

HTML



Microsoft Technology Associate Series
HTML5 Application
Development Fundamentals

COURSEWARE 8375-1

EXAM 98-375

Microsoft
Technology Associate

CCILearning™



Lesson 1:

Understanding Core Concepts

Lesson Objectives

In this lesson, you will learn about the requirements for creating a basic HTML web page including the DOCTYPE declaration, Character Encoding, and the basic structure of the html, head, and body tags. In addition, the correct syntax of elements and attributes are explored using the online W3Schools.com reference for HTML tags and attributes including new HTML5 tags and deprecated tags. Finally, you will learn about the W3C organization and the transition that is currently taking place from HTML 4.01 specification to HTML 5. JavaScript and CSS were also explained because the MTA 98-375 HTML5 Application Development Fundamentals exam focuses on web applications that are capable of running on touch-enabled screens using HTML5, JavaScript, and CSS. Upon completion of this lesson, you should be able to:

- create the basic structure and elements needed for an HTML5 page
 - understand how to use an online reference to find available tags and attributes
 - understand how to nest elements
 - understand that the MTA 98-375 exam will cover HTML5, JavaScript, and CSS
 - understand that there are many HTML editors and developer tools that can be used to develop HTML5 applications
 - create tags and attributes using the correct syntax
 - use the DOCTYPE declaration for HTML5
 - use a meta tag to specify the type of character encoding that will be used
 - use entities to display content that includes special characters
 - have a basic understanding of what HTML5, CSS, and JavaScript can be used for
-

Introduction to HTML5 Application Development

Objective 1.1 Understand the platform fundamentals

HTML and W3C

HTML5 is the latest standard for *Hypertext Markup Language (HTML)* that is used in web design and development. HTML is the markup language used to construct web pages and should not be confused with programming languages like Visual Basic, Java, C++, C# and many others. HTML by itself does not have the ability to perform complex calculations and make decisions based on user input. JavaScript has become the standard language used to accomplish complex calculations and make decisions but still uses HTML to display and format the input and output. The previous version of HTML, which is HTML 4.01, has been around since 1999 and was set as a standard by an international community called the *World Wide Web Consortium* or *W3C*.

Although HTML5 is HTML, it has been developed by the W3C to work directly with CSS3 and JavaScript to allow developers and designers the ability to create web applications that will also work on touch-enabled devices and easily adapt to smaller screens. In addition HTML5, CSS3 and JavaScript are all considered *platform-independent* technologies and therefore can run regardless of the operating system as long as the operating system has a web browser. For example, Windows 8 and Windows Phone 8 have the Internet Explorer browser, Mac OS and iPhone have the Safari browser, Linux OS has the Firefox browser and Android phones have the Google Chrome browser. Each of these browsers is capable of running HTML5, CSS3, and JavaScript. However, browsers also have versions, so as long as the browser being used is the newest version then most HTML5 and CSS3 features will be supported. The reason that some features are not supported is due to the fact that the W3C has not completely finalized HTML5 specifications but plans to release a stable HTML5 version by the end of 2014. If you would like to know how well your browser supports HTML5 you can visit html5test.com which gives a score and shows supported and unsupported features. The html5readiness.com website has a visual representation of supported HTML5 and CSS3 features by browser and also shows how the browsers have adapted each year. Finally, the caniuse.com website provides a matrix that includes almost all desktop and mobile browsers and can be filtered by browser type or category and also shows past, current and future browser support by browser version.



Figure 1.1: The image shown is the HTML5 logo that was created by the W3C as the new brand to visually identify HTML5. This screen shot was obtained from the W3C website at <http://www.w3.org/html/logo/> which contains useful information about HTML5 including frequently asked questions. *This work is attributed to the W3C.*

CSS and CSS3

Cascading Style Sheets (CSS) is used with HTML to separate style from content and currently the W3C latest standard is CSS3. The W3C created the CSS standard so that styling could be separated from content and therefore would allow designers and developers the ability to modify styles across the entire site in a much more efficient way. For example CSS rules are created that allow elements of the same type, name, or class to be styled as needed across several web pages at once. CSS is used for many things some of which are styling text fonts and colors, styling content, and layout out of pages. The new CSS3 standard has added the ability to adapt and readjust layouts from a desktop screen to a much smaller mobile interface without having to recode or perform many complex calculations to adjust the layout. In addition CSS3, has added several new modules for working with text effects, transformations, animations, rounded corners, gradients, borders, backgrounds, images and more.

Just like HTML5, CSS3 is still under development by the W3C but most current browsers still support CSS3 features. The CSS3 standard is also backwards-compatible with earlier versions of CSS so there is no need to remove older CSS code from existing sites.

JavaScript Versus Java

As stated above JavaScript is the language that is used in coordination with HTML5 to produce web pages that can perform complex calculations, make decisions based on conditional statements, and respond to events and user input. The W3C chose JavaScript because it is a language that is platform-independent and can execute as long as there is a web browser. So you might be asking yourself "What is the difference between JavaScript and Java?" Java is a full Object Oriented Programming language that must be compiled into machine language with a development tool like Visual Studio before it can execute. JavaScript was created based on the Java language, but is primarily intended to run or execute from within the client's web browser and therefore JavaScript is a much smaller and less complex programming language. In addition, the JavaScript code that is executed by a browser is considered *client-side processing* which differs from other web based programming languages like PHP and ASP.NET that are considered *server-side processing* and require a web server to execute.

HTML Tags and Elements

Objective 1.1 Understand the platform fundamentals

Basic Tag Syntax

HTML is a markup language that uses markup tags to describe the type of content that will be used in the construction of a web page which is often referred to as an HTML document. HTML markup tags are often referred to as just HTML tags and almost always are used in pairs meaning that there will be a start tag and end tag. The tag name is enclosed in angle brackets and the **start tag** differs from the **end tag** in that the end tag also includes a forward slash immediately before the tag name. For example, take a look at the sample code below that shows the basic tag syntax and syntax for the paragraph tag element.

Basic Tag Syntax:

```
<tagname>  
    Content Goes Here!  
</tagname>
```

Paragraph Tag Syntax:

```
<p>  
    This is where the Paragraph text is placed  
</p>
```

The basic tag syntax that is used consists of start tags and end tags like the example above and the tag name that is enclosed in angle brackets must be the same in the start and end tags. The tag name is not case sensitive and may be either upper or lower case, but most developers and editors will use all lower case letters for tag names. Although most tags will have start and end tags there are some tags that only require a single tag called *empty tags*. For example, the break tag is used to move content down the HTML document one line. Since the break tag is just used to move the content down one line there is no need for an end tag because we do not have to put content inside of the break tag as it will just move any content following the break tag to the next line in the HTML document.

Empty Tag Code:

```
Some Content on the Page <br /> This content will appear on the next Line
```

How the code above will appear on the web page:

Some Content on the Page
This content will appear on the next Line

Notice that the empty tag consists of the tag name wrapped in angle brackets, but the tag name is also followed by a space and forward slash indicating that the empty tag will not have an end tag. The `<hr>` tag called the horizontal row and the `` image tag are examples of two other empty tags that are commonly used in HTML markup.

HTML Elements and Nesting

HTML elements are often used to describe the same thing as HTML tags, but an HTML element also includes the content that is within the start and end tags. In the case of single or empty tags, the single tag would also be considered the HTML element. Throughout this book the words element and tag will be used with the same meaning, but remember that the use of the word tag could imply just a start, end, or empty tag and an HTML element includes both the start tag, the end tag, and the content within the two. The example code below shows the paragraph element, heading element, break element, and horizontal row element.

Elements with start and end tags:

Paragraph Element: `<p>`
`This Content is also part of the Paragraph Element`
`</p>`

Heading Element: `<h1>`
`This Content is also part of the Heading Element`
`</h1>`

Empty Tag Elements:

Break Element: `
`
Horizontal Row Element: `<hr />`

HTML tags can also be nested inside of each other. In the example code below you will see that there is a paragraph that contains text and the word Content is bold and the word Paragraph has been italicized.

Nesting Elements:

```
<p>This <b>Content</b> is also part of the <i>Paragraph</i> Element</p>
```

When nesting tags you must close the inner tags before closing the outer tags that will create a sandwich effect with the tags and their content. Remember that the space between tags, indents, or breaking down to the next line will not affect the display of your web page so therefore when nesting tags you should put inner nested tags on the next line and indent them so that it is visually easier to read the code. For example, the following two sections of code will produce the same results in the browser, but as you can see the spaced and indented code is much easier to read. Spaces and indents in your markup also make it easier to tell which elements are nested inside of other elements allowing the developer to identify if tags were properly nested and closed in the correct order.

Nesting without spaces or indents:

```
<!DOCTYPE html><html><head><title> Title in Browser Tab </title></head><body> This will be the text displayed on the web page! </body> </html>
```


Nesting with spaces and indents:

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>  
      Title in Browser Tab  
    </title>  
  </head>  
  <body>  
    This will be the text displayed on the web page!  
  </body>  
</html>
```

Deprecated Elements

As new standards are released from the W3C there are also elements that will no longer be supported by the W3C standards and therefore browsers will eventually stop displaying *deprecated elements*. For example, the W3C has deprecated the tag that was used for coloring and styling fonts, because they recommend the styling of text should be done using CSS. Although the W3C has deprecated an element it doesn't necessarily mean that the element will not display by the browser. Many current browsers still recognize deprecated tags, but eventually they will not display and therefore it is recommended that you no longer use deprecated tags in your markup. The process of a browser reading and displaying an HTML tag is called *rendering*. Since there are so many browser types you can imagine that some browsers may render or display the HTML markup in a different way and is based on how the development team interpreted the HTML markup when creating the browser software.

Exercise 1-1: Explore the HTML Tag Reference Online

1. Open a web browser on your computer and type in the following URL to your address bar.
`http://www.w3schools.com/tags/default.asp`
2. Notice that the HTML Reference is also labeled HTML5 Compliant and therefore you know that you are looking at a current and updated list.
3. Scroll down the list and notice that all tags and tag descriptions in red are labeled as "Not supported in HTML5. Use ... instead". These tags are considered deprecated tags.
4. Also notice that the HTML5 logo  is next to the tag name indicating that the tag is new to HTML5 and therefore you should check for support of that tag to ensure that the browsers that you recommend to use for your site currently support that tag.
5. Leave your browser window open for the next exercise.

Modifying an HTML Element Using Attributes

Objective 1.1 Understand the platform fundamentals

Attributes and Attribute Syntax

HTML elements have *attributes* that allow the developer to specify additional information about the element. Attributes must be placed in the start tag with a space following the tag name and each attribute name should be immediately followed by an equals and the value of the attribute wrapped in double quotes like the example syntax below.

Attribute Syntax

```
<tagname attribute_name="value of attribute">  
Content  
</tagname>
```

Each HTML element has a list of attributes that can only be used by that specific element, but there is also a list of *global attributes* that apply to all HTML elements. Just like HTML tags, HTML attributes and their values are not case sensitive but the W3C recommends using lowercase for all HTML tags, attributes, and their values. Look at the paragraph below and notice that the attribute named `id` is placed in the paragraph start tag followed by a space. Also immediately following the attribute is the equals sign with the value `top_paragraph` wrapped in quotes.

```
<p id="top_paragraph">This is the content of the paragraph</p>
```

Common Global Attributes	
HTML Attributes	Description
id	Specifies a unique identifier that the element will be referenced by in JavaScript, CSS, and the HTML document. Note: New in HTML5, all elements can have an <code>id</code> attribute.
class	Specifies a class that the element will be referenced by in JavaScript, CSS, and the HTML document. The class attribute is similar to a group that the element belongs to and multiple elements can be assigned to the same class. Classes are most often used to style several elements by assigning the style properties once to the class rather than multiple times if you were to assign the style to the <code>id</code> attribute.
style	The style attribute is a way of embedding CSS style properties inside of the HTML element. This is most often done to override styles that are set globally to the HTML document or to a class that the element belongs to.

New HTML5 Global Attributes	
HTML5 Attributes	Description
contenteditable	The <code>contenteditable</code> attribute specifies if the element is editable directly by the user through the browser. This attribute accepts the two values of <code>true</code> and <code>false</code> . For example, if you set the <code>contenteditable</code> attribute to <code>true</code> for the paragraph element then the content in the paragraph will be made editable by the user.
spellcheck	The <code>spellcheck</code> attribute specifies if the element can be checked by the browser for spelling and grammar. The <code>spellcheck</code> attribute can only be set in the text value of input tags except passwords, the content text of a <code><textarea></code> element, and elements like a paragraph that have the <code>contenteditable</code> attribute set to <code>true</code> .
hidden	The <code>hidden</code> attribute can be used to remove an element from displaying until the <code>hidden</code> attribute is taken away from the element through JavaScript.
draggable	The <code>draggable</code> attribute is used to specify if the element can be dragged for the purpose of drag and drop. This attribute accepts the three values of <code>true</code> , <code>false</code> , and <code>auto</code> . If set to <code>auto</code> the property will be set to the browsers default setting.

Exercise 1.2: Explore the HTML Attributes & Global Attributes

1. If your web browser is not still open to the W3Schools.com tag reference from Exercise 1.1, then open a web browser on your computer and type the following URL into your address bar and navigate to the W3Schools website HTML tag reference page.

<http://www.w3schools.com/tags/default.asp>

2. Scroll down the alphabetical list and find the tag that is used to define an image. Click the tag name and you will be taken to the image tag reference page.
3. Scroll down the image tag reference page to the Attributes section where you will find a table of attributes that are used with the attribute name, potential values, and a description of the attribute.

Notice in the description for the align attribute for the image tag the first sentence of text explains that the align attribute is not supported in HTML5 and that it was deprecated in the HTML 4.01 standard.

4. If you would like to learn more about a specific attribute or see a working example of the tag and the attribute being used click on the attribute name and a page will load showing additional information for the specific attribute that was clicked on.
5. After you are finished exploring attributes use your web browser to navigate to the following link to explore a list of global attributes that can be used with any HTML element.

http://www.w3schools.com/tags/ref_standardattributes.asp

Once the page loads you should see the heading near the top of the page indicating that you are on the HTML Global Attributes page.


HTML Global Attributes

[« Previous](#)


HTML attributes give elements meaning and context.

The global attributes below can be used on **any** HTML element.

HTML Global Attributes

 = Attribute added in HTML5.

6. Scroll down the page and you will find a table with a list of Global Attributes.

Notice that there are no global attributes that have been deprecated, but there are several new global attributes with the HTML5 logo  by the attribute name indicating that the attribute is new to the HTML5 specification.

Just like the element attributes table notice that you can click on the attribute name if you would like to learn more about a specific attribute or see a working example of the attribute.

7. Close the web browser.

Required Tags for All Pages

Objective 1.1 Understand the platform fundamentals

The DOCTYPE

The `<!DOCTYPE>` declaration tag is used by the browser to identify what type of document specifications the browser should use to render the HTML markup. This is similar to how operating systems use file extensions on a file name to identify what kind of program will be used to open the file. Compare the HTML 4.01 declarations to the HTML5 declaration below and notice that the HTML5 declaration is much simpler.

HTML5 Declaration	HTML 4.01 Declaration
<code><!DOCTYPE html></code>	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"></code>

Character Encoding and Entities

The *character encoding* for an HTML document specifies the standard used to define special characters and how the browser should render and interpret those characters. The character encoding is set in the `<meta>` tag which should be nested inside of the head tag.

HTML5 Character Encoding	HTML 4 Character Encoding
<code><meta charset="UTF-8"></code>	<code><meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1"></code>

There are certain characters that have special meaning in HTML markup that do not allow you to put them directly in the HTML markup. For example, if you use the `>` greater than or `<` less than characters in your HTML content the browser may have a hard time displaying that character since the greater than and less than signs are used in start and end tags. *Entities* are used to provide an alternate way to display HTML characters. The following table contains some of the most commonly used entities. The entity name or the entity number can be used in your HTML markup to display the characters that are in the Result column of the table below.

Commonly Used Entities			
Result	Description	Entity Name	Entity Number
	non-breaking space	<code>&nbsp;</code>	<code>&#160;</code>
<code><</code>	less than	<code>&lt;</code>	<code>&#60;</code>
<code>></code>	greater than	<code>&gt;</code>	<code>&#62;</code>
<code>&</code>	ampersand	<code>&amp;</code>	<code>&#38;</code>
¢	cent	<code>&cent;</code>	<code>&#162;</code>
©	copyright	<code>&copy;</code>	<code>&#169;</code>
®	registered trademark	<code>&reg;</code>	<code>&#174;</code>

By specifying the type of Character Encoding that the HTML document is using you will help to ensure that the browser displaying your page will display entities correctly.

Creating a Page with Required Tags

When creating a new web page most editors will automatically generate the required tags for an HTML document so let's take a look at them now. The code below will display in the browser an empty HTML5 web page with only a title displaying. Notice that the HTML5 DOCTYPE declaration is in the first line of markup followed by the HTML start tag. Inside of the html start and end tags is the head element and the body element. All elements and content inside of the body will display in the browser, but content in the head element will not be used to display content on the page. The head element is used for the meta tag to provide a reference to the Character Encoding that will be used, the title that will be displayed in the browser tab, links to JavaScript files or CSS files, and much more. You will see just how important the head element is in the upcoming lessons.

```
1 <!DOCTYPE html>
2
3 <html>
4
5   <head>
6     <meta charset="UTF-8">
7     <title>Title that Displays on the browser Tab</title>
8   </head>
9
10  <body>
11  </body>
12
13 </html>
14
```

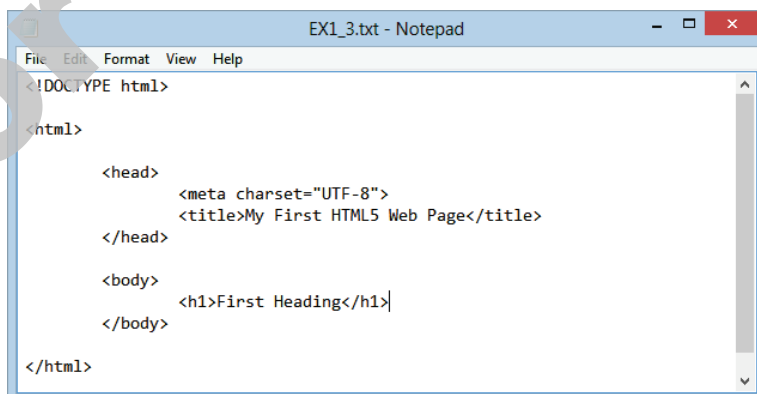
Figure 1.2: Basic Empty HTML5 document

Exercise 1.3: Creating your First HTML5 Web Page

This exercise requires that you have downloaded the student data files for this courseware and extracted to a folder on the desktop. Check with your instructor if they are in a different location. Alternatively, please refer to the Preface section of this book for instructions on how to download these files.

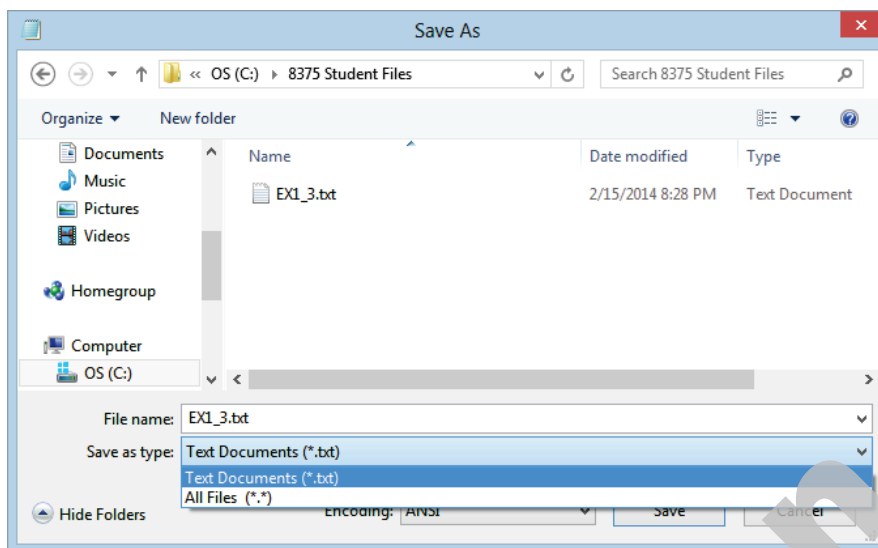
1. Navigate to the location of the Student Files folder and locate the *Ex1_3.txt* student exercise file and open it with Notepad.
2. Select and delete the words between the start and end <title> tag.
3. Type: My First HTML5 Web Page between the start and end <title> tags.
4. Place your cursor at the end of the start body tag and press ENTER to move the cursor down one line.
5. Press the TAB key to indent one tab space from the body tags and type: <h1>First Heading</h1>.

Compare your syntax to the markup below and make sure that they are the same.



```
EX1_3.txt - Notepad
File Edit Format View Help
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>My First HTML5 Web Page</title>
  </head>
  <body>
    <h1>First Heading</h1>
  </body>
</html>
```

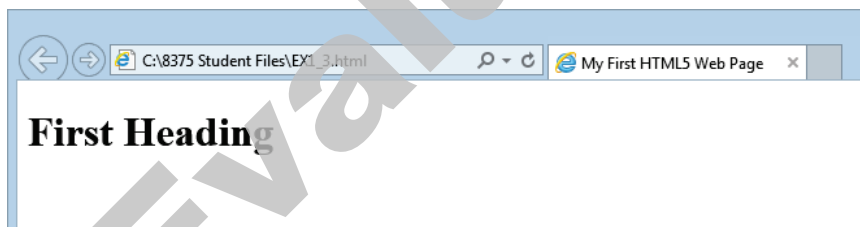
- On the File menu, click **Save As**.
- In the Save As dialog box change the Save as type to **All Files** as shown below.



- Now add the file extension `.html` to the File name. The File name should now be `EX1_3.html`.

Note: If file extensions are visible, replace the `.txt` file extension with `.html`.

- Click the **Save** button.
- Using Explorer or My Computer, navigate to location of your student files and locate the `EX1_3.html` file.
Notice that the new HTML5 file has your computers default web browser icon showing on the HTML file indicating that it will open with your default browser.
- Double click the html file and it will open with your default web browser.



Notice that on the page you can see the heading 1 text, and in the top of the window you can see the title "My First HTML5 Web Page" as the tab name.

- Close the web browser and Notepad.

Developer Tools and HTML Editors

Objective 1.1 Understand the platform fundamentals

In this section you will be learning about some of the developer tools that can be used for HTML editing and development. This publication will not be endorsing one over another, but suggest that you explore using a variety of editors to see which one you like the best. Some of the tools that we will cover are free and some are not so you should also evaluate which tool works best for your budget. Take note that some vendors will offer special pricing for students that would like to purchase their development tool at a discounted price.

Notepad and Notepad++

Notepad is a very basic text editing program that comes as part of the default installation of Windows operating systems, and therefore is almost always available if you need to quickly update your HTML markup. Since Notepad and all basic text editing programs do not have any benefits to help improve the coding process it is rarely used by web developers. Notepad++ is a free text editor designed to help improve the coding process and is publicly distributed under the GNU General Public License. Some of the features of Notepad++ include line numbers for debugging, syntax highlighting and folding, multi-document tab view to work with several files at once, and column mode editing that allows you to edit multiple lines of code at once. Many developers choose to use Notepad++ because it's free and also is considered platform independent meaning that it is not tailored to one platform such as Windows, Mac or Linux. Figure 1.3 below shows the Notepad++ interface. To download Notepad++, or to learn more about it, visit their website at notepad-plus-plus.org.

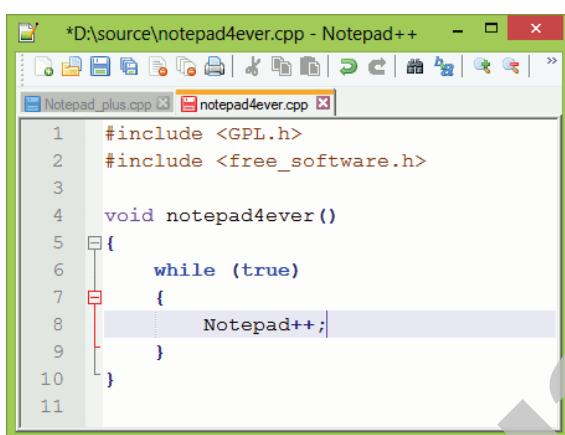


Figure 1.3: The Notepad ++ interface. Image obtained from <http://notepad-plus-plus.org>

Microsoft Visual Studio and DreamSpark

Visual Studio is the all in one developer tool that can be used with teams of developers that are collaborating on a variety of development projects. Visual Studio is not limited to web and HTML5 development, but some of the projects include .Net Windows software, ASP.NET Web applications, HTML5 touch enabled applications and more. Visual Studio also integrates with many other Microsoft products including Microsoft's SQL Server and allows the developer to interact with Microsoft SQL Server without having to use another software development tool. Microsoft DreamSpark is not an HTML editor, it's a program offered by Microsoft that supports technical education by providing access to Microsoft software for learning, teaching and research purposes. As a student, you can get access to Visual Studio and other Microsoft software packages for free. Although DreamSpark is free for students, schools that would like to gain access to additional Microsoft software packages that are above and beyond the free student access can pay a small subscription fee giving teachers and students access to the additional software. To sign-up or learn more about DreamSpark visit <http://www.dreamspark.com>.

Visual Studio Express 2013 for Web

Visual Studio Express 2013 for Web is a free version of Visual Studio that is designed to offer many of the same features that are included in Visual Studio but are designed specifically for web developers and is available for anyone to download. There are many features that we could look at with the express versions of Visual Studio, but some of the key features to note that help improve and speed up the development process is intellisense and auto-completion. For example, when creating an element that has a start and end tag the auto-completion feature will automatically create an end tag after creating the start tag. This helps speed up the process when creating HTML markup and also helps to avoid syntax errors. In addition, the intellisense feature will automatically offer attributes and their values when you are inside of a start tag and hit the space bar. The intellisense feature also helps improve the speed of development and helps reduce syntax errors.

Visual Studio Express also has a built in WYSIWYG designer that allows the developer to see and work with the visual environment rather than just with the code. Note that developers should be careful using WYSIWYG designers because they will create code based on what you change or move in the designer and often will not produce the code that you expect. However a great feature that is used with the designer is to help locate elements and locations in your HTML markup. For example you may have an HTML document with 200 lines of markup and therefore it's somewhat hard to find elements within those lines of markup, but the WYSIWYG designer will allow you to click on the element or content in the designer and then when you change to the code view your cursor will be placed at the exact location in the HTML markup designer. To download or learn more about Visual Studio Web Developer Express visit <http://www.asp.net/vwd>.

Exercise 1.4: Explore Visual Studio Express 2013 for Web

1. Using your web browser navigate to the following website:
<http://www.asp.net/visual-studio/overview/2013/>
2. Click the arrow to the right of Visual Studio 2013 in the left pane, then click on the first video in the list titled *Visual Studio 2013 Web Editor Features - Browser Link*. This will take you to the following URL:
<http://www.asp.net/visual-studio/overview/2013/visual-studio-2013-web-editor-features-browser-link>
3. View the following videos that are listed on the left side of the page as in the screenshot below.
 - Visual Studio 2013 Web Editor Features – Browser Link
 - Visual Studio 2013 Web Editor Features – CSS
 - Visual Studio 2013 Web Editor Features – JavaScript
 - Visual Studio 2013 Web Editor Features – Publishing
 - Visual Studio 2013 Web Editor Features – Page Inspector
 - Visual Studio 2013 Web Editor Features – HTML Editor
 - Visual Studio 2013 Web Editor Features – HTML5

Microsoft Search ASP.NET

ASP.NET Home Get Started Learn Hosting Downloads Community Forums Help

LearnNowOnline Online video tutorials for ASP.NET, MVC, SignalR and more. Try Free! Try Free

Visual Studio 2013

- Visual Studio 2013 Web Editor Features - Browser Link
- Visual Studio 2013 Web Editor Features - CSS
- Visual Studio 2013 Web Editor Features - JavaScript
- Visual Studio 2013 Web Editor Features - Publishing
- Visual Studio 2013 Web Editor Features - Page Inspector
- Visual Studio 2013 Web Editor Features - HTML Editor
- Visual Studio 2013 Web Editor Features -

Visual Studio 2013 Web Editor Features - Browser Link

By Scott Hanselman | 3 minutes | October 17, 2013

Like 55 Tweet 30 Print

3 minutes, 34 seconds

Scott Hanselman demonstrates Browser Link in Visual Studio 2013, which allows for two-way communication between Visual Studio and multiple connected browsers.

Note: Step 4 requires administrative permissions on the Windows Operating System and therefore most likely will not work if you are at school or any other managed network.

- If you would like to download and install Visual Studio Express 2013 for Web go to the Microsoft Download Center using the URL below. Download the software and read the System Requirements before starting the installation. Finally use the Install Instructions to install Visual Studio Express 2013 for Web.

<http://www.microsoft.com/en-us/download/details.aspx?id=40747>

Note: Alternatively, if the above URL does not take you to the download page for VS Express 2013 for Web then visit the Microsoft Download Center and search for "Microsoft Visual Studio Express 2013 for Web".

<http://www.microsoft.com/en-us/download>

Adobe Dreamweaver

Although Visual Studio and Visual Studio Express 2013 for Web are great development tools, some developers feel that they are tailored towards Microsoft technologies and therefore they choose tools such as Adobe Dreamweaver and Notepad++ as their developer tool of choice. Adobe Dreamweaver is a developer's tool that has many great features that are similar to Visual Studio Express 2013 for Web but Adobe Dreamweaver is not free. To learn more about Adobe Dreamweaver visit

<http://www.adobe.com/products/dreamweaver.html>.

Lesson Summary

In this lesson, you learned the requirements for creating a basic HTML web page including the DOCTYPE declaration, Character Encoding, and the basic structure of the html, head, and body tags. In addition, the correct syntax of elements and attributes were covered and explored using the online W3Schools.com reference for HTML tags and attributes including new HTML5 tags and deprecated tags. Finally, you learned about the W3C organization and the transition that is currently taking place from HTML 4.01 specification to HTML5. JavaScript and CSS were also explained because the MTA 98-375 HTML5 Application Development Fundamentals exam focuses on web applications that are capable of running on touch-enabled screens using HTML5, JavaScript, and CSS. Upon completion of this lesson, you are now able to:

- create the basic structure and elements needed for an HTML5 page
- understand how to use an online reference to find available tags and attributes
- understand how to nest elements
- understand that the MTA 98-375 exam will cover HTML5, JavaScript, and CSS
- understand that there are many HTML editors and developer tools that can be used to develop HTML5 applications
- create tags and attributes using the correct syntax
- use the DOCTYPE declaration for HTML5
- use a meta tag to specify the type of character encoding that will be used
- use entities to display content that includes special characters
- have a basic understanding of what HTML5, CSS, and JavaScript can be used for

Review Questions

1. The MTA 98-375 HTML5 Application Development Fundamentals exam covers HTML5, CSS, and _____ scripting language.
2. The _____ organization is responsible for creating the HTML5 specifications.
3. HTML tags and elements are often referred to as the same thing but the element includes the start and end tags as well as all content between the two tags.
 - a. True
 - b. False
4. _____ attributes are available to all HTML elements.
5. Attributes must be placed in the _____ tag of an element.
6. Tags that have been deprecated will not display in the web browser.
 - a. True
 - b. False
7. Which technology is used to style the content of a web page?
 - a. HTML5
 - b. JavaScript
 - c. CSS
 - d. PHP

MMM

Go online for Additional
Review and Activities

8. Which technology is used to perform calculations and handle conditional statements?
 - a. HTML5
 - b. JavaScript
 - c. CSS
 - d. XHTML
9. By declaring the HTML5 DOCTYPE the web browser will only render HTML5 tags and will not render any HTML 4.01 or earlier tags.
 - a. True
 - b. False
10. HTML elements, attributes, and their values are all case sensitive?
 - a. True
 - b. False
11. A simple text editor like Notepad can be used to develop HTML5 touch-enabled web applications with HTML5, JavaScript, and CSS3?
 - a. True
 - b. False
12. Character Encoding is specified in the meta tag with the _____ attribute?
13. If you would like to include special characters that the browser can't display in the page content you must use _____ specified by name or number in the HTML markup?
14. Adding spaces, indents, and line breaks to your HTML markup is recommended so that you can visually read and understand the markup much easier?
 - a. True
 - b. False
15. Which web programming language will NOT process server-side on the web server, but will process in the web browser client-side?
 - a. ASP.NET
 - b. JavaScript
 - c. PHP
 - d. Cold Fusion