the <param /> and </applet> tags (the alternate HTML).

Object Tag

Note that the applet tag is valid for Transitional XHTML, but not for Strict XHTML because the applet tag has been deprecated. Instead you can use the object tag, which is valid for Transitional and Strict XHTML

<!-- first object tag is for Firefox -->  
<object classid="java:flame.class" type="application/x-java-applet" height="200" width="200">  
<param name="text" value="Bungle+in+the+Jungle" />

<!-- 2nd object tag is for IE -->

<object classid="clsid:8AD9C840-044E-11D1-B3E9-00805F499D93" height="200" width="200">   
<param name="code" value="flame" />  
<param name="text" value="Bungle+in+the+Jungle" />  
</object>   
</object>

The **classid** attribute identifies which version of Java Plug-in to use.

For Firefox - classid="java:classname.class"  
  
(where classname.class should be replaced by the actual name, e.g.,   
classid="java:flame.class"  
  
  
For IE - classid="clsid:8AD9C840-044E-11D1-B3E9-00805F499D93"  
(always use this classid for IE, but note that it may not display the applet in some versions/platforms for IE)  
 **Note: I suggest that you upload your class file and all associated folders and files directly into the directory that contains your hw5.html file.**

For HTML5, the applet tag and the classid attribute are not valid. Instead, you should use the object tag with a param statement that has name="code" and value equaling the class name of the applet.

Here's a valid HTML5 example:

<object type="application/x-java-applet" height="200" width="200">  
<param name="text" value="Bungle+in+the+Jungle">  
<param name="code" value="flame.class" >  
</object>

Frames:

A frame is a division of a browser window which displays a distinct HTML page. A browser window can be divided into multiple frames, with each frame displaying a different HTML page.

Frames are used to help users easily find their way through tangled or difficult-to-navigate content (i.e. multiple chapters in an online book with numerous hyper-referenced footnotes). Frames provide a means for a persistent navigation strip to be available to a user at all times, while providing additional areas for display of content, whether primary or ancillary.

Downside:

Sub-pages within a frameset may not be bookmarked,

Due to the multiplicity of pages involved in a single frame site, the number of simultaneous hits received by a server for the first frame page is greatly increased, causing problems for sites with heavy usage

With a three-frame frames page, for instance, the server receives four hits, instead of one hit as with a regular HTML page, quadrupling traffic for that first page;

Search engines tend to catalog sub-pages within a frame site, bringing up sub-pages for users without a frameset or any context. There is a workaround for this problem, but it is an awkward one.

Many inexperienced web programmers (and some experienced ones) use frames to "keep" or "trap" their users within the confines of their frameset; this has become one of the biggest evils in contemporary web design. Marketing people, especially, wish to "brand" the web within a frame containing their company logo, forcing users to surf within this restricted web browser window space from which there is no escape.

Any link which leads out of your web site MUST erase any framesets in use.

Conclusion:

One should use frames only when absolutely necessary and avoid using them whenever possible.

Inline Frames:

An inline frame (iframe) cannot be viewed on all browsers. Internet Explorer and Firefox are two of the browsers that handle inline frames correctly.

<iframe> tag allows you to create a frame that can appear anywhere within a standard html document

used:

utilizing page sidebars

highlighting related external elements

you can target links to open other pages in the iframe

You must create a separate html document to contain the info tha will be placed in the inline frame

<iframe src="ifra.html" width="20%" height="70" align="left" name="ifra"></iframe>

The align tag allow you to set the justification for the frame within the core html page, just as if you were aligning an image.

The name attribute allows you to name the frame. This name allows you to target elements, such as a new page or an image, in the <a href....> statement on the current page.

How to put html code to appear in a frame? ie an image:

<a href="croc.jpg" target="ifr">clicking here</a>

frameborder attribute determines if a border of the frame will be visible and if so what size (“0” is invisible)

Note: For browsers that do not support inline frames, you should place text after the iframe tag, but before the /iframe tag, to tell the viewer that their browser does not support inline frames.

Example:

<iframe src="ifra1.html" width="300" height="150" align="right" name="ifr">  
<!-- Here's what would show using a browser that does not support inline frames -->  
Your browser does not support inline frames  
</iframe>