

REVIEW LESSON

MTA Course: Web Development Fundamentals

Lesson name: Web Development Fundamentals 1.1 (One 50-minute class period)

Topic: Customize the layout and appearance of a Web page

File name: WebDevFund_RL_1.1

Lesson Objective:

1.1: Customize the layout and appearance of a Web page. *This objective may include but is not limited to:* CSS, tables, embedding images, page layout for navigation.

Preparation Details

Prerequisite student experiences and knowledge

This MTA Certification Exam review lesson is written for students who have learned about Web design and Web application programming. Students who do not have the prerequisite knowledge and experiences cited in the objective will find additional learning opportunities using resources such as those listed in the Microsoft® Resources and Web Links below.

Students should have experience with creating Web pages that contain both internal and external cascading style sheets, table layouts within Extensible Hypertext Markup Language (XHTML), embedding images and files in a website, storyboarding (which illustrates the site navigation).

Instructor preparation activities

For this lesson, you will need a computer with Microsoft Office 2007® attached to a liquid crystal display (LCD) projector to display and review the attached Microsoft PowerPoint® document.

Provide students with blank index cards. As they hear a new word or phrase, have them write the word and definition on each index cards. Place all the cards on a bulletin board to help compile a list of all the vocabulary. At the end of the lesson, the student with the most index cards wins a prize (mechanical pencil, etc.).

Resources, software, and additional files needed for this lesson

- Index cards
- Markers
- Office (preferably 2007)
- LCD projector
- Student prizes (mechanical pencil, a free homework pass, etc.)

Teaching Guide**Essential Vocabulary**

HTML—HyperText Markup Language. A common markup language that provides a means to structure documents into a format that Web browsers can interpret and display as Web pages.

XHTML—Extensible Hypertext Markup Language. XHTML is similar to HTML with the main difference being that XHTML documents must contain well-formed XML elements, while HTML documents are not held to this strict standard.

CSS—Cascading Style Sheets. A technology developed by the World Wide Web Consortium (W3C) to separate style from content in a HyperText Markup Language (HTML) page. A style sheet is a file that contains information about the color, font, and layout used on a website. Like a script file, an external style sheet file can be cached by the browser to reduce subsequent page download time.

table—Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the <td> tag). The letters *td* stand for "table data," which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, and other HTML tags and elements.

embed—The process of including an external image, document, or other object within a Web page.

image—An element used in Web development to include graphics on a Web page. Images are usually one of the following formats: Joint Photographic Experts Group (JPEG), Graphical Interchange Format (GIF), or Portable Network Graphics (PNG)

inline style—Inline styles are defined using the 'style' attribute within an HTML element.

internal style—Internal styles are used to define the styles of elements only within the Web page in which the internal styles are included. Internal styles must be defined within a <style> tag, which is placed in the <head> section of the HTML file...

external style—All the style definitions are saved in a separate text file and referenced by one or more of the pages on the website. The link tag that applies the external style sheet to the Web page goes inside the <head> section of the HTML code.

selector—The name of the HTML tag that is being set by this style, such as body {.

property—The attribute of the HTML tag that is set in the selector tag, such as color:.

value—The value of the attribute for the CSS style, such as yellow;.

Lesson Sequence

Activating prior knowledge/lesson staging (10 minutes)

1. Hand out index cards and explain the process of writing down keywords and definitions (make sure the students place their names on the back of the cards so you can see who submitted the most cards).
2. Have the students open your school's home page. Ask them to brainstorm on how the developer organized the various elements on the page. Did they use tables? Did they use cascading style sheets (CSS)? If so, what elements are included in the style sheet?

Lesson activity (30 minutes)

1. Using the PowerPoint presentation WebDevFund_PPT_1.1, walk the students through the concepts for this lesson (approximately 7 to 10 minutes).
2. Distribute Student Activity document WebDevFund_SA_1.1. Assign each student a topic for a one-page website, or allow them to choose a topic (for example: they can choose their favorite band, clothing store, school, or community organization; or you can assign a topic to each student).
3. This activity assesses student understanding at multiple cognitive levels, as follows:
 - Analyze and evaluate a given problem (design a website using CSS, embedded images, tables/layers, and navigation).
 - Develop a Web page using the guidelines in the assignment.
 - Test the final product to ensure that it views as expected in a Web browser.
4. Display the vocabulary terms with definitions on the index cards from the students (add any that are missing). Ask students if they have questions about the terms.

Assessment/lesson reflection (10 minutes)

1. Ask the students to view the same page from the beginning of class. Have them view the source for this page (by right-clicking the Web page and selecting 'view source', or by using the browser's menu to access the 'view source' option). What type of style elements are on the page? How would they change the layout of the page? What data is used to dynamically change the page?
2. Ask the students to reflect on the activity and provide areas that need additional explanation.
3. Wrap-up and provide homework/enrichment opportunities.

Microsoft resources and Web links

Beginning Developer Learning Center/CSS:

<http://msdn.microsoft.com/en-us/beginner/bb308768.aspx>

Microsoft ASP.NET:

<http://www.asp.net>

Additional activities (homework or enrichment):

1. Provide students with the attached matching activity. Encourage them to research any terms that they are not completely familiar with.
2. Have students research the usability and accessibility guidelines for Web design. Ask them to write up a one-paragraph summary of their findings.

Suggested best practices

Peer review and class presentations encourage collaboration and quality products.

Use the prompting questions from Bloom's taxonomy, such as:

- How can you use...?
- Why do you think...?
- Can you list the parts...?
- Can you make changes to solve...?
- Can you predict the outcome if ...?
- Can you identify the different parts...?

(More are available if you research Bloom's Taxonomy.)