



Technology Institute for Music Educators

TI:ME COURSE 2B Internet Authoring

Syllabus, Workbook, and Appendices
By Steven Estrella

Edited by Steven Estrella and Floyd Richmond (2006)

Copyright © 2006 Technology Institute for Music Educators
<http://www.ti-me.org>

These materials were made possible by a grant from
NAMM (The International Music Products Association)

Table of Contents

Syllabus	4
Presentations and Activities	
Topic 1: History of the Internet and Music Education	7
Topic 2: How the Internet Works	8
Internet Terminology Worksheet.....	10
Topic 3: XHTML – Structuring Web Content	11
Introduction to Internet Authoring Software.....	12
A Basic XHTML Document	13
Unordered Lists.....	14
Ordered Lists.....	15
Links within a Page	17
Links to Other Pages	19
Links to Other Sites.....	20
Physical vs. Logical Styles	22
Preformatted Text	24
Tables.....	25
Horizontal Rules	27
Special Characters.....	29
Images.....	31
Framesets	33
Topic 4: CSS – Styling Web Content	37
Inline Styles	38
Internal Style Sheets.....	39
External Style Sheets.....	40
Creating an External Style Sheet	43
CONFLICTING RULES.....	45
ELEMENT CLASSIFICATION	46
ELEMENT BOX.....	47
Topic 5: Linking to Media Files	48
Text.....	48
Images.....	49
Browser Plugins	50
Files that Require Separate Applications	51

Topic 6: Embedding Media Files	52
Embedding Images.....	52
Embedding Flash Media.....	53
Embedding Video with QuickTime.....	55
Embedding Audio with QuickTime.....	56
Embedding Media with the QuickTime Poster Technique.....	57
Embedding Notation with the Scorch Plugin.....	58
Topic 7: Uploading to the Web	60
Topic 8: Dynamic Content	61
Rollover Image Swapping with Packaged Behaviors.....	61
ROLLOVER IMAGE SWAPPING WITH CUSTOM CODE.....	62
Popup Messages with Packaged Behaviors	63
Jump Menus with Packaged Behaviors	64
Client-Server Technologies (PHP, ASP, JSP)	65
Online Databases.....	66
Topic 9: Enhancing Your Site with Online Services.....	67
Blogging	67
Feedback with Form Mail	69
Counters.....	70
Guestbooks.....	71
Chat Rooms.....	72
Discussion Groups (list servers).....	73
Support Sites for Teachers.....	74
Calendars	75
Course Learning Systems (Blackboard, Web CT, etc.).....	76
Site Search	78
Topic 10: Final Project: Music Education Web Site.....	79
Storyboarding.....	80
File organization	81
Create and Upload.....	82
Maintenance.....	83
Appendices	
Brief History of the Internet and Web Audio.....	84
Web Audio Today	94
Web Sites with Audio for Music Education	100
Copyright Concerns for Web Developers	102
XHTML Basics.....	105
CSS Basics.....	108

Syllabus

Course Objective:

The objective of this course is to impart practical skills and knowledge to In-Service Teachers (ISTs) to allow them to integrate Internet authoring effectively into teaching and learning. The course covers the creation of Web sites for the music classroom. ISTs who complete this course will create web sites that include lessons and resources appropriate for the K-12 music classroom. ISTs will incorporate digital media including text, graphics, video, sound, and music into their projects. Software to be used in the class will include a professional level Web authoring program and software needed to edit media files. The format of course alternates presentations with class activities, many of which serve to assess student progress. Satisfactory participation in class activities and successful completion of final projects are required for certification. This course meets one third of the coursework requirements for Level Two TI:ME Certification. Prerequisites for this course include completion of both TI:ME level 1 courses OR equivalent experience.

Additional Information:

Internet Authoring (TI:ME 2B) is offered as either a two-credit or three-credit graduate course. Topics shown in ALL CAPS need not be included in the course when it is offered for two graduate credits only. The instructor of the course must be approved by TI:ME and an expert in teaching Internet authoring skills to in-service music teachers. It is strongly recommended that the class size be limited to allow for one IST per computer and no more than sixteen ISTs per instructor. Additional ISTs may be accommodated if computer workstations and assistants are available. Each IST will need approximately 20 hours working alone on a workstation in order to complete the final projects and class activities. The outline is designed for 25-hour units within a 2-credit graduate workshop. Items in UPPERCASE can be added if the course is offered for 3 graduate credits.

Hardware Requirements:

A computer lab/classroom consisting of multimedia computers connected to MIDI keyboards is the standard recommended configuration. Printers must be available for output of projects. The teacher's station should have a projection device and sound system to allow the instructor's screen to be viewed by the entire class. When possible, a sound system connecting all MIDI devices in the room should be used to allow for class activities. High speed Internet connections are necessary for each participant.

Software Requirements:

Internet authoring software that produces W3C-compliant code is required. As of 2006, Adobe/Macromedia DreamWeaver and Adobe GoLive are the recommended choices. Media editing software is also required for basic editing of images and sounds. Video editing software is optional.

Recommended Texts:

Technology Strategies for Music Education by Rudolph, Richmond, Mash, Webster, Bauer, and Walls. Second edition (2005) edited by Floyd Richmond; Hal Leonard Publications ISBN 0-634-09060-7.

Instructors will likely wish to require a text that provides tutorials and help with the Internet authoring software chosen for the class. The Missing Manual series has good books on many software products. See <http://www.missingmanual.com/>.

Prerequisites:

TI:ME Level 1 certification or equivalent experience is prerequisite to taking this course.

Learning Objectives:

ISTs will acquire the following declarative and procedural knowledge during this course.

Declarative Knowledge:

The IST demonstrates an understanding of:

1. Domain names
2. Hosting
3. File Management
4. Client vs. Server
5. Dynamic vs. Static pages
6. FTP for uploading content to a site
7. XHTML and HTML
8. CSS for styling Web pages
9. Forms
10. JavaScript and DHTML
11. Media linking and embedding (Flash, QuickTime, etc.)
12. Blogging
13. Virtual classroom technologies
14. Ways to use Web sites to enhance teaching and learning
15. Copyright issues applicable to Internet authoring

Procedural Knowledge:

The IST demonstrates basic skills in using an Internet authoring program to produce classroom materials. The skills to be acquired include the following:

- a. Creating Web pages using XHTML and CSS
- b. Organizing Web pages into an original Web site
- c. Embedding and linking images, audio, animation, and video
- d. Creating interactive elements such as online tests and games

Final Projects:

1. The IST completes all classroom assignments.
2. The IST creates an interactive, multimedia Web site that is useful in music teaching and contains text, images, audio, and interactive elements such as an online test or game.

Topic 1: History of the Internet and Music Education

The instructor will present an outline of the history of the Internet with special attention to how the music education community has used Internet resources for teaching and learning in music. See the Brief History of the Internet and Web Audio in the appendices.

Class Discussion:

ISTs will share their early experiences with the Internet. At what point in the history of the Web did it become useful to music teachers?

Activity 1.1: Browsing Music Education Web Sites

ISTs will research and browse web sites created by and for music teachers. ISTs will report back to the class on their findings and discuss the common features found on music education Web sites. A short list of appropriate sites is found at the workbook resource page at: <http://www.ti-me.org/members/2BInternet/>. Login as members and use the current password for the ti-me.org members-only section.

Notes: (use this space for note taking)

Topic 2: How the Internet Works

The instructor will present a lecture covering how the Internet works. On the workbook resource page at: <http://www.ti-me.org/members/2BInternet/>, the instructor and ISTs will find links to Web sites that explain Internet infrastructure. The lecture and discussion should cover the following:

What is the Internet?

The Internet is a network of computers. Every computer connected to the Internet has a unique number assigned to it by its Internet Service Provider (ISP). This number is known as an Internet Protocol (IP) address and consists of a series of numbers such as 66.235.219.164. Early on, it was discovered that users didn't like having to remember a series of numbers, so a domain name system was created to link the numerical address of each web server to a name such as ti-me.org.

Example:

When an Internet user types ti-me.org into a web browser, the visitor's Internet Service Provider (ISP) takes note of the user's computer IP address. The ISP then passes on the requested domain name to a computer called a domain name server (DNS). The DNS maintains a list of all domain names and their IP addresses. The DNS finds the correct IP address for ti-me.org. That allows the ISP to direct the user's request to the appropriate computer (in this case, TI:ME's Web server at 66.235.219.164). TI:ME's Web server then sends a page of content to the user's computer by way of the user's ISP.

Domain Names

The Internet Corporation for Assigned Names and Numbers (ICANN at icann.org) manages domain names and IP addresses worldwide. Many companies, such as GoDaddy.com, work with ICANN to register domain names. These companies are called Domain Name Registrars. Domain names are registered for a fee (\$9 per year is typical in 2006). To have a Web site on the Internet with a unique domain name requires registration with a domain name registrar. An alternative is to locate your Web site on an existing domain. For example, the Web site that supports this workbook does not have its own domain. Instead it is part of the ti-me.org domain and located at <http://www.ti-me.org/members/2BInternet/>. Login as members and use the current password for the ti-me.org members-only section.

Hosts, Clients, and Servers

A host (or server) is a computer that makes documents available for other computers (clients) to download over the Internet. A hosting company rents hard drive space on their servers to anyone who wishes to have a Web site. Commercial

hosting space costs as little as \$50 per year in 2006. Free hosting space for teachers is often available through their school districts. Free hosting space is also available from hosting services that insert advertisements in your pages. Ad-supported hosting space is usually a bad idea for education web sites because the teacher has no control over the content of the ads.

Online Services

Many servers on the Internet do much more than merely store and display Web pages. These servers provide online services such as blogs, message boards, discussion groups, various forms of social networking, interactive games, quizzes, and surveys. Online services are exciting additions to traditional, static Web sites.

Activity 2.1:

ISTs will spend 30 minutes browsing Web sites that explain Internet infrastructure. ISTs will then have a class discussion to share their questions and clear up misconceptions about how the Internet works. ISTs will complete the worksheet below on Internet terms and concepts.

Internet Terminology Worksheet

1. A _____ is a computer on the Internet that makes documents available for other computers to download.
2. A _____ is a computer on the Internet that downloads documents from other computers.
3. The term _____ refers to a unique number assigned to each computer that communicates on the Internet.
4. The term _____ refers to a company that provides a unique number for a customer's computer to allow that computer to communicate on the Internet.
5. The term _____ refers to addresses on the Internet expressed in text form such as ti-me.org, musictheory.net, or aol.com.
6. The term _____ refers to a computer that translates numbers like 66.235.219.164 into names like ti-me.org.
7. The term _____ refers to software that downloads and displays content from the Internet.
8. The term _____ refers to an online journal.
9. The term _____ refers to hard drive storage space for Internet content.
10. The symbol _____ is an essential part of an _____ address. The words before the symbol constitute the _____. The words after the symbol constitute the _____.

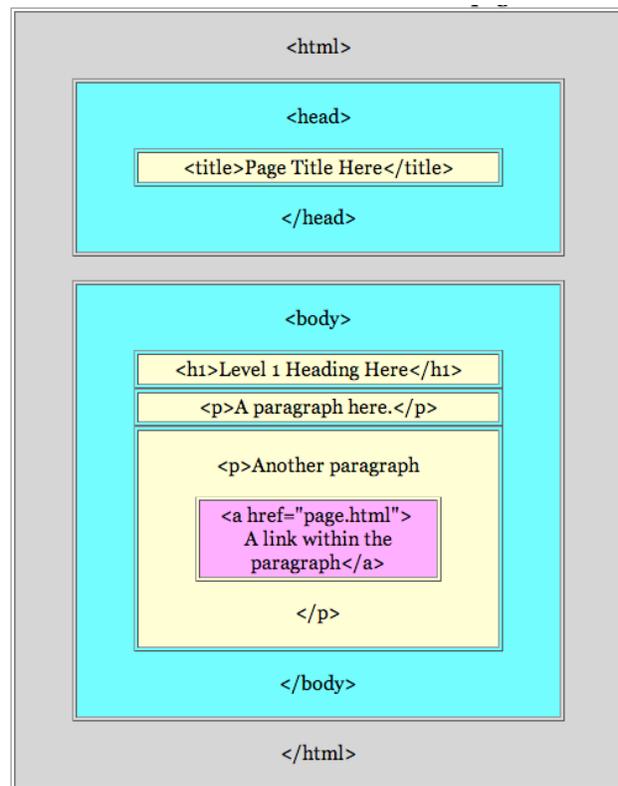
Notes:

Topic 3: XHTML

The instructor will explain the concepts behind HTML and XHTML including Document Type Definitions (DTD) and the role of the World Wide Web Consortium (W3C) in maintaining standards. ISTs will then begin a series of activities to create XHTML pages and view them locally. The instructor will explain concepts as they are introduced in each lesson. Instructors and ISTs should read “XHTML Basics” in the appendices.

Web pages can be created by many different types of software programs from the humblest text editors like TextEdit or TextWrangler (Mac) and NotePad or NoteTab Lite (Windows) to full-featured Web site development tools like Dreamweaver and GoLive. Regardless of how you create the file, the file itself is a plain text file that can be read by any computer. One of the original purposes of the Web was to allow different computer platforms to share files, so plain text was a great choice to ensure compatibility.

Within an XHTML file you will find content surrounded by special markup tags. For example, text that is intended to be displayed as **bold** will be surrounded by the markup tags `` and ``. These codes tell the Web browser that reads the page where to begin boldface and where to end it. All content on a Web page is contained by one or more tags. This is the **Principle of Containment**. One way to ensure that your content is properly contained is to type the opening and closing tags before you insert your content. In older version of HTML, like HTML 4.01, tags can be upper or lower case letters. XHTML, the very latest version of HTML, requires lowercase tags.



Tag Hierarchy of a Web Page

Activity 3.1: Introduction to Internet Authoring Software

Internet Authoring software will automatically generate almost all of the code shown in this workbook. Throughout the workbook, code is included for the sake of illustration. Generating code from scratch won't be required, but studying, understanding, and editing it might permit some operations not supported by Internet Authoring software.

The instructor will show ISTs how to use Internet Authoring software to:

1. Establish a site folder called “mysite” to store new documents.
2. Create a new page using the XHTML Transitional DTD.
3. Edit a page in “Design View” using a graphical editing environment similar to a word processor.
4. Edit a page in “Code View” and learn how the code relates to the final appearance of the document in the Web browser.
5. Preview a page locally using any W3C-compliant Web browser (FireFox, Mozilla, and Safari are recommended.)

Notes:

Activity 3.2: A Basic XHTML Document

ISTs will create the first of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “A Basic XHTML Document” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson01.htm” in the mysite folder created in Activity 3.1.

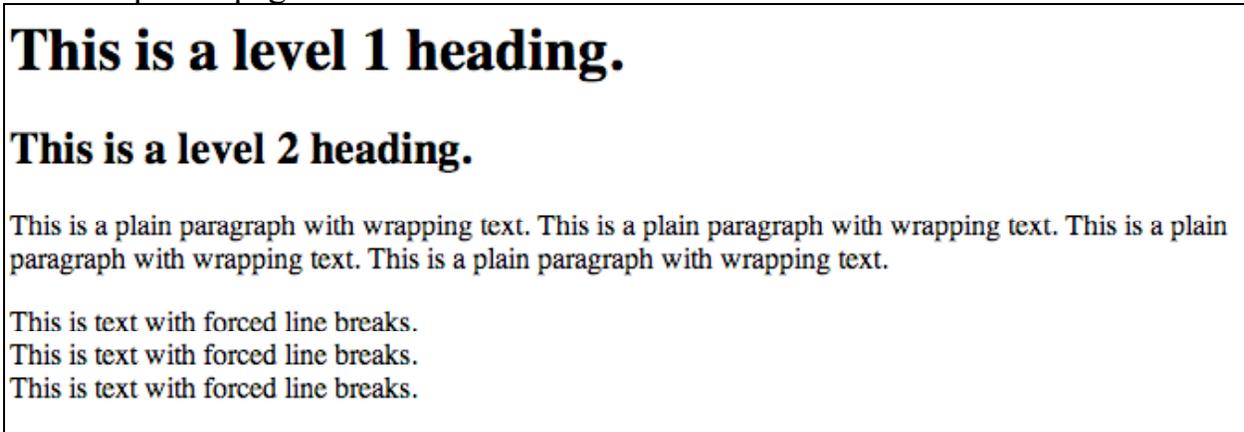
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Basic XHTML Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>This is a level 1 heading.</h1>

<h2>This is a level 2 heading.</h2>

<p>This is a plain paragraph with wrapping text. This is a plain paragraph
with wrapping text. This is a plain paragraph with wrapping text. This is a
plain paragraph with wrapping text.</p>

<p>
This is text with forced line breaks.<br />
This is text with forced line breaks.<br />
This is text with forced line breaks.<br />
</p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.



Notes:

Activity 3.3: Unordered Lists

ISTs will create the second of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Unordered Lists” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson02.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Unordered Lists</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Unordered Lists</h1>
<p>When you need to present a list of items and the order is not meaningful,
use an unordered list like the one below.</p>
<p>Composers I like:</p>
<ul>
<li>Gesualdo</li>
<li>Bach</li>
<li>Mozart</li>
<li>Beethoven</li>
</ul>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Unordered Lists

When you need to present a list of items and the order is not meaningful, use an unordered list like the one below.

Composers I like:

- Gesualdo
- Bach
- Mozart
- Beethoven

Notes:

Activity 3.4: Ordered Lists

ISTs will create the third of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Ordered Lists” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson03.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Ordered Lists</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Ordered Lists</h1>
<p>When you need to present a list of items and the order is meaningful, use
an ordered list like the one below.</p>
<p>Step-by-step procedure for converting CD tracks to MP3 files:</p>
<ol>
<li>Insert CD into computer.</li>
<li>Launch iTunes.</li>
<li>Set import format preference to MP3.</li>
<li>Set quality preference to 128 kbps stereo. </li>
<li>Select Convert to MP3 from the Advanced menu.</li>
</ol>
<p>When you wish to present a hierarchical outline, specify the type of
number used for each level of the list.</p>
<p>Music History before 1750:</p>
<ol type="I">
<li>Medieval (500-1450)
<ol type="A">
<li>Gregorian Chant</li>
<li>Organum and early polyphony</li>
<li>Motets</li>
</ol>
</li>
<li>Renaissance (1450-1600)
<ol type="A">
<li>Cantus Firmus Masses (Palestrina)</li>
<li>Madrigals (Willaert, Lassus, Gesualdo)</li>
</ol>
</li>
```

```
<li>Baroque (1600-1750)
<ol type="A">
<li>Monody (Early Baroque - Caccini, Peri, Monteverdi)</li>
<li>Complex Counterpoint (Late Baroque)</li>
<li>Concerto Grosso (Bach, Vivaldi)</li>
</ol>
</li>
</ol>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Ordered Lists

When you need to present a list of items and the order is meaningful, use an ordered list like the one below.

Step-by-step procedure for converting CD tracks to MP3 files:

1. Insert CD into computer.
2. Launch iTunes.
3. Set import format preference to MP3.
4. Set quality preference to 128 kbps stereo.
5. Select Convert to MP3 from the Advanced menu.

When you wish to present a hierarchical outline, specify the type of number used for each level of the list.

Music History before 1750:

- I. Medieval (500-1450)
 - A. Gregorian Chant
 - B. Organum and early polyphony
 - C. Motets
- II. Renaissance (1450-1600)
 - A. Cantus Firmus Masses (Palestrina)
 - B. Madrigals (Willaert, Lassus, Gesualdo)
- III. Baroque (1600-1750)
 - A. Monody (Early Baroque - Caccini, Peri, Monteverdi)
 - B. Complex Counterpoint (Late Baroque)
 - C. Concerto Grosso (Bach, Vivaldi)

Notes:

Activity 3.5: Links within a Page

ISTs will create the fourth of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Links within a Page” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson04.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Named References and Links</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1><a id="pagetop" name="pagetop">Links within a Page</a></h1>
<p>Links to content found on the same page are accomplished using named
references within an anchor tag. To recreate this document, copy and paste
any text you choose to create a long document with a named reference at the
bottom and a link to that reference at the top. </p>
<p><a href="#adage">Go to adage at the end</a></p>
<p>Type any text you like below. Copy and paste it as many times as you like
to create a really long page.</p>
<p>Type any text you like below. Copy and paste it as many times as you like
to create a really long page.</p>
<p><a id="adage" name="adage">Here is the adage:</a></p>
<p>All work and no play made Johann Sebastian Bach a very prolific
composer.</p>
<p><a href="#pagetop">Return to Top of Page</a></p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Links within a Page

Links to content found on the same page are accomplished using named references within an anchor tag. To recreate this document, copy and paste any text you choose to create a long document with a named reference at the bottom and a link to that reference at the top.

[Go to adage at the end](#)

Type any text you like below. Copy and paste it as many times as you like to create a really long page.

Type any text you like below. Copy and paste it as many times as you like to create a really long page.

Type any text you like below. Copy and paste it as many times as you like to create a really long page.

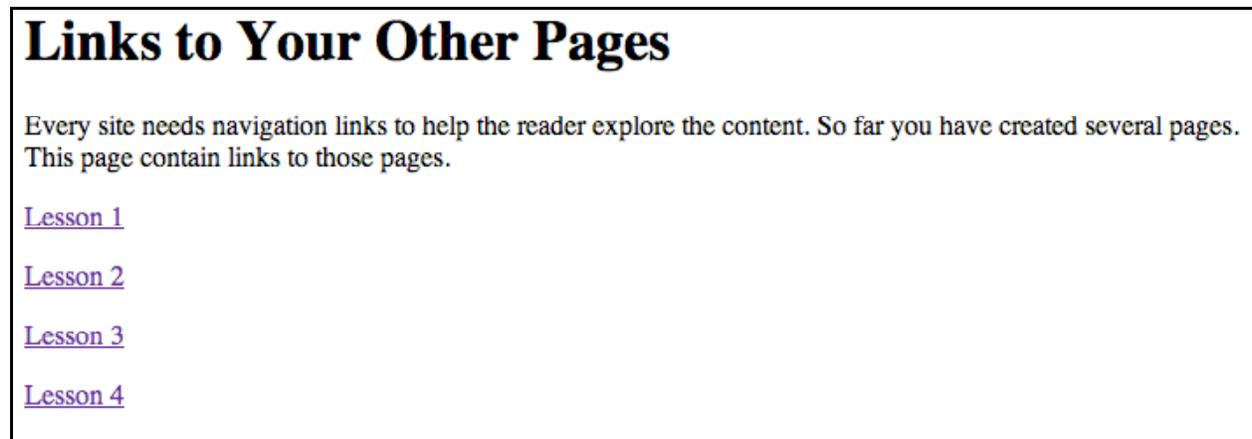
Activity 3.6: Links to Other Pages

ISTs will create the fifth of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Links to Other Pages” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson05.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Links to other pages</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Links to Your Other Pages</h1>
<p>Every site needs navigation links to help the reader explore the content.
So far you have created several pages. This page contain links to those
pages.</p>
<p><a href="lesson01.htm">Lesson 1</a></p>
<p><a href="lesson02.htm">Lesson 2</a></p>
<p><a href="lesson03.htm">Lesson 3</a></p>
<p><a href="lesson04.htm">Lesson 4</a></p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.



Notes:

Activity 3.7: Links to Other Sites

ISTs will create the sixth of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Links to Other Sites” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson06.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Links to other sites</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Links to Other Sites</h1>
<p>Many Web sites contain a page with links to related content found on other
Web sites. When creating links to other sites, you have many options
regarding the window in which the new site will load.</p>
<p>Here are some example links to other sites.</p>
<p><a href="http://www.stevenestrella.com/composers" target="_top">Dr.
Estrella's Incredibly Abridged Dictionary of Composers</a><br />
(Replaces this window)<br />
If you click this link, the page you are reading will be replaced by the
Dictionary of Composers. When you see the new page, click the Back button to
return to this page.</p>
<p><a href="http://www.menc.org" target="_blank">MENC</a><br />
(Creates a new window)<br />
If you click this link, a new window will appear containing the Web site for
MENC. You may drag the new window anywhere on the screen and close
it when you are finished. This page will remain on your screen.</p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Links to Other Sites

Many Web sites contain a page with links to related content found on other Web sites. When creating links to other sites, you have many options regarding the window in which the new site will load.

Here are some example links to other sites.

[Dr. Estrella's Incredibly Abridged Dictionary of Composers](#)

(Replaces this window)

If you click this link, the page you are reading will be replaced by the Dictionary of Composers. When you see the new page, click the Back button to return to this page.

[MENC](#)

(Creates a new window)

If you click this link, a new window will appear containing the Web site for MENC. You may drag the new window anywhere on the screen and close it when you are finished. This page will remain on your screen.

Notes:

Activity 3.8: Physical and Logical Styles

ISTs will create the seventh of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Physical and Logical Styles” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson07.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Physical and Logical Styles</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Physical and Logical Styles</h1>
<p> XHTML exists to structure content. CSS exists to add style components.
Physical and Logical style tags within XHTML do a bit of both. These very
basic styles can assist in structuring content. For example, you generally
use <i>Italics</i> or <em>Emphasis</em> for bibliographical entries.
<b>Bold</b> or <strong>Strong</strong> could be used for glossary terms. Web
browsers display content surrounded by these tags differently from
surrounding text but these tags also can add structure to a document when
used consistently. Physical styles are rendered by Web browsers to match
physical print styles. Logical styles are more open to variation on different
Web browsers, PDAs, cell phones, and other devices that browse the Web. </p>
<h2>Physical Styles</h2>
<p><b>Bold</b><br />
<i>Italic<br />
</i><tt>Monospaced typewriter font<br />
</tt><u>Underline</u> - deprecated, not recommended<br />
<s>Strike through</s> - deprecated, not recommended<br />
Text that displays below the line is called <sub>Subscript.<br /></sub>
Text that displays above the line is called <sup>Superscript.</sup></p>
<h2>Logical Styles</h2>
<p><strong>Strong</strong> - usually looks like bold but could be rendered as
ALL CAPS or in a different typeface. <br />
<em>Emphasis</em> - usually looks like italics but could be rendered as
<u>Underlined</u> or in a different typeface.</p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Physical and Logical Styles

XHTML exists to structure content. CSS exists to add style components. Physical and Logical style tags within XHTML do a bit of both. These very basic styles can assist in structuring content. For example, you generally use *Italics* or *Emphasis* for bibliographical entries. **Bold** or **Strong** could be used for glossary terms. Web browsers display content surrounded by these tags differently from surrounding text but these tags also can add structure to a document when used consistently. Physical styles are rendered by Web browsers to match physical print styles. Logical styles are more open to variation on different Web browsers, PDAs, cell phones, and other devices that browse the Web.

Physical Styles

Bold

Italic

Monospaced typewriter font

Underline - deprecated, not recommended

~~Strike through~~ - deprecated, not recommended

Text that displays below the line is called Subscript.

Text that displays above the line is called ^{Superscript}.

Logical Styles

Strong - usually looks like bold but could be rendered as ALL CAPS or in a different typeface.

Emphasis - usually looks like italics but could be rendered as Underlined or in a different typeface.

Notes:

Activity 3.9: Preformatted Text

ISTs will create the eighth of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Preformatted Text” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson08.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Preformatted Text</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Preformatted Text</h1>
```

```
<p>Sometimes it is convenient and quick to create text content in a simple
text editor and use the space bar to line up tables. Here is a simple table
using preformatted text.</p>
```

```
<pre>
Student Name    Quiz 1    Quiz 2    Quiz 3    Total    Average
Wolfgang        90        80        70        240      80
Ludwig          75        95        90        260      87
Richard         60        80        70        210      70
</pre>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Preformatted Text

Sometimes it is convenient and quick to create text content in a simple text editor and use the space bar to line up tables. Here is a simple table using preformatted text.

Student Name	Quiz 1	Quiz 2	Quiz 3	Total	Average
Wolfgang	90	80	70	240	80
Ludwig	75	95	90	260	87
Richard	60	80	70	210	70

Notes:

Activity 3.10: Tables

ISTs will create the ninth of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Table Tag” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson09.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Tables</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Tables</h1>
<p>Here is a table created using the table tag.</p>
<table border="1" cellspacing="1" cellpadding="1" summary="student scores">
<tr>
<th>Student Name</th>
<th>Quiz 1</th><th>Quiz 2</th><th>Quiz 3</th><th>Total</th>
<th>Average</th>
</tr>
<tr>
<td align="left">Wolfgang</td>
<td align="right">90</td><td align="right">80</td>
<td align="right">70</td><td align="right">240</td>
<td align="right">80</td>
</tr>
<tr>
<td align="left">Ludwig</td>
<td align="right">75</td><td align="right">95</td>
<td align="right">90</td><td align="right">260</td>
<td align="right">87</td>
</tr>
<tr>
<td align="left">Richard</td>
<td align="right">60</td><td align="right">80</td>
<td align="right">70</td><td align="right">210</td>
<td align="right">70</td>
</tr>
</table>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Tables

The table tag provides many more options for the display of tabular data. Tables are also sometimes used in page layout. Here is a simple table created using the table tag.

Student Name	Quiz 1	Quiz 2	Quiz 3	Total	Average
Wolfgang	90	80	70	240	80
Ludwig	75	95	90	260	87
Richard	60	80	70	210	70

Notes:

Activity 3.11: Horizontal Rules

ISTs will create the tenth of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Horizontal Rules” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson10.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Horizontal Rules</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Horizontal Rules</h1>
<p>Here is a simple horizontal rule with no special formatting.</p>
<hr />
<p>Here is a left aligned horizontal rule that occupies 50% of the page.</p>
<hr align="left" width="50%" />
<p>Here are a group of horizontal rules at 450, 400, 350, 300, 250, 200, 150,
100, and 50 pixels each. These are centered.</p>
<hr align="center" width="450" size="9" />
<hr align="center" width="400" size="8" />
<hr align="center" width="350" size="7" />
<hr align="center" width="300" size="6" />
<hr align="center" width="250" size="5" />
<hr align="center" width="200" size="4" />
<hr align="center" width="150" size="3" />
<hr align="center" width="100" size="2" />
<hr align="center" width="50" size="1" />
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Horizontal Rules

Here is a simple horizontal rule with no special formatting.



Here is a left aligned horizontal rule that occupies 50% of the page.



Here are a group of horizontal rules at 450, 400, 350, 300, 250, 200, 150, 100, and 50 pixels each. These are centered.



Notes:

Activity 3.12: Special Characters

ISTs will create the eleventh of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Special Characters” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson11.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Special Characters</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Special Characters</h1>
<p>Some common special characters are:</p>
<p>The copyright symbol: &copy;.</p>
<p>Umlauts on vowels: &auml; &euml; &iuml; &ouml; &uuml;</p>
<p>Acute accents on vowels: &aacute; &eacute; &iacute; &oacute; &uacute;</p>
<p>Grave accents on vowels: &agrave; &egrave; &igrave; &ograve; &ugrave;</p>
<p>Registered trademark symbol: &reg;</p>
<p>In context you might use special characters such as those in Kod&aacute;ly
and Sch&uuml;tz. </p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Special Characters

Some common special characters are:

The copyright symbol: ©.

Umlauts on vowels: ä ë ÿ ö ü

Acute accents on vowels: á é í ó ú

Grave accents on vowels: à è ì ò ù

Registered trademark symbol: ®

In context you might use special characters such as those in Kodály and Schütz.

Notes:

Activity 3.13: Images

ISTs will create the twelfth of 12 XHTML documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Special Characters” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lesson12.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Placing Images</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Placing Images</h1>
<p>Here is the TI:ME Logo.</p>
<p>

</p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.



Notes:

Activity 3.14: Framesets

ISTs will create a frameset and home page document to organize and navigate the documents listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will create a basic two-frame frameset using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “lessonframes.htm” in the mysite folder created in Activity 3.1. The instructor will explain the target attribute and its particular importance in frames-based Web site designs.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>My XHTML Lessons</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
  <frameset cols="200,*">
    <frame src="lessonnav.htm" name="leftFrame" scrolling="No"
noresize="noresize" id="leftFrame" />
    <frame src="lessonhome.htm" name="mainFrame" id="mainFrame" />

  <noframes>
    <body>
      <p>This site requires a frames-compatible browser.</p>
    </body>
  </noframes>
</frameset>
</html>
```

The lessonframes.htm document loads lessonnav.htm in the left frame and lessonhome.htm in the main frame on the right. The code for those two pages follows.

Here is the code for lessonnav.htm. ISTs will create this page in Design View and save it as “lessonnav.htm” in the mysite folder.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Lesson Navigation</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<b>XHTML Examples</b>
<ol>
<li><a href="lesson01.htm" target="mainFrame">A Basic Page</a></li>
<li><a href="lesson02.htm" target="mainFrame">Unordered Lists</a></li>
<li><a href="lesson03.htm" target="mainFrame">Ordered Lists</a></li>
<li><a href="lesson04.htm" target="mainFrame"> Links within a page</a></li>
<li><a href="lesson05.htm" target="mainFrame">Links to Other pages</a></li>
<li><a href="lesson06.htm" target="mainFrame">Links to Other Sites</a></li>
<li><a href="lesson07.htm" target="mainFrame">Physical vs. Logical</a></li>
<li><a href="lesson08.htm" target="mainFrame">Preformatted Text</a></li>
<li><a href="lesson09.htm" target="mainFrame">Table Tag</a></li>
<li><a href="lesson10.htm" target="mainFrame">Horizontal Rules</a></li>
<li><a href="lesson11.htm" target="mainFrame">Special Characters</a></li>
<li><a href="lesson12.htm" target="mainFrame">Images</a></li>
</ol>
<p><b>CSS Examples</b></p>
<ol>
<li><a href="cssinline.htm" target="mainFrame">Inline Styles</a></li>
<li><a href="cssinternal.htm" target="mainFrame">Internal Style
Sheet</a></li>
<li><a href="cssexternal.htm" target="mainFrame">External Style
Sheet</a></li>
<li><a href="basicstyle.css" target="mainFrame">A Basic Style Sheet</a>
</li>
</ol>
<p><a href="lessonhome.htm" target="mainFrame">Home</a></p>
</body>
</html>
```

ISTs will create the CSS examples in the next set of activities.

Here is the code for lessonhome.htm. ISTs will create this page in Design View and save it as “lessonhome.htm” in the mysite folder.

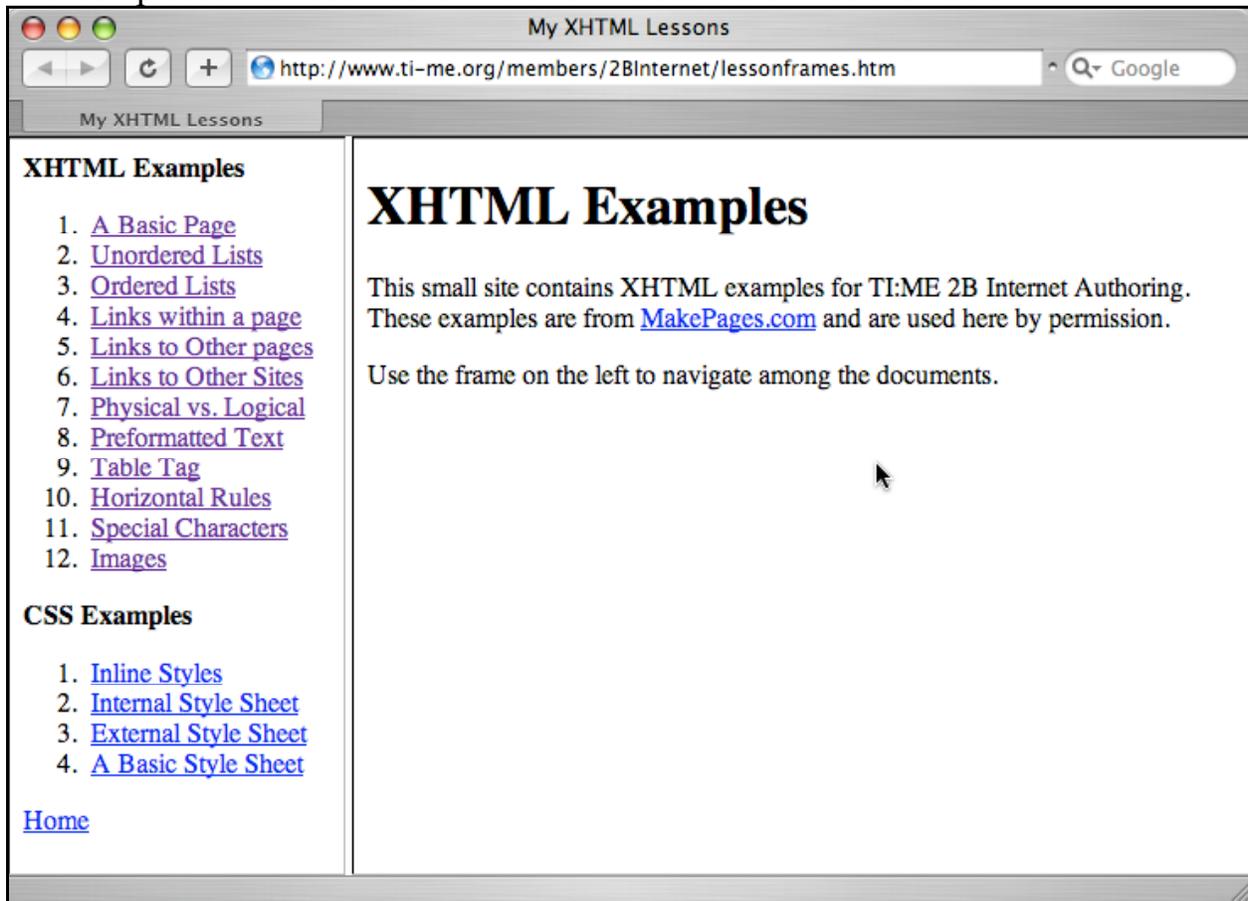
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>XHTML Lessons Home Page</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>XHTML Examples</h1>

<p>This small site contains XHTML examples for TI:ME 2B Internet Authoring.
These examples are from <a href="http://www.makepages.com"
target="_blank">MakePages.com</a> and are used here by permission.</p>
<p>Use the frame on the left to navigate among the documents.</p>

</body>
</html>
```

ISTs will complete this topic by launching a Web browser and opening lessonframes.htm.

The completed frameset looks like this in a standard Web browser.



Notes:

Topic 4: Cascading Style Sheets (CSS)

The instructor will explain the concepts behind CSS including style rules and the placement of style sheets. ISTs will then begin a series of activities to create XHTML pages that incorporate CSS and view them locally. The instructor will explain concepts as they are introduced in each lesson. Instructors and ISTs should read “CSS Basics” in the appendices.

Notes:

Activity 4.1: Inline Styles

ISTs will create the first of four CSS examples listed as XHTML and CSS

Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Inline Styles” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “cssinline.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Inline Styles</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Welcome to Sidebars Anonymous</h1>
<p>This paragraph uses the default white background of the page.</p>
<div style="background : silver;"><p>Here is a block of text that will appear
with a silver background in the browser, regardless of the background color
of the page.</p></div>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Welcome to Sidebars Anonymous

This paragraph uses the default white background of the page.

Here is a block of text that will appear with a silver background in the browser, regardless of the background color of the page.

Notes:

Activity 4.2: Internal Style Sheets

ISTs will create the second of four CSS examples listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Internal Style Sheet” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “cssinternal.htm” in the mysite folder created in Activity 3.1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Internal Style Sheets</title>
<style type="text/css">
body { font-family: serif; color: black; background: white;}
h1 { font-family: sans-serif; color: maroon; }
</style>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<body>
<h1>Welcome to Internal Style Sheets</h1>
<p>The heading above is displayed in a sans-serif font in maroon. This text,
however, is shown in black in a serif font.</p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.



Notes:

Activity 4.3: External Style Sheets

ISTs will create the third of four CSS examples listed as XHTML and CSS

Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “Internal Style Sheet” using the Design View feature of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below. ISTs will then save the page with the name “cssexternal.htm” in the mysite folder created in Activity 3.1.

The page, cssexternal.htm, links to an external style sheet called basicstyle.css. ISTs will create this style sheet in activity 4.4.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Enjoying Style Sheets</title>
<link rel="stylesheet" type="text/css" href="basicstyle.css" />
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1"
/></head>
<body>
<h1>Level 1 headings in color and sans serif</h1>
<h2>Level 2 headings in color and sans serif</h2>
<h3>Level 3 headings in color and sans serif</h3>
<h4>Level 4 headings in color and sans serif</h4>
<h5>Level 5 headings in color and sans serif</h5>
<h6>Level 6 headings in color and sans serif</h6>
<p>This is a plain paragraph with wrapping text. It should be in a serif
typeface. <b>This is a bold sentence.</b></p>
<p class="white">White text paragraph. This is a paragraph with wrapping
text shown in white. It should be in a serif typeface.</p>
<p class="indented">Indented paragraph. This is a paragraph with wrapping
text and the first line indented. It should be in a serif typeface. This is a
paragraph with wrapping text and the first line indented. It should be in a
serif typeface.</p>
<p class="dropcap">Dropcap paragraph. This is a paragraph with wrapping
text and dropcap on the first letter. It should be in a serif typeface. This
is a paragraph with wrapping text and dropcap on the first letter. It should
be in a serif typeface. This is a paragraph with wrapping
text and dropcap on the first letter. It should be in a serif typeface. This
is a paragraph with wrapping text and dropcap on the first letter. It should
be in a serif typeface.</p>
<p class="box">Boxed paragraph. This is a boxed paragraph with wrapping text.
It should be in a serif typeface. This is a boxed paragraph with wrapping
text.</p>
```

```
<p>The paragraph below is hidden.</p>
<p class="hidden">Invisible paragraph. This is an invisible paragraph with
wrapping text. It should be in a serif typeface. This is a plain paragraph
with wrapping text. This is an invisible paragraph with wrapping text. It
should be in a serif typeface.</p>
<p class="monosans">monospaced sans serif text here</p>
<h2>LINKS</h2>
<p>Roll your mouse over these links to see the style sheet effects.</p>
<p><a href="http://www.google.com">Google</a></p>
<p><a href="http://www.menc.org">MENC</a></p>
<p>A Horizontal Rule will appear below. The color property for this element
is not supported in all browsers but the width property is widely
supported.</p>
<hr />
<p class="legal">Please note: The construction of this page caused many font
tags to be left homeless and unemployed.</p>
</body>
</html>
```

The completed page looks like this in a standard Web browser.

Level 1 headings in color and sans serif

Level 2 headings in color and sans serif

Level 3 headings in color and sans serif

Level 4 headings in color and sans serif

Level 5 headings in color and sans serif

Level 6 headings in color and sans serif

This is a plain paragraph with wrapping text. It should be in a serif typeface. **This is a bold sentence.**

White text paragraph. This is a paragraph with wrapping text shown in white. It should be in a serif typeface.

Indented paragraph. This is a paragraph with wrapping text and the first line indented. It should be in a serif typeface. This is a paragraph with wrapping text and the first line indented. It should be in a serif typeface.

Dropcap paragraph. This is a paragraph with wrapping text and dropcap on the first letter. It should be in a serif typeface. This is a paragraph with wrapping text and dropcap on the first letter. It should be in a serif typeface. This is a paragraph with wrapping text and dropcap on the first letter. It should be in a serif typeface. This is a paragraph with wrapping text and dropcap on the first letter. It should be in a serif typeface.

Boxed paragraph. This is a boxed paragraph with wrapping text. It should be in a serif typeface. This is a boxed paragraph with wrapping text.

The paragraph below is hidden.

monospaced sans serif text here

LINKS

Roll your mouse over these links to see the style sheet effects.

Google

MENC

A Horizontal Rule will appear below. The color property for this element is not supported in all browsers but the width property is widely supported.

Please note: The construction of this page caused many font tags to be left homeless and unemployed.

Notes:

Activity 4.4: Creating an External Style Sheet

ISTs will create the fourth of four CSS examples listed as XHTML and CSS Examples at: <http://www.ti-me.org/members/2BInternet/>

ISTs will recreate the document labeled “A Basic Style Sheet” using the CSS editing tools of the Internet Authoring software. They will then review the code they created using the Code View feature. The code should be similar to the example below but the indentations and line wraps produced by Internet Authoring software packages will vary. ISTs will then save the page with the name “basicstyle.css” in the mysite folder created in Activity 3.1.

```
body {
    font-family: Georgia, "Times New Roman", Times, serif;
    color: #000000;
    background-color:#CCCCCC;
    margin: 1em;
    width: 600px;
}
.box {
    padding: 1em;
    border-top-width: thin;
    border-right-width: thin;
    border-bottom-width: thin;
    border-left-width: thin;
    border-top-style: solid;
    border-right-style: solid;
    border-bottom-style: solid;
    border-left-style: solid;
}
.dropcap:first-letter {
    font-size: 2em;
    font-weight: bolder;
    color:#006666;
    float: left;
}
h1 {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 130%;
    color: #006666;
}
h2 {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 120%;
    color: #006666;
}
h3 {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 110%;
    color: #006666;
}
```

```
h4 {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 105%;
    color: #006666;
}
h5 {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 100%;
    color: #006666;
}
h6 {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 95%;
    color: #006666;
}
.white {color: #FFFFFF;}
.hidden {visibility: hidden;}
.indented {text-indent: 20pt;}
.legal {font-size: 70%; font-style: italic;}
.monosans {font-family: monaco, fixedsys, monospace;}
b {font-weight: bold; color: #006666;}
hr {color: #006666; width: 600px;}
a:link {color: #990000; text-decoration: underline;}
a:visited {color: #666666; text-decoration: underline; }
a:hover {color: #FFFFFF; text-decoration: none; background-color: #990000;}
a:active {color: #000000;text-decoration: none; background-color: #990000;}
```

Notes:

ACTIVITY 4.5: CONFLICTING RULES

(Content from makepages.com. Used here by permission.)

A Web page may link to an external style sheet, use an internal style, and also use inline styles. When multiple style rules apply to the same content, the Web browser uses a set of rules to resolve the conflict based on a cascade order. The rules are complex but are based on easily-understood principles of specificity and proximity. The more specifically a rule applies to an element, the more weight it is given in resolving any conflicts. An id selector, for example, is more specific than a class selector, which in turn is more specific than an element selector. Proximity matters as well. An inline style will override an internal style, which in turn will override a rule found in an external style sheet. To force a given rule to take precedence you may use the !important value as part of the declaration.

```
h1 { color: maroon !important;}
```

ISTs will create one or more XHTML documents that use an internal style sheet as well as inline styles that override rules in the internal style sheet.

Notes:

ACTIVITY 4.6: ELEMENT CLASSIFICATION

(Content from makepages.com. Used here by permission.)

CSS recognizes three types of elements. **Block-level** elements include paragraphs, heading, lists, tables, and divs. Block-level elements always begin on a new line. **Inline** elements include anchors (`<a>`), spans, form elements, and images. Inline elements exist in the natural flow of the document without forced line breaks. Inline elements may contain other inline elements but may not contain block-level elements. **List-item** elements are the `` elements found in unordered and ordered lists. List-item elements have markers associated with them such as bullets, numbers, or letters.

ISTs will create one or more XHTML documents that include block-level, inline, and list-item elements.

Notes:

ACTIVITY 4.7: ELEMENT BOX

(Content from makepages.com. Used here by permission.)

An invisible element box surrounds every element in CSS. Each element box has an outer edge, margin, border, padding, and inner edge that contain the width and height of the content.

ISTs will create one or more XHTML documents that demonstrate different ways to alter the element box of block-level and inline-level elements.

Notes:

Topic 5: Linking to Media Files

The instructor will explain how the `<a>` tag can be used to link to a variety of different types of media. The instructor will draw a distinction between the types of files that can be displayed by a typical Web browser (html, txt, gif, jpg, pdf, png, swf, mov, wmv, rm) and those that are typically displayed by separate applications (doc, xls, ppt, tiff, eps). The instructor will show ISTs how to create simple links using Internet Authoring Software.

For more information and live examples for this topic, see the resource page at: <http://www.ti-me.org/members/2BInternet/linktomedial.htm>

Activity 5.1: Linking to Text Files

Plain text files continue to be useful resources because the absence of special formatting makes it very easy for visitors to copy and paste the text into their favorite word processor for editing. When a text file loads in a Web browser, the text wrapping and line breaks may vary from browser to browser. Text files created on DOS/Windows use one standard for line breaks. Text files created on Macintosh use another standard. A third standard is used for text files created on UNIX computers. If your text editor gives you a choice, create text files with DOS/Windows line breaks to ensure maximum compatibility. Don't worry, Macintosh and UNIX text readers automatically translate DOS/Windows line breaks, so there is rarely any problem.

Each IST will create an XHTML file that links to a plain text file using code modeled after the following:

```
<a href="anytextfile.txt" target="_blank">Link to a text file</a>
```

Notes:

Activity 5.2: Linking to Image Files

The first Web browsing software only recognized text. Later browsers were capable of displaying graphics in GIF (Graphic Interchange Format) and JPG (Joint Photographic Experts Group) formats. GIF format is limited to 256 colors so GIF is most useful for simple icons. The GIF format can also include multiple images in a single file. Such files are called animated GIFs. JPG is a better format for photographs because it allows for millions of colors in an image. Today's browsers also display PNG (Portable Network Graphics), a newer format that allows for transparency as well as photorealistic color. Links to images in these file formats allow visitors to view images in a separate window or to download the images to their hard drive.

There are other file formats used in print production such as TIFF, EPS, and PDF. You can link to these file formats but when the visitor clicks the link, the linked file will not be displayed in the browser. Rather it will be downloaded to the visitor's hard drive. In the case of PDF files, however, a special Web browser plugin is available from Adobe to allow PDF files to be viewed within a Web browser window.

Each IST will create an XHTML file that links to one or more image files using code modeled after the following:

```
<a href="anygraphicfile.gif" target="_blank">Link to image file</a>
```

Notes:

Activity 5.3: Linking to Media Files that use Browser Plugins

Web browsers require additional software to play audio, video, Flash animation, and other file formats. The other software may take the form of a Web browser plugin or ActiveX control that allows a particular media type to be displayed within a Web browser window. For example, Flash animations play easily in almost every Web browser because the plugin software for Flash is usually distributed and installed with the browser. When a new version of the Flash player plugin is released, however, the visitor must often download and install the new version to continue to have access to the latest Flash media. QuickTime movies, Windows Media Player files, RealPlayer files, and other media formats each have their own plugin software.

Each IST will create an XHTML file that links to one or more media files using code modeled after the following:

```
<a href="anymediafile.swf" target="_blank">Link to media file</a>
```

Notes:

Activity 5.4: Linking to Media Files that use Separate Applications

Many software products such as Microsoft Office, Adobe Illustrator, Finale NotePad, and others create documents that can't be opened by a Web browser. You can still make these documents available for download through a simple link but the visitor will have to open the downloaded document using a separate application. For example, you can link to a Finale NotePad file. When the visitor clicks the link, the Finale NotePad file is downloaded to the visitor's hard drive. The visitor can then launch Finale NotePad and open the file. Notice the instructions to the visitor on how to download the files. Also, notice that the code does not use the `target="_blank"` attribute because the linked file will not be opening in a separate web browser window. Web browsers and platforms are inconsistent in how they handle special file formats. For best results, create zip archives of documents you wish to make available for download.

Please Note: Some Web browsers can be configured to launch separate “helper” applications when selected file types are downloaded. Other browsers limit this behavior to certain “safe” file types such as pdf files. To learn more about this feature, view the preferences menu item for the browser you are using.

Each IST will create an XHTML file that links to one or more media files using code modeled after the following:

```
<a href="otherfile.zip" target="_blank">Zip archive of files</a>
```

Notes:

Topic 6: Embedding Media Files

The instructor will show ISTs how to embed common media file formats using Internet Authoring Software. In some cases, media embedding code must be typed. This is the case with QuickTime content. ShearSpire, Inc. has developed a QuickTime Code Generator and has licensed it to TI:ME for use in this course. You will find a copy of the QuickTime Code Generator on the resource pages for this workbook and you can use it to create the QuickTime embedding code you need for your projects.

For more information and live examples for this topic, see the resource page at: <http://www.ti-me.org/members/2BInternet/embedmedia.htm>

Activity 6.1: Embedding Images

GIF, JPG, and PNG image formats may be embedded in Web pages. The width and height attributes should be included in the code to make it easier for the Web browser to allocate space correctly as the page loads. The alt attribute should be included for compliance with XHTML specifications and accessibility for those with vision problems or those who browse with image loading turned off in their browser.

- GIF files are limited to 256 colors so GIF is best for simple images. GIF files can have a transparent background.
- JPG files may contain millions of colors. JPG is the most common file format for photographs on the Web. Compare the same photo saved in GIF and JPG formats. The JPG image has smoother gradients and a more realistic appearance.
- PNG is a more recent file format that was created to replace GIF. Like GIF, PNG files may have a transparent background. PNG files, however, may contain millions of colors. That makes PNG a nice alternative to GIF for displaying higher resolution images. Newer Web browsers support PNG.
- It is customary to store images in a separate folder. The code to embed an image will therefore contain a pathname that includes the folder name.

Each IST will create an XHTML file that embeds images using code modeled after the following:

```

```

Notes:

Activity 6.2: Embedding Flash Media

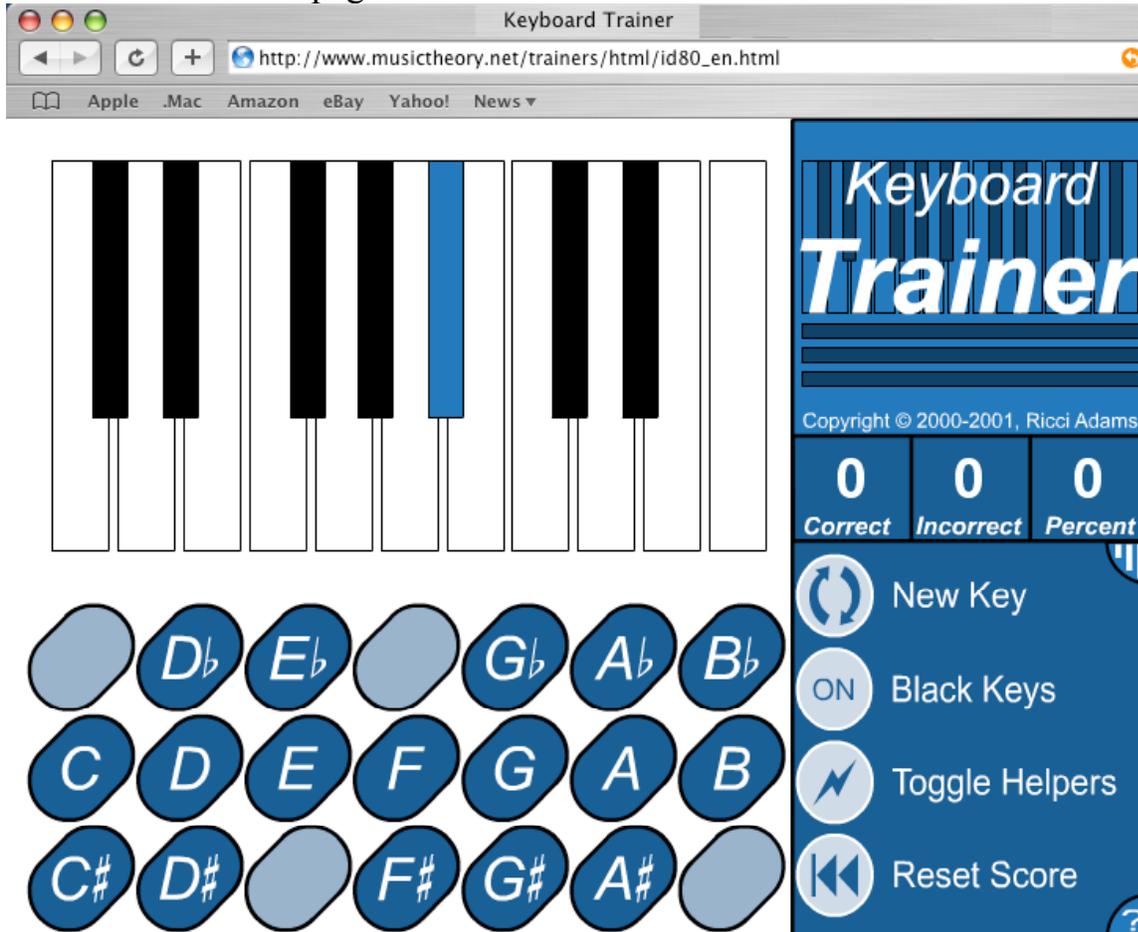
Web browsers require additional software to play audio, video, Flash animation, and other file formats. The other software may take the form of a Web browser plugin or ActiveX control that allows a particular media type to be displayed within a Web browser window. For example, Flash animations play easily in almost every Web browser because the plugin software for Flash is usually distributed and installed with the browser. When a new version of the Flash player plugin is released, however, the visitor must often download and install the new version to continue to have access to the latest Flash media. QuickTime movies, Windows Media Player files, RealPlayer files, and other media formats each have their own plugin software. Plugin software often gives the developer the option to embed the media into an existing XHTML page rather than have the media file open in a separate window.

The code to embed audio, video, and animation media is different for each media type. The most common type of media to embed is Flash (SWF). Internet Authoring software packages make it easy to embed Flash media without the need for coding by hand. The code, as you can see, is lengthy so it is best to have the Internet Authoring software generate the code whenever possible. If you must type the code yourself, however, be aware that both the <object> and <embed> tags are needed for compatibility with all major browsers on both Macintosh and Windows. Notice how the <embed> tag is nested within the <object> tag in the examples below. If you look at the code carefully, you can see that parameters such as width and height are included in both the <object> and <embed> tags. This redundancy is necessary because browsers that support the <object> tag ignore the <embed> tag and browsers that support the <embed> tag ignore the <object> tag.

Each IST will create an XHTML file that embeds Flash media using code modeled after the following:

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab
#version=7,0,19,0" width="700" height="400" title="Joplin Analysis">
<param name="movie" value="flash/joplin.swf" />
<param name="quality" value="high" />
<embed src="flash/joplin.swf" quality="high"
pluginspage="http://www.macromedia.com/go/getflashplayer"
type="application/x-shockwave-flash" width="700" height="400"></embed>
</object>
```

Here is a screen shot of a music education program created using Flash and embedded in a Web page.



Notes:

Activity 6.3: Embedding Video with QuickTime

The QuickTime plugin can handle a variety of audio and video file formats. The most typical ones being .mov, .avi, .mp3, and .mid. The code to embed QuickTime media is complex but it can be easily copied from online examples and modified as needed. One note about dimensions: add 20 to the height dimension of the movie to allow for the controller. In the first code example below, the actual height of the movie is 120 pixels. The code, however, uses 140 as the height value to allow for the controller bar that appears on the bottom of every QuickTime movie.

Each IST will create an XHTML file that uses QuickTime to embed a video file with code modeled after the following:

```
<object classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" width="160"
height="140" codebase="http://www.apple.com/qtactivex/qtplugin.cab">
<param name="src" value="videos/fingering1.mov" />
<param name="autoplay" value="false" />
<param name="controller" value="true" />
<param name="loop" value="false" />
<embed src="videos/fingering1.mov" width="160" height="140"
autoplay="false" controller="true" loop="False"
pluginspage="http://www.apple.com/quicktime/download/">
</embed>
</object>
```

Notes:

Activity 6.4: Embedding Audio with QuickTime

Audio files such as .mid, .wav, and .mp3 can be embedded using QuickTime. The code is the same as for embedding video except that the height dimension is always 20. You can choose any width you like for the controller but using a value of 200 or more will allow the visitor to easily track the progress of the playback.

Each IST will create an XHTML file that uses QuickTime to embed an audio file with code modeled after the following:

```
<object classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" width="200"
height="20" codebase="http://www.apple.com/qtactivex/qtplugin.cab">
<param name="src" value="sounds/bach.mid" />
<param name="autoplay" value="false" />
<param name="controller" value="true" />
<param name="loop" value="false" />
<embed src="sounds/bach.mid" width="200" height="20" autoplay="false"
controller="true" loop="false"
pluginspage="http://www.apple.com/quicktime/download/">
</embed>
</object>
```

Notes:

Activity 6.5: Embedding Media with the QuickTime Poster Technique

Embedding a large media file on a Web page is a bit of an inconvenience for visitors with slow Internet connections. To avoid subjecting unsuspecting visitors to long download times, you can use the QuickTime poster movie technique.

The QuickTime plugin allows the designer to embed a small section of a movie (usually one frame), called a poster movie, and link it to the original, larger movie. The target and href attributes in the embed tag are set to load the new movie in the same space on the screen as the poster movie. In the code below, note that the autoplay attribute has been removed because it is irrelevant for a still image. Also note that the HREF URL is relative to the location of the movie not the location of the page. Since the poster frame is displayed without a controller, it is necessary to specify a white background color to mask the extra 20 pixels allocated in the code for the controller needed for the full movie. The audio example uses a small QuickTime movie called audiostream.mov that contains a still image of a playback controller.

Each IST will create an XHTML file that uses QuickTime to embed an audio file with code modeled after the following:

```
<object classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" width="200"
height="40" codebase="http://www.apple.com/qtactivex/qtplugin.cab">
<param name="src" value="sounds/bach.mid" />
<param name="autoplay" value="false" />
<param name="controller" value="true" />
<param name="loop" value="false" />
<embed src="sounds/bach.mid" width="200" height="40" autoplay="false"
controller="true" loop="false"
pluginspage="http://www.apple.com/quicktime/download/">
</embed>
</object>
```

ShearSpire, Inc. has developed a QuickTime Code Generator and has licensed it to TI:ME for use in this course. You will find a copy of the QuickTime Code Generator on the resource pages for this workbook and you can use it to create the QuickTime embedding code you need for your projects.

Notes:

Activity 6.6: Embedding Notation with the Scorch Plugin

Sibelius introduced the Scorch plugin in the late 1990s to allow musicians to share their scores easily online. Many sites, such as the Vermont MIDI Project, use the Sibelius Scorch plugin to help students share their work with mentors. Creating Scorch files is easy. Create a notation file using Sibelius, then choose Save As from the file menu. Save the file as a Web page. Then upload the new .htm file along with the original .sib (Sibelius) file to your Web server. The resource page contains an example of a Sibelius Scorch page created this way. If you code it yourself, however, you can embed a Sibelius file in any page using the code below.

```
<object id="ScorchPlugin"
classid="clsid:A8F2B9BD-A6A0-486A-9744-18920D898429"
width="604"
height="823"
codebase="http://www.sibelius.com/download/software/win/ActiveXPlugin.cab#3,0,0,4">
<param name="src" value="scorch/AllthroughtheNight.sib" />
<param name="type" value="application/x-sibelius-score" />
<param name="scorch_minimum_version" value="3000" />
<param name="scorch_preferred_version" value="3000" />
<param name="scorch_shrink_limit" value="90" />
<embed src="scorch/AllthroughtheNight.sib"
scorch_minimum_version="3000"
scorch_preferred_version="3000"
scorch_shrink_limit="90"
width="604"
height="823"
type="application/x-sibelius-score"
pluginspage="http://www.sibelius.com/cgi/plugin.pl">
</embed>
</object>
```

```
<p>If you can't see the score, <a href="http://www.sibelius.com/scorch/">get
the Sibelius Scorch
plug-in here</a></p>
```

The completed page looks like this in a standard Web browser.

All Through the Night

Wales and England

F B^b G⁷ C B^b C⁷

Sleep, my child and peace at - tend thee, All through the
While the moon her watch is keep - ing All through the

F B^b G⁷ C

night; Guard - ian an - gels God will send thee,
night; While the wea - ry world is sleep - ing

B^b C⁷ F C⁷

All through the night; Soft the drows - y
All through the night; O'er thy spi - rit

hours are creep - ing, Hill and vale in slum - ber steep - ing,
gent - ly steal - ing, Vis - ions of de - light re - veal - ing,

F B^b G⁷ C B^b C⁷ F

I my lov - ing vig - il keep - ing, All through the night.
Breathes a pure and ho - ly feel - ing, All through the night.

If you can't see the score, [get the Sibelius Scorch plug-in here](#).

Created using [Sibelius](#).

Visit the largest collection of new scores on the web at www.sibeliusmusic.com.

Notes:

Topic 7: Uploading to the Web

To make files available for public viewing on the Web, you must first upload (i.e. copy) those files to a Web server. Establishing an account with a Web hosting provider may be beyond the scope (and budget) of students in this workshop. Free commercial hosting space (angelfire.com, geocities.com, etc.) is not a good alternative, however, because the hosting service adds advertising content to every page you upload. Often institutions offering TI:ME 2B Internet will have hosting space available for students. If that is not the case, TI:ME has some temporary space available for hosting of student files. For details, see the resource page at: <http://www.ti-me.org/members/2BInternet/>

Uploading to a Web server requires the following information:

Host Name: (example <ftp.ti-me.org>)

Login Name: (example 2bstudent@ti-me.org)

Password: (can be anything)

Path or Directory: (example /student01/)

Activity 7.1: FTP

Each IST will use FTP software or the FTP feature of Internet Authoring software to upload files to assigned space on a Web server. Each IST will then type the appropriate URL into a Web browser to view the uploaded files.

Notes:

Topic 8: Dynamic Content

The instructor will show ISTs how to use packaged behaviors in Internet Authoring software to create rollover image swapping, popup messages, and jump menus.

For more information and live examples for this topic, see the resource page at:
<http://www.ti-me.org/members/2BInternet/>

Activity 8.1: Rollover Image Swapping with Packaged Behaviors

ISTs will create XHTML pages to demonstrate image swapping. ISTs will use the packaged behaviors available in Internet Authoring software.

Notes:

ACTIVITY 8.2: ROLLOVER IMAGE SWAPPING WITH CUSTOM CODE

ISTs will create XHTML pages to demonstrate image swapping. ISTs will copy and adapt model JavaScript code for this purpose found at:

<http://www.ti-me.org/members/2BInternet/>

Notes:

Activity 8.3: Popup Messages with Packaged Behaviors

ISTs will create XHTML pages to demonstrate popup messages for glossary items. ISTs will use the packaged behaviors available in Internet Authoring software. It is possible to have popup messages appear when a page loads, when an image is clicked, or when text is clicked. One possible educational application is to use popup messages to present definitions of highlighted terms.

Notes:

Activity 8.4: Jump Menus with Packaged Behaviors

One common use of form elements is to create a navigation menu also known as a jump menu. Internet Authoring software contains packaged behaviors to create jump menus like the one shown on the resource pages for this workbook. ISTs will create XHTML pages to demonstrate jump menus for navigation.

Notes:

Activity 8.5: Client-Server Technologies (PHP, ASP, JSP)

JavaScript is only one way to produce dynamic Web pages. ISTs will locate sites on the Internet that use PHP (Hypertext Preprocessor), ASP (Active Server Pages), and JSP (Java Server Pages).

EACH IST WILL CREATE AN XHTML PAGE CONTAINING THE FOLLOWING CODE:

```
<?php echo $_SERVER['HTTP_USER_AGENT']; ?>
```

THE PAGE WILL BE SAVED AS "browserdetect.php" AND WILL BE UPLOADED TO A SERVER THAT SUPPORTS PHP.

Notes:

Activity 8.6: Online Databases

Backend databases are very useful for online databases such as TI:ME's Lesson Plan database at ti-me.org. Often databases combine a backend technology like MySQL with a server side scripting language such as PHP. Each IST will locate a site (other than ti-me.org) that uses a backend database.

Notes:

Topic 9: Online Services

The instructor will introduce ISTs to the many online services that assist in creating Web-based content. These services include blogs, message boards, discussion groups, virtual classrooms, and sites that specifically cater to the needs of teachers.

Please Note: The instructor should introduce and explain each topic with examples on the Web but it may not be possible for each IST to complete each activity due to time constraints or technical issues. Each IST, however, should complete at least 3 of the activities listed in this topic.

For more information and links for this topic, see the resource page at:

<http://www.ti-me.org/members/2BInternet/>

Activity 9.1: Creating a Blog

Blogs are online journals, maintained by one or more individuals, who publish opinions and articles on the Web. Blogs are easy to publish and that allows all sorts of people to maintain lively websites on a variety of topics. A blog can be an important part of a music teacher's presence on the Web. Because blogs are easy to update, they are often a good choice for information that changes frequently such as news and announcements of upcoming events.

Each IST will create a blog using any of the free blogging services such as blogger.com. ISTs will use their blogs to post announcements and news related to their music programs. ISTs will create links to their blogs on standard XHTML pages.

For helpful links, see the resource page at:

<http://www.ti-me.org/members/2BInternet/>

Here is a screen shot of a typical blog.

The screenshot shows a Blogger blog interface. At the top, there is a blue navigation bar with the Blogger logo, a search box for 'SEARCH THIS BLOG', and buttons for 'GET YOUR OWN BLOG', 'FLAG?', and 'NEXT BLOG'. Below the navigation bar is an orange header with the title 'Web Class Blog' in white. The main content area is divided into two columns. The left column contains a 'Links' section with 'Google News', 'Edit-Me', and 'Edit-Me', and an 'archives' section with 'July 2005' and 'May 2006'. The right column features a post dated 'Tuesday, May 09, 2006' with the title 'Summer 2006 Web Development Classes'. The post includes a photo of a man in a suit and glasses, and a paragraph of text. At the bottom of the post, it says '# posted by Steven @ 12:42 PM 0 comments'.

Links
Google News
Edit-Me
Edit-Me

archives
July 2005
May 2006

Tuesday, May 09, 2006

Summer 2006 Web Development Classes



On day 1 of the class we often begin by installing the Dreamweaver and Fireworks software. I sometimes take pictures of all the students for use in their web projects. I can take them later in the week if you prefer. Here is a photo of me from my personal archives.

posted by Steven @ 12:42 PM 0 comments

Notes:

Activity 9.2: Creating a Feedback Form

XHTML provides the `<form></form>` tag to allow designers to create online forms to query visitors. One of the most common uses of forms is to request feedback from visitors to your site. Of course, a simple email link will also work but using a form is a better choice when you have specific questions to ask or if you are concerned about publishing your email address on the Web.

Each IST will create a simple feedback form using Internet Authoring software. Each IST will use an existing form mail CGI made available by the instructor or will obtain a form mail account from an online provider of form mail processing. Each IST will upload and test a page with a feedback form on their assigned Web space.

For helpful links, see the resource page at:
<http://www.ti-me.org/members/2BInternet/>

Here is a screen shot of a typical feedback form.

TI:ME 2B Internet Authoring Resource Pages

Feedback Form

Please let us know what you think of the TI:ME 2B Internet Resource Pages.

Required fields are marked with *

Name:*

E-mail:*

Phone:

Your message:*

Notes:

Activity 9.3: Creating a Counter

Counters display the number of times a page has been viewed. Counters are useful in gauging the popularity of different pages on your site. Many companies on the Web make ad-supported counters available for free. In many cases, you have control over which ads are shown and the ads are generally tasteful and unobtrusive.

Each IST will create an account with a company providing Web page counters. Each IST will install a simple counter on a page and upload it to their assigned Web space.

For helpful links, see the resource page at:
<http://www.ti-me.org/members/2BInternet/>

Here is a screen shot of a page with counters.

Sample Web Page Counters

Basic Ad-Free Counter from [iPowerWeb](#)
(our Web hosting service)



From [AmazingCounters.com](#)


Web Hosting Packages

From [Tech-Counter.com](#)

0000000028
[home treadmill](#)

Notes:

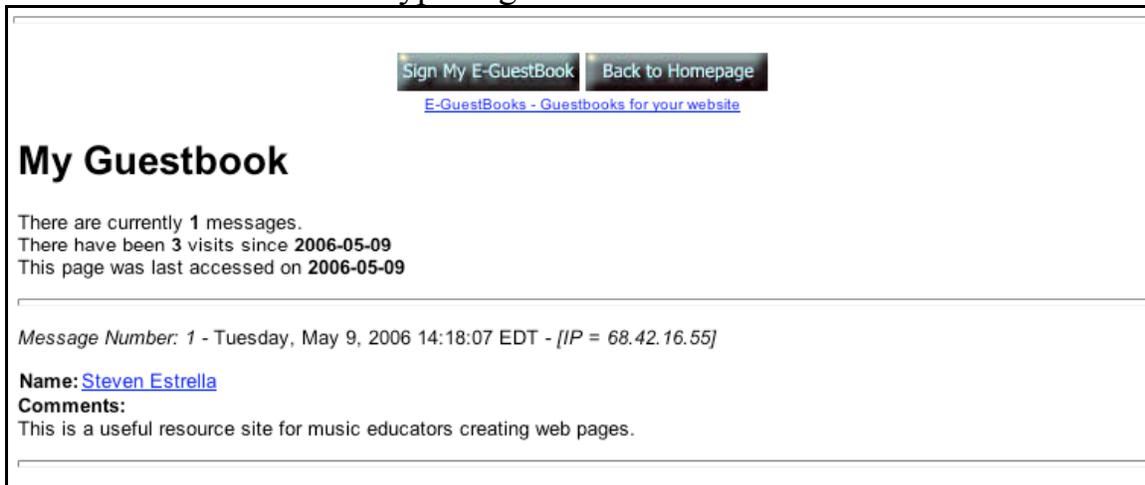
Activity 9.4: Creating a Guestbook

A guestbook on your site allows visitors to leave comments that are viewable by any of the other visitors. Many companies on the Web make ad-supported guestbooks available for free. In many cases, you have control over which ads are shown and the ads are generally tasteful and unobtrusive. Guestbook software often automatically deletes common obscene words but you should still monitor the content of your guestbook to make sure visitors are using it appropriately.

Each IST will create an account with a company providing Web page counters. Each IST will add a link to their guestbook on a standard XHTML page and upload it to their assigned Web space.

For helpful links, see the resource page at:
<http://www.ti-me.org/members/2BInternet/>

Here is a screen shot of a typical guestbook.



Notes:

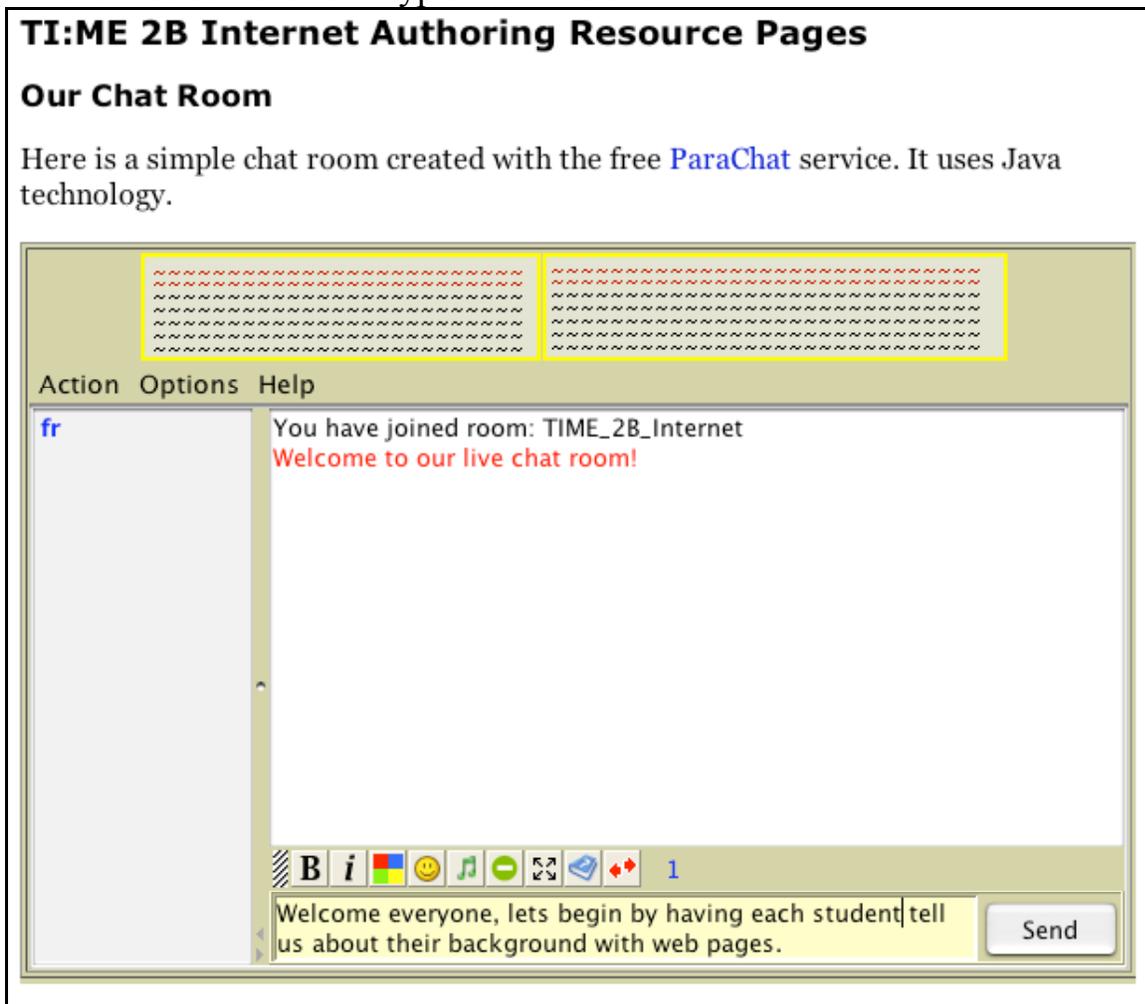
Activity 9.5: Creating a Chat Room

A chat room on your site allows visitors to communicate with other visitors in real time using a text-only interface. Many companies on the Web make ad-supported chat rooms available for free. Chat rooms are useful but to protect children who use your site, you may wish to make your chat room available in a password-protected area of your site. That way only you and your students can use the chat room.

Each IST will create an account with a company providing chat rooms. Each IST will embed a chat room on a standard XHTML page and upload it to their assigned Web space.

For helpful links, see the resource page at:
<http://www.ti-me.org/members/2BInternet/>

Here is a screen shot of a typical chat room.



Notes:

Activity 9.6: Using a Discussion Group

Discussion groups (formerly known as listservers) are Web-based communities where members can share their thoughts with the group. Members may choose to receive email when a member posts content to the group or they may choose to view the content using the group's Web site. Google and Yahoo are among many providers of free discussion groups online. You will find a link to a Yahoo discussion group for TI:ME 2B Internet classes at our resource page:

<http://www.ti-me.org/members/2BInternet/>

School teachers can use discussion groups to help students communicate and collaborate with one another and to build a sense of community and extend the classroom.

Each IST will join the TI:ME 2B Internet discussion group, post a series of messages, and read messages on the group's Web site.

Here is a screen shot of a typical discussion group.

The screenshot shows the Yahoo! Groups interface for the '2BInternet' group. The page title is '2BInternet · TI:ME 2B Internet Classes'. The left sidebar contains navigation links: Home, Messages, Members Only (Post, Chat, Files, Photos, Links, Database, Polls, Members, Calendar, Promote), Info, and Settings. The main content area is titled 'Home' and includes a 'Join This Group!' button. Below this, it states 'Activity within 7 days: (No Activity)'. The 'Description' section explains that the group is a forum for students taking TI:ME 2B Internet Authoring, with a resource page link: <http://www.ti-me.org/members/2BInternet/>. It also notes that members should login with a members-only password. The 'Most Recent Messages' section shows '(No messages for this group)'. The 'Group Email Addresses' section lists: Related Link: <http://www.ti-me.org/members/2BInternet/>; Post message: 2BInternet@yahoogroups.com; Subscribe: 2BInternet-subscribe@yahoogroups.com; Unsubscribe: 2BInternet-unsubscribe@yahoogroups.com; List owner: 2BInternet-owner@yahoogroups.com. A second 'Join This Group!' button is located at the bottom of the main content area.

Notes:

Activity 9.7: Support Sites for Teachers

During the past decade, a number of services have arisen to allow teachers to more easily create Web pages, quizzes, calendars, games, and other online activities to support their classes. In 2006, some of the most popular services include Quia.com, 4Teachers.org, and TeacherWeb.com. These services charge a fee but often have a free-trial period for evaluation.

For links, see our resource page:

<http://www.ti-me.org/members/2BInternet/>

Each IST will sign up for a free trial account at a teacher support site and create one or more quizzes, calendars, games, or other online activities for students.

Notes:

Activity 9.8: Online Calendars

One of the most useful things you can provide for your students is a calendar of school music events. Several companies make free, ad-supported calendars available to the public. In 2006, these include Calendars.net, Google Calendars, and Yahoo Calendars. You can create calendars and control who can view them and even allow students to edit them if you like.

For links, see our resource page:

<http://www.ti-me.org/members/2BInternet/>

Each IST will sign up for a free calendar service and create a publicly-viewable calendar with three or more calendar entries. Each IST will create a link to their calendar on a standard XHTML page and upload it to their assigned Web space.

Here is a screen shot of a typical online calendar.

The screenshot shows the Yahoo! Calendar interface for a user named 'sgestrella'. The calendar is set to July 2006. The interface includes a search bar at the top right, a navigation menu with 'Mail', 'Addresses', 'Calendar', and 'Notepad', and a 'Welcome [guest] [Sign In]' message. The main calendar area shows a monthly view with events listed in a grid. The events are:

- Week 26: No events.
- Week 27: 'CCSU/SMI Instructional Software and Multimedia AND Creating Web sites' on Monday (3), Tuesday (4), Wednesday (5), Thursday (6), and Friday (7).
- Week 28: No events.
- Week 29: 'TI:ME 1A in Pinellas County, FL' on Monday (17), Tuesday (18), Wednesday (19), Thursday (20), and Friday (21).
- Week 30: No events.
- Week 31: No events.

At the bottom of the calendar, there are links for 'Previous Month' and 'Next Month'. A message at the bottom right states 'You cannot edit this calendar.'

Notes:

Activity 9.9: Course Learning Systems

Course Learning Systems are full-featured Web sites that provide many services to teachers including:

- providing dedicated web space for each class
- storing student data and linking to the school registration system
- hosting of class documents such as syllabi and schedules
- providing email communication among students and faculty in a class
- providing chat rooms and discussion forums
- providing self-grading quizzes

Course Learning Systems are invaluable in distance learning but are also enhance communication in traditional courses. In 2006 the most popular Course Learning Systems are BlackBoard and WebCT. These programs are usually purchased and installed at the institution level. They can be customized to reflect the logo and style of the hosting institution.

For links, see our resource page:

<http://www.ti-me.org/members/2BInternet/>

Each IST will explore the feature set of one course learning system service. The instructor will lead a class discussion and show examples of course learning systems in action.

Here is a screen shot of a typical course learning system customized for a particular institution.

The screenshot displays the elearn.vfcc.edu website. At the top left is the VALLEY FORGE logo. The top navigation bar includes 'Home', 'Help', and 'Logout' icons, and the URL 'elearn.vfcc.edu'. Below the navigation bar are buttons for 'Welcome', 'Courses', and 'Community'. The main content area is titled 'Welcome' and features a 'Modify Content' and 'Modify Layout' button. On the left, there are two vertical panels: 'Tools' with links like 'Announcements', 'Calendar', 'Tasks', 'View Grades', 'Send Email', 'Address Book', and 'Personal Information'; and 'Links' with links like 'Assemblies of God', 'Campus Calendar', 'Campus Feedback', 'Forge Friends', 'SRC', and 'Support'. The central area is titled 'My Announcements' and lists several announcements, including 'Have You Used the Community Tab?', 'Basics of Online Learning', and 'Welcome to "Advanced Blackboard"'. On the right, there are two panels: 'This Day in History' with a link to 'Read the complete story here' and a historical entry about '1935 - LAWRENCE OF ARABIA DIES'; and 'My Courses' which lists 'Courses you are teaching' (TC01: Training Course 01) and 'Courses in which you are enrolled' (BOL101: Basics of Online Learning, TC02: Training Course 02, Training Course 03, Training Course 04, Training Course 05, Training Course 06).

Notes:

Activity 9.10: Site Search

As your Web site grows, it becomes more difficult for visitors to find all the content you have developed. Google, Yahoo, and others make it easy to add a search feature to your site. All it takes is signing up with their service and copying a small bit of code into one of your XHTML pages.

For links, see our resource page:

<http://www.ti-me.org/members/2BInternet/>

Each IST will sign up for a free site search utility. Each IST will add site search code to a standard XHTML page and upload it to their assigned Web space.

Notes:

Topic 10: Final Project

The instructor will lead a class discussion about Web sites that are useful to music teachers. Each IST will review the music education sites linked on the resource page (<http://www.ti-me.org/members/2BInternet/>) and begin planning to create a site.

The minimum specifications for a final project are as follows:

- The project must contain a minimum of 10 XHTML pages.
- The project must use one or more external CSS style sheets. ISTs must check their code to be sure that the antiquated `` tag is never used. The only exception is the use of the `` tag in code copied from online services (chat rooms, guestbooks, etc.).
- The project should have a coherent navigation scheme that permits the visitor to easily find all the available content.
- The project must use separate subfolders for images, sounds, video, and other media types as needed.
- The site may use frames but no more than 3 frames maximum. Frames-based sites must make careful use of the target attribute to avoid having other sites appear within the frameset. Each page must have a home link to allow the visitor to reload the original frameset.
- Sites without frames must have navigation available on each page.
- At least one of the XHTML pages should have embedded audio, video, or Flash media.
- At least three of the XHTML pages must link to or embed online services as described in Topic 9.

Activity 10.1: Creating a Storyboard or Outline

Each IST will create a storyboard or outline to summarize the content and services to be included in the final project. Here is an example from cardiganmiddleschool.com.

Project Outline: Cardigan Middle School Web Site

Target Audience: Students and parents.

Purpose: Student recruitment, event coordination, and communication.

Opening Page (index.htm): Large school logo with link to enter the site.

Frameset Page (frames.htm): Simple left-right frameset with navigation bar on the left and content on the right.

Navigation Page (navigation.htm): This page loads in the left frame of the frameset and remains there as the visitor browses the site. Small school logo at top. Links to each page below. Four major links: About CMS Music, Events Schedule, Audio Files, and Learning Links. Within About CMS Music there are five subtopics: History, Faculty, Curriculum, Ensembles, Requirements.

Home Page (home.htm): This page loads in the right frame of the frameset and is replaced by other content as the visitor chooses links from the navigation frame. The home page will contain a summary of the site's contents and the address and phone number of the school.

About CMS Music (about.htm): Description of the music program and links to the related sections history, faculty, curriculum, ensembles, and requirements. Home link on the bottom goes back to frames.htm with target="_top" to ensure the frameset replaces the entire window.

History (history.htm): Text history of the school with special emphasis on explaining the name. Home link back to frames.htm.

Faculty (faculty.htm): Brief biographies, photographs, and email addresses for the faculty. Home link back to frames.htm.

Curriculum (curriculum.htm): Description of the curriculum organized by grade. Gives major achievement goals for each grade. Home link back to frames.htm.

Ensembles (ensembles.htm): Description and photos of our performing ensembles. Home link back to frames.htm.

Requirements (requirements.htm): Statement of what is expected of all students. Information about the student handbook. Home link back to frames.htm.

Events Schedule (events.htm): Information about upcoming concerts and other events. This page could link to a blog or online calendar. Home link back to frames.htm.

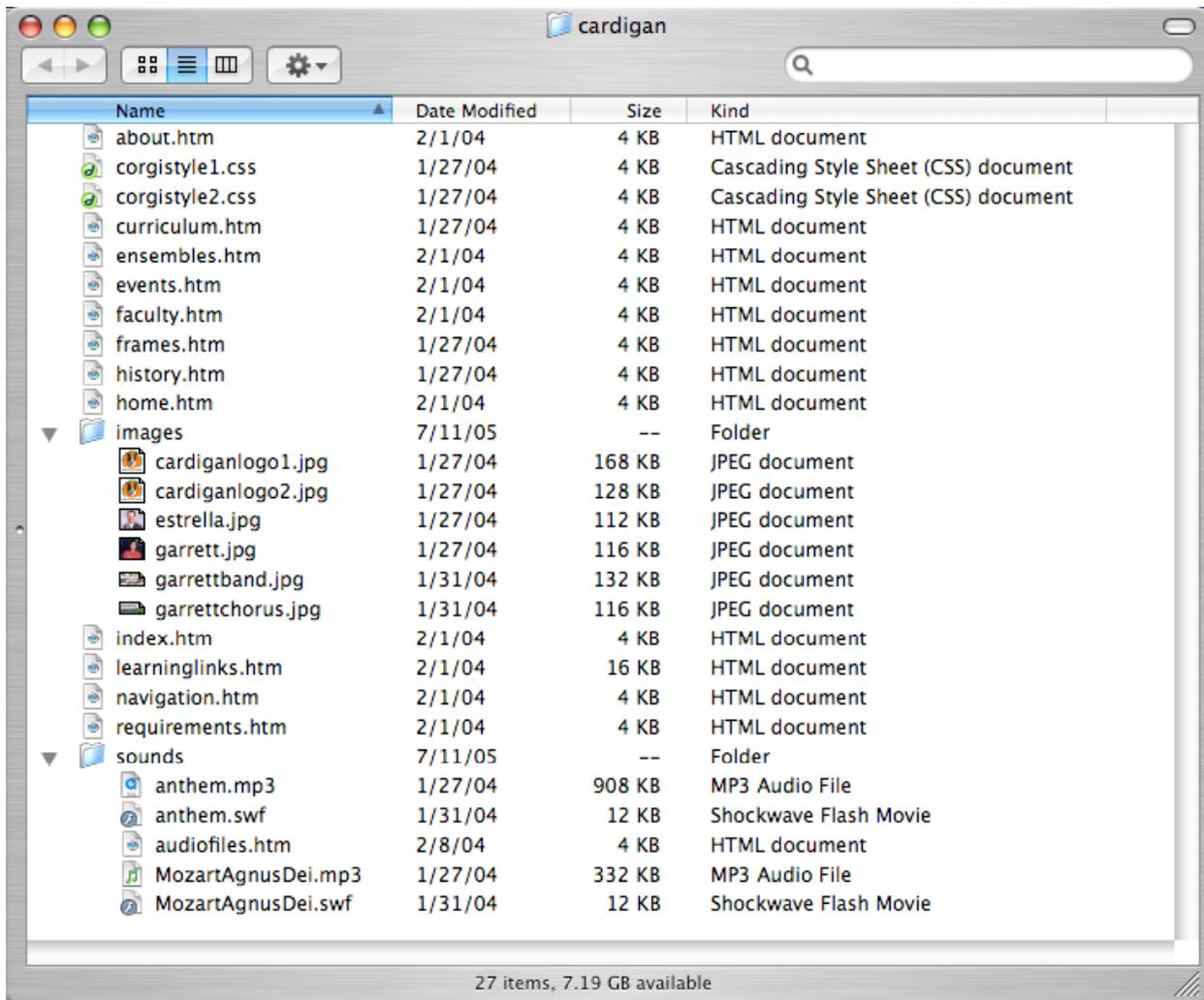
Audio Files (sounds/audiosfiles.htm): This page is located within the sounds folder because it embeds Flash media and MP3 files that must all be in the same folder as the XHTML file that embeds them. The audio files demonstrate the pride we take in our performing ensembles. Home link back to frames.htm.

Learning Links (learninglinks.htm): This page contains a rich set of links to online resources for music study. The list is organized by content area. Home link back to frames.htm.

Notes:

Activity 10.2: File Organization

Each IST will create a folder and subfolders to organize the files for the final project. Here is an example from cardiganmiddleschool.com.



Notes:

Activity 10.3: Create and Upload

Each IST will create a final project and upload all files to their assigned Web space. Each IST will send an email to the instructor when the site is complete. The email should include a complete and accurate Web site address (URL).

Notes:

Activity 10.4: Maintenance

The instructor will lead a class discussion about how to maintain a Web site. This will include:

- Registering and renewing a domain
- Obtaining hosting space and working with ISPs
- Dealing with your school district if they provide your hosting space
- Uploading files – binary vs. text protocols
- File naming conventions

Notes:

Appendix 1

Brief History of the Internet and Web Audio

(excerpt from makepages.com, used here by permission)

Read the timeline below to get a sense of the history of the Internet and the development of audio on the Web. See <http://www.dejavu.org/> for a fascinating peak at how the Web looked on early Web browsers.

1969

U.S. Defense Department's ARPANET (Advanced Research Projects Agency) project. ARPA was a Pentagon agency that sought to design a network that could withstand partial destruction and still function. Academic computer scientists were hired to develop the network, which was launched in 1969 with four nodes, or sites: the University of California at Los Angeles; the University of California, Santa Barbara; Stanford Research Institute; and the University of Utah.

1971

By 1971, ARPANET had expanded to 23 sites.

1979

USENET News network is created for academic researchers who lacked the necessary clearance to use the Defense Department's ARPANET.

1980

By 1980, ARPANET had expanded to more than 200 sites. Local area networks began to proliferate in universities, businesses, and corporations. The desire to link these networks began to grow.

Early 1980s

The National Science Foundation (NSF) built five supercomputer centers to give any academic researcher access to high-powered computers. The NSF built its own network, based on internet protocol (IP) technology (based on the ARPANET technology), to connect the five centers. Individual university networks were now linked. Soon the network connections were being used for other purposes, such as email.

1985

The National Center for Supercomputing Applications (NCSA), at the Champaign-Urbana campus of the University of Illinois, was one of the supercomputing

facilities built by the NSF. NCSA began to create tools, such as telnet, ftp, and gopher browsers, to assist people in using the internet.

1990

Tim Berners-Lee, working on a NeXT computer for the European Organization for Nuclear Research (CERN), developed the first Web browser called WorldWideWeb. The name of this browser has become synonymous with the Web as a whole. Berners-Lee finished the browser on Christmas day in 1990. He developed a standard file format called HTML that consisted of plain text marked up with special tags to tell a web browser how to display the content.

1991

In March of 1991 Berners-Lee released WorldWideWeb to the scientists at CERN to help them share ideas. Almost all content on the Web is text at this time. Most persons who use the Web at this point are running text-only Web browsers on UNIX workstations.

Many universities and other institutions use the gopher, a fore-runner to the Web and the first truly distributed information system on the Internet.

1992

By 1992, several text-only Web browsers were available for UNIX workstations, PCs running DOS, and Macintosh computers.

1993

In February of 1993, Marc Andreessen and Eric Bina from NCSA released Mosaic, the first graphical web browser, to the public for free. Mosaic ran on UNIX workstations running the X-Windows graphical shell. A few months later, a version was released for Macintosh computers. Mosaic included support for sound in Sun Microsystem's 8-bit AU format.

The Lynx hypertext browser developed at the University of Kansas is modified by student Lou Montulli in March of 1993 to allow it to display documents found on the Internet. Lynx becomes the preferred software for text-only Web browsing and it is still used today.

1994

NCSA Mosaic increases in popularity and Web authors begin to add graphics to their pages in GIF format. A few sounds are made available for download in Sun Microsystem's AU format or Microsoft's WAV format. At this time, Sun's 8-bit

AU format is the most prevalent sound format on the Web because of the high number of UNIX workstations at Universities. These sounds take minutes to download using the high-speed connections available at universities but they take hours to download on the 9600 bps modems used by everyone else. The graphics and text, however, are enough to excite the imagination of the public and the Web becomes very popular very quickly.

Netscape Navigator 1.0 is released in December of 1994. It was also known as Mozilla 1.0.

1995

By 1995 the web was the fastest-growing service on the Internet. CERN passed the mantle of web development to INRIA (the Institut National pour la Recherche en Informatique et Automatique), in France. INRIA is the European division of the World Wide Web Consortium (W3C) based at MIT in the United States. The W3C maintains the standards for the web. The most pervasive web standard is HTML (Hypertext Markup Language). HTML (now XHTML) is the language used to create web pages like the one you are reading now. Netscape's new Navigator web browser quickly overtakes NCSA Mosaic in popularity.

The JPEG (Joint Photographic Experts Group) format for high-quality photographs becomes popular and the Web becomes more graphically rich. Universities and then businesses begin to create promotional Web sites. Individuals start to create home pages. E-mail becomes the killer application of the 20th century. Chat rooms abound. Online bulletin board systems and mailing lists enter the mainstream of popular culture.

Macromedia Director, an established development environment for CD-ROM production since 1988, gives birth to Shockwave. Shockwave allows playback of Director files over the Web. The size of the plugin and the size of the Shockwave files are both impediments to the success of Shockwave.

A young programmer named Jonathan Gay creates a program called FutureSplash Animator and establishes a company called FutureWave. The company will later be purchased by Macromedia and this program will become Flash.

In April, Progressive Networks Launches RealAudio at National Association of Broadcasters Conference in Las Vegas. Rob Glaser, President of Progressive Networks, discusses RealAudio.

Netscape Navigator 1.1 is released in April of 1995.

Netscape Navigator 2.0 is released in beta form in Fall of 1995.

On August 23, 1995, Windows 95 is released and contains a Web browser of its own called Internet Explorer 1.0.

1996

RealNetworks introduced RealAudio. Streaming audio becomes possible on the Web. The quality is not great. The sound is monophonic and the fidelity is compared to AM radio. Audio lectures become popular on the Web but music doesn't sound good enough for most listeners to enjoy. Apple Computer releases the QuickTime plugin for QuickTime 2.0. This plugin works with Netscape Navigator and allows authors to embed compressed audio and video content on a Web page. Netscape releases its own LiveAudio plugin for Netscape Navigator 3.0. This offers Web authors a way to embed audio on Web pages and the plugin is scriptable using JavaScript.

FutureSplash Animator is released in the summer of 1996. Microsoft and Disney both sign up as clients of FutureWave to create more impressive Web sites for their companies. In December of 1996, FutureWave is acquired by Macromedia and FutureSplash Animator is renamed Flash 1.0. Flash 1.0 has no programming language and no sound, but objects within the Flash presentation can be hyperlinked to allow for some interactivity. Flash becomes popular for two-dimensional linear animations, especially banner ads.

In April, Progressive Networks releases RealAudio 2.0 with better sound quality (though still only monophonic), browser integration with Netscape, an open architecture to allow other developers to work with RealAudio, bandwidth detection, and firewall support.

In September, RealAudio 3.0 debuts with new features including stereo audio over 28.8 kbps modems and near CD-quality sound at higher bandwidths. New content providers include leading music labels Sony, Warner, Polygram, BMG and MCA and national broadcasting companies including CBS Radio, NBC, ABC and the CBC.

In October, Progressive Networks, Netscape, and 40 other companies announce support for a proposed standard for delivery of real-time media over the Internet and intranets. The new standard is to be called Real Time Streaming Protocol (RTSP).

Netscape Navigator 2.0 is released in March of 1996.

Internet Explorer 2.0 on Windows begins to add features to compete with Netscape Navigator. At this point Navigator is clearly the better browser but Internet Explorer gains market share simply because it is given away for free by Microsoft. Updates to both browsers were frequent during 1996.

Internet Explorer 2.0 for Macintosh is released in April of 1996. Netscape Navigator 3.0 is released in August of 1996. Internet Explorer 3.0 for Windows is released in August of 1996.

Internet Explorer Browser market share estimate in Fall of 1996 is 33%.

1997

QuickTime and RealAudio increase the quality of compressed audio that can be streamed to Web visitors. RealVideo is introduced in February. In September, Progressive Networks changes its name to RealNetworks and makes RealPlayer 4.0 available. RealPlayer 4.0 plays both audio and video streams. RealPlayer 5.0 is released in November. RealSystem 5.0 is a client-server streaming media system that includes RealPlayers, RealServers, and publishing tools. RealSystem 5.0 provides a suite of high quality, cross platform products for viewing and delivering real-time multimedia over the Internet and corporate intranets. Features include full-screen video, integration of Flash animations and commerce features such as advertisement insertion and pay-per-view.

Microsoft's Internet Explorer browser begins to attract interest because it is free. Microsoft begins to include Windows Media Player with the browser. Flash 2.0 adds support for streaming sound.

Netscape Navigator 4 is the current Netscape offering and its share of the browser market has diminished to less than 50%. Web developers no longer feel comfortable developing pages that rely on the LiveAudio plugin because Internet Explorer 4.0 does not support it.

The Mosaic browser reaches version 3 and is then discontinued. Internet Explorer 3.0 for Macintosh is released in January of 1997. Netscape Navigator 4.0 is released in June of 1997. Internet Explorer 4.0 for Windows is released in October of 1997.

1998

Apple licenses new compression technology from Sorenson Vision Inc, QDesign Corporation (music compressor) and Qualcomm (Pure Voice compressor) for use in QuickTime 3.0. Roland Corporation's Sound Canvas sound set is also included, providing over 200 MIDI-compatible sounds and instruments. QuickTime 3 can play back more than 30 different audio and video file formats and offers real-time streaming of digital content over the Internet. Flash 3 adds more interactivity through pre-set behaviors. No scripting language is available for Flash at this time. In April, RealNetworks announces support for the W3C's proposed standard called Synchronized Multimedia Integration Language (SMIL). In July, RealNetworks releases RealSystem G2. The new RealPlayer G2 software has better codecs for music and more reliable streaming technology for uncertain network conditions. The software has one-button access to favorite programming from more than 50 leading media content providers. RealSystem G2 also includes native support for most existing media file formats on the Web including ASF, AVI, JPEG, MPEG, VIV, and WAV. RealNetworks advertises its support of open standards such as SMIL and RTSP.

Netscape releases its browser code to open source and makes its Communicator suite of browser and e-mail software freely available to the public. The Mozilla project begins work on an open-source, standards-compliant Web browser. Internet Explorer 4.0 for Macintosh is released in January of 1998. Netscape Navigator/Communicator 4.5 is released in October of 1998.

1999

QuickTime 4.0 is released. Streaming is improved and support for MP3, SMIL, and other formats is added. More than 5 million people download QuickTime 4 during the first 50 days after its release. Content providers take notice. Soon movie trailers and news are available in large numbers on the Web using QuickTime. Apple and Akamai create a high-quality network for Internet streaming. Flash 4 now supports MP3 streaming and can play QuickTime movies with QuickTime sound compression. In turn, QuickTime 4 can play Flash SWF files. Sweet!

In May, RealNetworks announces RealJukeBox, the first complete digital music system for the Internet. When consumers listen to an audio CD from the CD-ROM drive of their PC while connected to the Internet, the RealJukebox will reach out to the CDDB database and automatically download the name of the artist, album and song to the desktop. CDDB's DRS identifies over 360,000 CD album titles, making it by far the largest online CD information source in the world. In November,

RealPlayer 7 is released with performance and navigation improvements and better support for MP3 files.

Internet Explorer 4.5 for Macintosh is released in January of 1999.

Internet Explorer 5.0 for Windows is released in March of 1999.

Internet Explorer Browser market share at end of 1999 is more than 50%.

2000

QuickTime downloads exceed 100 million. Apple releases Darwin Streaming Server as an open source initiative. Many new streaming servers are added to the Internet using this free software. Apple and RealNetworks announce a collaboration for Internet streaming. RealNetworks and Apple also agree to support the principles of the “Ask, Tell, Help” initiative, which promotes good ‘Internet manners’ by ensuring that each company’s player applications will inform and ask the user before changing the user’s default player selection for common media formats.

Flash 5 is released with a major overhaul of the scripting language, ActionScript. It now becomes possible to build extremely interactive audio applications such as mixers, interactive music analysis, mousejam type improvisation environments, music drills, and complex games.

Netscape 6 is released. The browser model and LiveAudio plugin from version 4.x of Netscape is completely abandoned in favor of support for standard technologies endorsed by the W3C.

RealAudio 8.0 is released by RealNetworks providing CD quality sound at 64Kbps – half the size of MP3.

Internet Explorer 5.0 for Macintosh is released in March of 2000.

Netscape Navigator 6.0 is released in November of 2000.

Internet Explorer 5.5 for Windows is released in July of 2000.

Flash is well-established as a complete multimedia development environment with 500,000 developers and 250 million users of the player.

2001

QuickTime 5 is released and more than 100 million downloads are recorded during 2001. CNN, NPR, HBO, ESPN Radio, The Cartoon Network, Disney, Warner

Bros., Honda, General Motors, Nokia, and many other companies adopt QuickTime for delivery of audio and video content on the Web.

In September, RealNetworks unveils RealOne Platform, an integrated media platform for connecting consumers to content. At the core of the platform is its media engine, RealOne Player, which combines the RealPlayer and RealJukebox and adds to them a new media browser.

Internet Explorer 6.0 for Windows is released in October of 2001.

2002

QuickTime 6 is released in July and surpasses 100 million downloads in less than one year. QuickTime 6 is chosen as the basis for the MPEG-4 open standard.

Flash MX adds support for integrated video within Flash using Sorenson compression. A number of pre-designed components make it easier to design interactive audio applications with sliders, knobs, check boxes, and buttons.

Netscape 7 is released and is, arguably, the most standards-compliant browser available. Mozilla also releases its 1.0 browser based on much the same code-base as Netscape. Internet Explorer 5 and 6 are the most popular Web browsers in use, however, because each is bundled with a version of Microsoft Windows.

Netscape Navigator 7.0 is released in August of 2002.

Phoenix, a new browser by Mozilla, is made public on September 23, 2002
Apple's new Safari browser is added to MacOS X.

2003

Apple revolutionizes the online music world by releasing iTunes Music Store for Macintosh in Spring 2003 and for Windows in October of 2003. Apple succeeds in addressing the intellectual property concerns of the recording industry and creates an attractive and inexpensive way for consumers to buy audio tracks and burn them to CDs. Apple pioneers the use of AAC (advanced audio codec) format to limit the distribution of digital audio files and protect the interests of copyright holders while giving consumers the freedom to store and play their music files on multiple devices. In Fall of 2003 competitors begin to adopt the iTunes model.

Director MX is released with more integration of Flash SWF files. The SequenceXtra plugin allows for interactive MIDI on the Web using Director MX

but it is not reliable and consistent yet on all platforms. Flash MX 2004 adds more support for data driven Web sites and more pre-built components.

In August, RealNetworks acquires Listen.com and its Rhapsody digital music service providing additional competition to iTunes Music Store.

Microsoft releases Windows Media Player 9 with smart jukebox features, easy CD transfer to computer, CD burning, and compatibility a range of portable players with the notable exception of the iPod, the most popular portable music player.

Microsoft stops development of Internet Explorer for Macintosh.

Safari becomes a popular Web browser on Mac OSX.

Internet Explorer market share is near 95%.

Phoenix browser changes name to Firebird, then Firefox for beta release.

2004

The web has become the dominant platform for the delivery of information worldwide. The pace at which homes and schools get connected to the internet continues to increase rapidly. More and more educational content and interactive learning experiences are becoming available on the web. Sound, animation, and video use on the internet is increasing rapidly as high-speed connections are installed worldwide. As the web becomes more of a multimedia environment, the applications of the web to music education become more attractive.

Firefox captures substantial market share while still in beta. The official 1.0 release takes place on November 8, 2004.

Browser Market Share Summary for October 2004

Microsoft Internet Explorer	92.31%
Firefox	2.69%
Netscape	2.14%
Safari	1.46%
Mozilla	0.82%
Opera	0.49%

2005

Blogging, file sharing, and dynamic pages are now normal parts of everyday web experience. More and more tools such as quia.com, youtube.com, myspace.com, and blogger.com are allowing the general public to contribute substantial content to the Internet.

Adobe purchases Macromedia. During the transition, Macromedia releases Studio 8 containing new versions of Flash, Dreamweaver, and Fireworks. Adobe begins to integrate the Macromedia product line into its own offerings.

Firefox gains in popularity. The slow, steady decline of market share for Internet Explorer continues.

Browser Market Share Summary for March 2005

Microsoft Internet Explorer	88.59%
Firefox	6.71%
Netscape	1.80%
Safari	1.73%
Mozilla	0.62%
Opera	0.46%

2006

Browser market share constantly fluctuates but Firefox is becoming more popular due to its security advantages. Most Windows users still use Internet Explorer because it comes pre-installed but more sophisticated users increasingly switch to Firefox on Windows to avoid contracting viruses and spyware. With 10% market share for Firefox, developers across the world are forced to avoid Internet technologies that only work with Internet Explorer.

Microsoft stops development of Windows Media Player for Macintosh and enters into an agreement with codex vendor Flip4Mac to provide a codex to allow QuickTime player to play Windows Media files.

Browser Market Share Summary for March 2006

Microsoft Internet Explorer	84.70%
Firefox	10.05%
Safari	3.19%
Netscape	1.05%
Opera	0.54%
Mozilla	0.34%

Appendix 2

Web Audio Today (2006)

(excerpt from makepages.com, used here by permission)

During the dotcom madness of the late 1990s there were many predictions about the future of Web audio. Some folks imagined a convergence of film, television, and the Internet. That has not happened for a number of reasons but the most important reason it has not happened is that high-speed connections to the Internet (aka broadband) are just now becoming more common than slower dial-up connections.

Another important reason, however, is that of user expectations. Television had audio from the beginning and all the content designed for television was designed with audio in mind. The Internet, on the other hand, was designed for exchange of text documents and most people view the Internet more like an electronic book or library than as a broadcast medium. As a result, exciting discussions about adding film-style soundtracks to Web sites and using sounds in rollover buttons and other user interface features need to be tempered with an understanding of how people actually want to use the Web.

For example, in a typical business setting, library, or school computer lab, Web audio may be viewed as a distraction. This problem can be avoided by following the principle of consent and control. Web site visitors should never be surprised by audio. Instead, Web sites that have audio should allow the visitor to control it or turn it off completely. In most cases, audio should be turned off on a Web site by default. When audio is available on a web site it can serve many purposes.

- Webcasts, Streaming, and Compression
- e-Commerce
- e-Learning
- Music Education
- Music Marketing and Promotion

Webcasts, Streaming, and Compression

Webcasts are broadcasts on the Web. Webcasts can be live events or recorded material and like traditional broadcast media, webcasts can be viewed by many people at the same time. When a Web site serves recorded material, each visitor can begin playback at any time. One visitor may be half way through a video

webcast while another is just beginning the playback. As a result, webcasts offer enormous flexibility for the delivery of presentations on the Web.

Webcasts are made possible by a technology known as **streaming media**. Before streaming came on the scene in 1995, media files had to be completely downloaded before any playback could begin. Those with slow connections to the Internet had to wait minutes or even hours for audio or video data to download. Streaming changed all that. Digital content is first compressed and then broken into small packets of data. These packets are sent sequentially over the Internet to the Web site visitor. The visitor's computer places these packets into a buffer. When the buffer has enough packets then playback begins. During playback, the visitor's computer continues to download, buffer, and queue up more packets to create continuous playback. Of course, when the network gets congested, the playback of one set of packets sometimes ends before the next set of packets is ready. At that point, the broadcast ceases to be continuous and the visitor usually misses some content.

It is technically possible to stream media using the old Hypertext Transport Protocol (HTTP) but the nature of that protocol is not conducive to the continuity needed for music and video. The User Datagram Protocol (UDP) and the more recent Real Time Streaming Protocol (RTSP) were both developed to make streaming more reliable. Both of these protocols keep sending data even if previous packets never reached their destination. That puts a priority on continuity so visitors may experience only an occasional hiccup in the audio or video stream instead of major gaps. Today RTSP is the preferred protocol in use by all major streaming technologies.

After the visitor is finished viewing the streaming media, you might think that the complete media file is stored somewhere on the visitor's hard drive and would be available for off-line playback at some other time. That is not the case with streaming. If the visitor wishes to view the media again, an Internet connection is required and the streaming process must begin anew. Streaming, therefore, affords the content provider a bit more control over their intellectual property. Savvy computer users, of course, can use any of a number of software programs to capture the stream to their hard drives for later off-line playback.

Streaming alone, however, would not be adequate for sending lengthy broadcasts over the Internet. **Compression** of the data is also needed. Uncompressed audio and video files are far too large for practical transmission on the Internet. A single minute of stereo audio at CD quality, for example, takes up about 10 megabytes of

data. When compressed, however, the same audio often takes up only 1 megabyte or less. The best compression technologies today, such as MP3 for audio and MPEG-4 for video, create very small audio files that sound almost as good as the original uncompressed audio. Compression technologies for audio and video reduce file size by removing data from the file according to a set of rules called a **codec** (COmpression DECompression). When the visitor receives the smaller file, the media player application (RealOne, QuickTime, or Windows Media Player) attempts to reconstruct the original file using the same set of rules.

One way to understand compression-decompression is to imagine you need to send someone a series of important dates from the 20th century such as the following dates when major wars began. A very simple codec might remove the first two numbers of each set of four. The dates could then be transmitted in a file half the size of the original.

Uncompressed Data	Compressed Data
1917194519501961	17455061

When the visitor receives this file, our simple codec inserts 19 before each set of two numbers to reconstruct the original file. This simple example would result in a **lossless compression** because the final reconstructed file would be identical to the original. Unfortunately, Audio and video compression is more complex and the final reconstructed file is not identical to the original. This is known as **lossy compression**. Today's compression technologies, however, are so refined that the loss of quality is often minimal and the quality is more than acceptable for most people.

How Does Streaming Work?

Streaming media requires an audio and/or video source, a computer for encoding and compressing the data, and a dedicated server to send the encoded data at whatever rate is needed by the end user. A dedicated streaming media server is required to ensure that the computer serving the files is not burdened with requests for other types of files such as Web pages or graphics.

From the visitor's point of view here is what happens.

- The visitor clicks a link to view or listen to streaming media. The URL for the link probably begins with `rtsp://` because the file is served with Real Time Streaming Protocol (RTSP) instead of HTTP.

- The Web browser contacts the streaming media server at the address found in the link. The browser asks the streaming media server to begin sending the media.
- The streaming media server begins to send data packets to the visitor's computer.
- The visitor's computer responds by launching a plugin or application and begins storing the data packets in a buffer.
- After a few seconds of buffering the playback begins.
- During playback, additional packets are received, stored in the buffer, and placed in the queue for playback.
- If network conditions are good, the playback will be continuous.

When To Use Streaming

If your Web site uses short sound samples of 20 seconds or less, setting up a dedicated streaming media server is probably not a great use of your time. Connection speeds today are adequate to allow visitors to download short audio samples so streaming isn't really needed. Streaming is most appropriate when you wish to make large media files available for viewing by tens or hundreds of people at the same time.

Who is Streaming?

If you have the bandwidth today, you can enjoy news, sports, music, film, radio, and other entertainments using a Web browser in combination with Web browser plugins and stand-alone applications. Webcasts can be either live or pre-recorded digital audio and/or video that is streamed to the visitor. Almost all the major media companies offer Webcasts today and it is becoming easier for smaller Web sites to offer streaming media files. The streamed file can be either embedded in a Web browser window using a plug-in or ActiveX control or played by a stand-alone application.

What are the Formats?

The three major streaming technologies today are RealOne, Windows Media Player, and QuickTime. RealOne, from RealNetworks, is the current king of streaming media and for very good reason. RealNetworks has done a great job of making an easy-to-use player (RealONE player) that works reliably on Windows, Macintosh, Linux, and various UNIX platforms. They also make streaming server technology and offer business services to help content providers track and manage large numbers of users. RealNetworks has products for every stage of streaming media development from content creation to delivery.

Content Creation Tools	Helix Producer Plus
Native codecs	realAudio8 and realVideo9
Other supported codecs	MPEG-4, AAC, MP3, QuickTime, and many others
Digital Rights Management	Helix DRM
Streaming Server Software	Helix Universal Server
Client-Side Media Player Application	realONE player

RealNetworks markets a streaming server called the Helix Universal Server (previously RealServer 8). The Helix Universal Server is very popular because it provides services vital to streaming media customers. First, it delivers content specially encoded for different bandwidths. If a visitor is using a dial-up modem, the server automatically chooses media encoded for slower connections. The quality of the experience, therefore, varies with the bandwidth of the visitor. Second, the Helix Universal Server supports the RealVideo 9 codec as well as codecs from QuickTime, Windows Media Player, and about 50 other formats. RealNetworks technologies also support digital rights management technologies to protect intellectual property rights of content creators.

Content Creation Tools	Microsoft Producer
Native codecs	Windows Media Audio and Video 9 Series
Other supported codecs	MPEG-4
Digital Rights Management	Windows Media DRM 9 Series
Streaming Server Software	Windows Media Services 9 Series (part of Windows Server 2003)
Client-Side Media Player Application	Windows Media Player (No Linux or UNIX version) (MacOS X version is missing features.)

Microsoft has its own complete package for creating and delivering streaming media. Microsoft's solution consists of a set of authoring applications (Microsoft Producer and Windows Media Encoder), a streaming server solution that works only with Microsoft Web servers, and a player application (Windows Media Player) that works on Windows and to some extent Macintosh. Microsoft does not

support any UNIX or Linux platforms. Its lack of broad and consistent platform and browser support detracts from what is otherwise a good product. Also, Microsoft's streaming server uses the older UDP protocol whereas RealNetworks and Apple QuickTime both use the newer RTSP standard.

Content Creation Tools	Final Cut Pro, SoundTrack, QuickTime Player Pro, and many third party tools.
Supported codecs	More than 100 including MPEG-4, MP3, AAC, all major graphic formats, Flash 5, 3GPP, QuickTime VR object and panoramas
Digital Rights Management	AAC codec provides DRM for audio. DRM for both audio and video is available when QuickTime content is served with RealNetworks' Helix Universal Server.
Streaming Server Software	QuickTime Streaming Server 5 (part of MacOS X server) Darwin Streaming Server (Open Source)
Client-Side Media Player Application	QuickTime Player (Macintosh and Windows)

Apple makes the QuickTime Streaming Server and QuickTime Player application. They also make QuickTime Broadcaster for encoding live webcasts. QuickTime supports more than 100 media types including MP3, Flash 5, all major graphic file formats, DV, MIDI, and many more. Apple's QuickTime Streaming Server works on MacOS X server but Apple has also created an open source version called Darwing Streaming Server that can be used on Linux and most UNIX systems. QuickTime is the choice among video professionals and QuickTime also offers much to audio professionals as well.

One Man's Opinion

My experience with the various Webcast technologies is that they are all great for convention keynote addresses, investor relations speeches, and slide shows with background audio and narration. When the focus is on music, however, I have found that QuickTime delivers the best quality audio and video. That's probably why movie trailers are almost exclusively made available on the Web in QuickTime format.

Appendix 3

Web Sites with Audio for Music Education (excerpt from [makepages.com](http://www.makepages.com), used here by permission)

This page lists some interesting sites that use audio to teach music. Please note that each link on this page will open in a new window. Here you will also find some focus questions for evaluating music education Web sites.

- Examples of Music Instruction Resources on the web
 - Music Theory
 - Ricci Adam's MusicTheory.net at <http://www.musictheory.net> (Flash audio and interactive animations — EXCELLENT SITE!)
 - Gary Ewer's "Easy Music Theory" at <http://www.musictheory.halifax.ns.ca/index.html>
 - Dr. Estrella's media suite that accompanies *Music First!*, a theory textbook by Gary White published by McGraw-Hill. These activities are available online at: [The McGraw-Hill Music First! Online Learning Center](#) (Flash audio and interactive animations — EXCELLENT SITE!)
 - Dr. Estrella's media suite that accompanies his textbook *Study Outline and Workbook in the Fundamentals of Music*. These activities are available online at: [The McGraw-Hill Online Learning Center](#)
 - More links can be found at <http://www.stevenestrella.com/mused/websites.html>
 - Music History
 - Internet Public Library's Music History 102 at <http://ipl.si.umich.edu/div/mushist/> (RealOne used for audio examples.)
 - Dr. Estrella's Incredibly Abridged Dictionary of Composers at <http://www.stevenestrella.com/composers/> (Links to amazon.com for audio examples.)
 - Music Composition
 - Composers in Electronic Residence at <http://www.edu.yorku.ca:8080/~CIER/>
 - Young Composers at <http://www.youngcomposers.com/>
 - Vermont MIDI Project at <http://www.vtmidi.org/>
 - Sites by Music Teachers
 - Home Practice Online at <http://astro.temple.edu/~swatson/>
 - MusicTechTeacher.com at <http://www.musictechteacher.com>
 - Web improvisation experiences
 - Lullabye at <http://www.makepages.com/lullabye/lullabye.html>
 - BluesJam at <http://www.makepages.com/bluesjam/bluesjam.html>

- See chapters 1, 2, 5, and 11 of McGraw-Hill's Online Learning Center for *Music First!* at <http://www.mhhe.com/musicfirst>
- See chapters 2 and 3 of McGraw-Hill's Online Learning Center for *Study Outline and Workbook in the Fundamentals of Music* at <http://www.mhhe.com/hill11>
- Focus Questions for Evaluating Online Music Education Resources
 - Is there a primary author or responsible person? Is he or she an expert?
 - Has the site content been reviewed?
 - Is it free or does it require payment?
 - How is the site supported financially? Will it endure?
 - How recently was it updated? Is it a ghost site?

Appendix 4

Copyright Concerns for Web Developers (excerpt from makepages.com, used here by permission)

This page contains information on copyright of interest to persons creating audio for use on the Web and in interactive media. Here I will provide some general principles you should understand when creating your own audio content or using someone else's.

DISCLAIMER: The author is not a lawyer. These guidelines are researched and should be of assistance to you but they should not be relied on as the sole source of legal advice for any work you do.

Original Content

Copyright law exists to encourage the creation of new artistic works by providing a financial incentive to the creators. If you produce original music, video, art, or literature, then you own the copyright. That means that Congress confers upon you the exclusive right to copy and sell what you own. If anyone else copies or sells your content (or any portion of it) without your permission, you can file suit and collect substantial damages (\$500 to \$100,000 per infraction). You own the copyright, so you control how your work is used. That means you are free to use it in any work you do. Of course, if you own the copyright for a piece of music, that also means you can sell that copyright to another if you so desire.

If you are producing original audio for a client, you need to have a clear written agreement about who owns the copyright for the completed work. In general, any thing you do for a client that is labeled as a work-for-hire belongs to the client entirely unless you expressly exclude portions of the work that are under license to others. That means that when you complete an audio project for a client and they pay your invoice, they have purchased the rights to your music. At that point, the client owns the copyright and you don't. The point here is to communicate clearly in writing with your client so that no misunderstandings arise later.

Copyright law grants content creators exclusive right to copy and sell their content for a limited time. When the first U.S. copyright law was passed in 1790, the term of copyright was only 14 years with a possible extension of another 14 years. Over time, a number of political and economic factors have persuaded Congress to extend the term of copyright. Today the copyright of a work endures for a term consisting of the life of the author and 70 years after the author's death. In 1998,

the Sonny Bono Copyright Term Extension Act was passed to help align U.S. copyrights with international copyrights. As a result, nothing new will pass into the public domain until January 1, 2018. This new law is often called the "Mickey Mouse Law" because of Disney's interest in it. Without this law, Mickey Mouse would have entered the public domain on January 1, 2003. Donald Duck, Goofy, and many others would follow. The point here is that copyright is a matter of law and the term of copyright can be changed through legislation.

Public Domain, Royalty-Free Loops, Mixer Kits, etc.

Another way to legally add music to a Web site or media project is to use public domain audio files or purchase CDs or DVDs containing royalty-free audio. Public domain audio can be used without any restrictions. Royalty-free audio, however, has some restrictions. Generally you receive a license when you purchase CDs of royalty-free music. That license allows you to use the music in the products you produce for your clients but specifically prohibits you from reselling the music itself. In other words, the music still belongs to the company that sold you the CD, they are just licensing it for use by you and your client.

Sometimes you can find Sample CDs at your local music store but you can preview and purchase so much more using the Internet. Here are some links to get you started.

- Sample Arena - <http://www.samplearena.com>
Here you can find public domain dance and trance loops. A small number of loops can be downloaded for free in the guest section. Subscribers to the site gain full access.
- Killersound - <http://www.killersound.com/>
Killersound creates music and sound design for new media. This is a great site for loops, sound effects, and mixer kits in many genres. The mixer kits are particularly interesting. Each kit comes with a set of loops that all work together. You can use these loops in mixers such as the sample 3-track mixer found on MakePages.com. Killersound makes their files available for purchase as downloads or CDs.
- SoundShopper - <http://www.soundshopper.com/>
This site is very similar to Killersound.
- Search on Google for Royalty Free Loops
- GarageBand and SoundTrack by Apple Computer
Acid by Sony
These software packages come with thousands of royalty free loops. The software makes it easy to mix loops together to create original compositions. Each loop is automatically transposed and synchronized to whatever key and

tempo you choose for your project. These amazing software products make audio development for media a lot easier than it was a few years ago.

Licensing Content

If you just have to use content that is owned by another, then you will have to approach the copyright owner for permission. In this regard you have to consider the different types of licenses. The **mechanical license** covers use of a composition and protects ownership of the composition regardless of who performs it. For example, if you are a piano player and you record *I Got Rhythm* (1930) from the Gershwin songbook, you will have to pay a mechanical license fee even though you are the performer on the recording. On the other hand, if you record *Maple Leaf Rag* (1899) by Scott Joplin, there are no mechanical license fees to pay because "Maple Leaf Rag" is in the public domain. A special type of mechanical license is called a **synchronization license**. If you are using copyrighted music in a Flash animation or other interactive media, you will need to negotiate a synchronization license with the copyright owner.

A second important type of license is the **Master License** which covers use of existing recorded performances. If you wish to use an excerpt from a CD performance of any work, you must have a Master License. Even if the work being performed is in the public domain, the recorded performance of that work can still be copyrighted. That means that if you create and record original performances of public domain works, then you will own the rights to those recordings. In almost all cases, the Harry Fox Agency is your contact to negotiate a license to use someone else's music content or performance. The Harry Fox Agency is in charge of licensing for the National Music Publishers Association (NMPA at nmpa.org). NMPA is also a great place to learn more about copyright and licensing.

Fair Use

Many people have heard of the Fair Use provision of the copyright law. This provision allows scholars, critics, and teachers to copy portions of copyrighted works for the purposes of teaching, criticism, and scholarship but not for performance. The excerpts must not comprise a part of the whole that would constitute a performable unit such as a section, movement or aria. In no case can the excerpt be more than 10% of the whole work. The number of copies distributed cannot exceed one copy per student.

Appendix 5

XHTML Basics

(excerpt from makepages.com, used here by permission)

Web pages can be created by many different types of software programs from the humblest text editors like TextEdit or TextWrangler (Mac) and NotePad or NoteTab Lite (Windows) to full-featured Web site development tools like Dreamweaver and GoLive. Regardless of how you create the file, the file itself is a plain text file that can be read by any computer. One of the original purposes of the Web was to allow different computer platforms to share files, so plain text was a great choice to ensure compatibility.

Within an XHTML file you will find content surrounded by special markup tags. For example, text that is intended to be displayed as **bold** will be surrounded by the markup tags `` and ``. These codes tell the Web browser that reads the page where to begin boldface and where to end it. All content on a Web page is contained by one or more tags. This is the **Principle of Containment**. One way to ensure that your content is properly contained is to type the opening and closing tags before you insert your content. In older version of HTML, like HTML 4.01, tags can be upper or lower case letters. XHTML, the very latest version of HTML, requires lowercase tags.

Sample XHTML Document

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>HTML Lesson 1: A Basic Document</title>
</head>
<body>
<h1>This is a level 1 heading.</h1>
<h2>This is a level 2 heading.</h2>
<p>This is a plain paragraph with wrapping text. This is a
plain paragraph with wrapping text. This is a plain paragraph
with wrapping text. This is a plain paragraph with wrapping
text.</p>
<p>This is text with forced line breaks.<br />
This is text with forced line breaks.<br />
This is text with forced line breaks.<br />
</p>
</body>
</html>
```

A Detailed Explanation:

You will probably have some questions about the odd tag at the top of the document.

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

This XML declaration is not required but is recommended to make your documents compatible with future Web browsers that rely more heavily on XML (extensible markup language). This is too big of a topic to go into on this site, so see the [W3C](#) pages on XHTML for more information if you are interested. Another benefit of including the XML declaration is that it indicates the character encoding used in your document. Most documents made for the web today are served using the Western/Latin1 encoding which is known as iso-8859-1 as seen in the XML declaration above. If you intend to create pages in non-western languages like Japanese or Korean, however, you will have to use other encodings. Another common option for character encoding is to include the encoding in a meta tag. If you choose not to include the XML declaration in your documents, please do include the meta tag.

```
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
```

The next two lines of code refer to a document type definition (dtd). This piece of code lets the Web browser know what version of HTML it should use when rendering the page for your reader. Even if you omit the !DOCTYPE tag, most Web browsers do a good job of rendering your pages as long as you do a good job in coding them properly. That is why when coding in older versions of HTML using the !DOCTYPE tag was optional. Today when coding in XHTML you MUST use the !DOCTYPE tag. The !DOCTYPE tag for this page indicates that it was coded using strict XHTML 1.0. Most pages you create, however, will use the XHTML 1.0 transitional setting to allow you the flexibility to incorporate older tags from previous versions of HTML.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

The next tag pair is <html> </html>. These tags contain all the content on the page. This tag pair tells the browser that the text contained within <html> and </html> was created using hypertext markup language (HTML). The one attribute present in the <html> tag is added for XHTML documents to assist the browser in rendering them correctly.

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

Next you see the `<head> </head>` tags. These tags contain information about your document such as the title. Many things such as scripts, style sheets, and meta tags will go into the `<head> </head>` section of a typical HTML document but the most common one is the `<title> </title>` tag pair. The `<title> </title>` tag pair contains the text that appears in the title bar of the document's window.

The `<body> </body>` tag pair contains the content that will actually appear on your Web page. The first content in this example is a level 1 heading. XHTML provides six levels of headings with tags `<h1>` through `<h6>`. The example page contains a level 1 heading between the `<h1> </h1>` tags and a level 2 heading between the `<h2> </h2>` tags.

The `<p> </p>` tag pair contains a paragraph of content. The text within the tags will wrap around the window naturally. Open your example page in your Web browser and try resizing the window to see how the text wrap changes to accommodate the new window size.

The next `<p> </p>` tag pair contains three lines of text. Each line ends with the `
` tag. The `
` tag does not contain any content but is used to force a line break on the page. The `
` tag is unusual in XHTML because it is a self-closing tag. In older versions of HTML, such as HTML 4.01, the `
` tag is coded as `
` and no closing tag is required. In XHTML, however, you must always code line breaks as `
`. There are other exceptions to the containment principle. The `<hr />` tag produces a horizontal rule on the page. The `` tag displays an image on the page. These tags and a few more like them are also considered self-closing tags.

The Principle of Containment - content is contained by pairs of tags

Almost every tag in the example above has a corresponding closing tag. Many times it will be necessary to include one pair of tags within another pair of tags. This is known as nesting. When nesting tags, make sure you close each pair in the proper order as seen below.

Good Nesting

```
<p>Begin a paragraph then add some <b>bold text.</b></p>
```

Bad Nesting

```
<p>Begin a paragraph then add some <b>bold text.</p></b>
```

Appendix 6

CSS Basics

(excerpt from makepages.com, used here by permission)

This page contains a summary of CSS style rules. Use this page as a reference as you work through the remaining lessons. Read it once now and then come back to it as needed. It may be helpful to you to print this page for future reference.

Style Rules

The example below illustrates a typical style rule. The p represents the HTML element `<p></p>`. The style rule contains instructions to use the font-family arial at 90% size. If the visitor's browser can't find the arial font, it will use its default sans-serif font instead. A style rule consists of a **selector** and a set of one or more **declarations** surrounded by curly braces. In this example the p is a selector. A declaration is the combination of a **property** and its **values** . In this example, font-family and font-size are properties, and the font names and 90% are values. Use a colon between the property and its values. Use a semi-colon at the end of each declaration. The declaration is contained by the curly braces { and }.

```
p { font-family: arial, sans-serif; font-size: 90%; }
```

Selectors

Element selectors refer to HTML elements. You can create style rules for a single HTML tag such as the first rule below. You can also create style rules that apply to a group of HTML tags by separating them with commas as in the second rule below.

```
h1 { font-family: sans-serif; color: maroon; }
```

```
h1, h2, h3, h4, h5, h6 { font-family: sans-serif; color: maroon; }
```

Contextual selectors indicate the style of an HTML in a given context. The example below will set the color of bold text to green wherever the `` tag pair is contained within a `<h1></h1>` tag pair.

```
h1 b { color: green; }
```

Class selectors change the style of HTML content identified as belonging to a given class using the class attribute. The example below shows the style rule and

the HTML class attribute. Notice the selector begins with a period.

```
.huge { font-size: 300%; }  
  
<p class="huge">This will be really big text.</p>
```

ID selectors change the style of HTML content identified using the ID attribute. The example below shows the style rule and the HTML ID attribute as it would appear in an HTML document. Notice the selector begins with a hash mark.

```
#tiny { font-size: 50%; }  
  
<p id="tiny">This will be really small text.</p>
```

Pseudo-class selectors are available for the `<a>` tag only to reflect the different states of a link. Note the use of the colon within the selector.

```
a:link { color: blue; }  
  
a:visited { color: black; }  
  
a:active { color: green; }  
  
a:hover { color: red; }
```

Pseudo-element selectors are available to identify sub-parts of an element. The commonly used pseudo-elements are first-letter (for dropcaps, mostly) and first-line. The dropcap effect is usually created using a class selector as seen below. This allows you to create a drop cap effect on any paragraph just by changing its class attribute value.

```
p.dropcap:first-letter { font-size: 300%; float: left;  
color: red; }  
  
p:first-line { color: red; }
```

Choosing Values

You may choose values using keywords, length values, percentage values, colors, and URLs.

Keywords include values such as small, large, medium, normal, dotted, etc.

Length values are positive or negative numbers followed by a two-character abbreviation to indicate the unit of measurement. Absolute values can be stated in inches (in), centimeters (cm), points (pt, 72 points to an inch), or picas (pc, 1 pica is 12 points). Relative values may be stated in Em-height (em, the height of the character box for a given font), X-height (ex, the height of the lower-case x in a given font), or pixels (px). In practice, pt is the most consistently implemented absolute value and em is the best choice for a relative value. X-height is reliable only on IE5Mac.

Percentage values are often used for font-size and line-height. Percentage values are relative to the default values of the visitor's browser.

Colors may be given in hexadecimal or rgb notations or color names may be used.

The standard hex-pair notation gives red, green, and blue values for a color in three pairs of two-character hexadecimal numbers (#FFFFFF for white, #000000 for black, #FF0000 for red, etc.). The short form hexadecimal notation uses only the first character of each pair and assumes the second character to be the same (#F0F would create the same purple color as #FF00FF).

RGB notation may use percentage values or decimal values (0 to 255) for red, green, and blue. Both notations below would produce a purple color by combining red and blue with no green.

```
rgb (100%, 0%, 100%)
```

```
rgb (255, 0, 255)
```

Keywords exist for the 16 colors aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow.

URLs are often used to choose background images for a page. CSS defines urls as relative to the page containing the style sheet.

```
url( niftybackground.jpg )
```