Front End Web Development, Skill Labs

Course Specifications

Course Number: ACI76-039SL_rev1.0
Lab Length: Approximately 10 hours

Website Development Basics

Introduction
Overview

As a user of the Internet, you use a web browser to access information. There are many type of web browsers. Two examples are Google Chrome and Microsoft Edge. To visit a web site, you type in an Uniform Resource Locator (URL) into the Address bar of the web browser and hit enter. A web site loads into the browser for you to view. This web site also known as a web page uses a special file type called Hypertext Markup Language (HTML). HTML is the markup language translated by the browser so you can view it. A web page consists of different elements such as a HTML page, cascade style sheet (CSS), javascript files, images, and other types of files that make up a web page. There are really two types of web pages - static and dynamic. If the URL has an .html or .htm then it is a static web site. If the URL has items like .php, .jsp, or other file extensions, then it is a dynamic web page. These URLs are dynamic web pages which means when you request the URL, the HTML pages gets generated on the fly and then sent to the web browser for you to view. You will focus in these labs on creating a static web site.

In web site development, you usually have two parts of the web application - the front end and the back end. The front end is the user's interface to the web application under development. The back end usually consists of web developers who create the dynamic web pages using a special web scripting language. The back end is beyond the scope of these labs but it is an important part of create web sites today.

Development usually takes place on a local machine, such as a computer. Once developed, a domain name is obtained and the files are uploaded to a hosting account in order to be published for the public to view. Once a website is live, users use a specific address to view the live website. The developer must understand domain, server, and hosting concepts to ensure websites are published correctly and visible on the Web.

Outcomes

In this lab, you will learn to:

- Understand file management and hierarchy for website development.
- Identify the components of a website address, known as a Uniform Resource Locator (URL).
- Learn to prepare website components to be uploaded to website host on the World Wide Web.

	Key Term	Description
1	Uniform Resource Locator (URL)	A website address composed of a protocol, domain name, and path.
2	Domain name	A name assigned to a website used in conjunction with an IP address to locate a specific website.

	Key Term	Description
3	Internet Protocol (IP) address	Unique number assigned to a website or a device that is connected to a network.
4	Root directory	The main, or base, folder used to store HTML files and subfolders of a website.
5	File extension	Used to identify a file type; designated with a period followed by a string of letters.
6	Website hosting provider	Service that provides online storage of files and folders associated with a website.
7	ICANN	Internet Corporation for Assigned Names and Numbers, an organization responsible for managing and maintaining domain names.

HTML5 Basics I

Introduction
Objective

- Learn HTML5 syntax.
- Code the basic structure, semantic, and text support elements for a website page.
- Save a website file.

Overview

The first step to coding a website page is to declare the doctype and build the website page structure. Structure tags are all required and must be coded in a specific order. Each structure tag has a specific purpose in the file, but none actually display any content. Once the doctype declaration is made and the website page's structure is coded, content is created for the website page. All website content must be supported with valid and appropriate HTML tags.

	Key Term	Description
1	HTML	Hypertext Markup Language; the language used to structure and display content in a Web browser on the World Wide Web.
2	HTML5	The current version of HTML used in the website development industry.
3	W3C	World Wide Web Consortium; the organization that oversees standardization of the Web to ensure compatibility across all browsers.
4	Doctype declaration	Informs the Web browser what version of HTML will be used in the website page. For HTML5, the declaration is <!doctype html>.
5	HTML tags	Keywords written to define how content is displayed in a website. Most are written in pairs: opening and closing. Tags names are coded between < and > brackets.
6	Metadata	Data that describes data. It does not display in the browser window. Instead, metadata defines the document title, scripts, and other information about the document.

HTML5 Basics II

Introduction
Overview

Lists are used on a website page to display text into visually interesting and organized formats. Lists are also often used to create navigation bars on a website. You will code various list types, learn to code hyperlinks to outside websites, as well as internal links within a website to form navigation. HTML

tables are another method of organizing content on a website page. When used to display small amounts of tabular data, tables can be an effective design element.

Outcomes

In this lab, you will learn to:

1. Code various types of lists in HTML.
2. Code internal and external hyperlinks.
3. Code a basic HTML table.

	Key Term	Description
1	Unordered List	A vertical stack of text items. A bullet point displays in front of each list item. Unordered lists are used when the order of the list items is not relevant.
2	Ordered List	A numbered or alphabetical list of text items. A numeric or alpha bullet displays in front of each list item. An ordered list is typically used when the order of the items has relevance.
3	Hyperlink	Text that opens a different website file when selected; may link to a page within the same website or a page within a different website.
4	Absolute URL	The full URL of a website, including the protocol and full domain name. Absolute URLs are needed when linking to a file on a different server than the file that contains the link.
5	Relative URL	A file path or page name within a website. Relative URLs are portable and efficient but can only be used to point to files on the same server as the file that contains the link.
6	Table	Displays data in a grid format. Made up of columns (vertical) and rows (horizontal). The intersection of a column and row is called a cell.
7	Attribute	Specified within an HTML tag to define specific properties of the HTML element. Attributes are not included in closing tags.
8	Navigation Bar	Collection of hyperlinks that point to various files within a website.

HTML5 Basics III

Introduction
Overview

HTML is used to add images and interactive forms to website pages. Both images and forms are important elements to understand as they are needed within most websites. Images are embedded into the website page using the <img> tag and referenced using a relative file path. Forms are coded similar to tables with a single <form> and </form> tag pair that contains all the form elements.

Outcomes

In this lab, you will learn to:

1. Code an image onto a website page.
2. Code a Web form onto a website page.
3. Create a variety of form elements within a website form.
   z+-

	Key Term	Description
1	Image	A graphic or photo. Valid file types for a website image include GIF, JPG, and PNG

	Key Term	Description
		files.
2	Alternate Text	The text that displays on the page or is read to the user if the image doesn't display in the browser window or the user cannot see the image.
3	Interactivity	Website elements respond when a user executes an action. This can include clickable elements such as links and buttons as well as elements that accept text typed by the user.
4	Web Form	A website page element that gathers data from the user in the form of typed data and selected options. Web form data is typically sent to a database hosted on the server.

CSS3 Basics I

Introduction
Overview

Cascading Style Sheets (CSS) are used in combination with HTML documents to develop attractive and accessible websites. Whereas HTML is used to define content on a website, CSS is used to style the content. CSS can add font styling, colors, positioning, and even simple animations to website page elements for more appealing design. All HTML elements can be styled with CSS styles.

Outcomes

In this lab, you will learn to:

1. Understand how CSS applies to HTML content.
2. Use CSS syntax.
3. Add styling to website page content using external CSS.

	Key Term	Description
1	Cascading Style Sheet (CSS)	A styling language used to describe how HTML content displays in a website browser.
2	External CSS	A file containing only CSS syntax and linked to the document(s) containing the elements to be styled.
3	Style Rule	Made up of three parts: selector, property, and value. Combined, these parts apply styling to website page elements.
4	Selector	Portion of a CSS style that identifies the element that is to be styled. Usually a tag name.
5	Property	Portion of a CSS style that declares what will be styled.
6	Value	Portion of a CSS style that determines how a property will be applied.
7	Hex Color Code	A six-character combination of letters and numbers defined by various quantities of red, green, and blue (RGB) to display specific colors on a device screen. Designated with a # symbol in front of the six characters (e.g., #1a5cff).

CSS3 Basics II

Introduction
Overview

While external stylesheets are used to style all the files within a website, we can use internal CSS to make adjustments to a single page within the website. Inline (embedded) CSS is used to make changes at the element level. Custom styles can be written using IDs and classes in place of HTML element (tag) names to further customize the styling of the website. All forms of CSS work together to allow the developer control over the styling of the website as a whole.

Outcomes

In this lab, you will learn to:

1. Apply internal CSS to an HTML document.
2. Apply inline (embedded) CSS to elements within an HTML document.
3. Write an ID and apply it to an element within an HTML document.
4. Write a class and apply it to one or more elements within an HTML document.

	Key Term	Description
1	Internal CSS	Coded in the head of the HTML document. Applies to the applied file only. Overrides any matching external styles.
2	Inline (embedded) CSS	Coded within the HTML tags and applies to that element only. Overrides any matching internal and external styles.
3	CSS ID	Assigns styles to a specific element based on the id attribute applied to the element. Can be used one time only within an HTML document. Name of ID has a # symbol in front of it in the CSS.
4	CSS Class	Coded with a period (full stop ‘‘.’’) in front of the class name. Styles within the class are applied to HTML elements with the class attribute.

Building a website

Introduction
Overview

There are many steps involved in building a website. The file plan is generally the first step to determine what pages are needed. A template is then created which contains all information that will be on every page, such as doctype declaration, structure tags, semantic tags, h1 heading, logo, and global navigation bars. Often, a second navigation bar is added to the footer. This is a courtesy navigation so users can navigate through the website after scrolling down the page. The external style sheet is created and linked to all pages through the template as well. This allows editing and making global changes through a single document.

Outcomes

In this lab, you will learn to:

1. Compose a completed website.
2. Test website for accessibility.
3. Make design selections based on standards and accessibility guidelines.

	Key Term	Description
1	Global Navigation	A link to every page within the website is included in a grouping, usually at the top

	Key Term	Description
		of each website page. The order and placement of the links is identical on all pages.
2	Courtesy Navigation	A second, usually text-based, navigation coded in the footer of each website page. This allows the user to navigate the website without needing to scroll back up to the top navigation bar.
3	Website	A collection of HTML, CSS, and JavaScript files and codes that work together to form a single entity. All files are linked together and include a way for the user to travel from file to file within each page. All files share the same layout, styling, and theme.
4	Font	A set of characters that display with a particular style and size.
5	Serif Fonts	Fonts that have a projection, or tail, that extends off the end of a character stroke.
6	San-serif Fonts	Fonts without serif, or tails, in the character keystroke.
7	Cursive Fonts	Fonts with connecting strokes from one character to another.
8	Web Content Accessibility Guidelines (WCAG)	Standard of guidelines and recommendations used to ensure accessibility of website content to all users.

Introduction to JavaScript

Introduction

Overview

JavaScript is a vital tool in website development. HTML defines the content of the website page, CSS styles the design and layout of the website, and JavaScript allows the developer to display engaging and dynamic content. Complex front-end and back-end developments both use JavaScript for more complex interactivity, animations, and action within the browser.

Outcomes

In this lab, you will learn to:

1. Learn JavaScript syntax.
2. Examine the Document Object Model (DOM).
3. Edit the DOM in the browser console using JavaScript.

	Key Term	Description
1	JavaScript (JS)	An object-oriented programming language that works in the browser to create dynamic website content.
2	Document Object Model (DOM)	An interface that represents the website page as objects so the document structure can be changed and styled.
3	Variable (var)	An arbitrary name assigned to a storage location in JavaScript; must begin with a letter, underscore, or dollar sign.
4	Method	An action applied to or performed on an object. A JavaScript method contains a function definition.
5	Function	Performs a specific task in JavaScript.
6	Camel Case	Naming convention used in JavaScript where the first letter of each word, phrase, or section is capitalized after the first (e.g., camelCase).

	Key Term	Description
7	Browser Console	An interactive platform built into the browser that allows developers to log information about the browser object.
8	querySelector() Method	Method that returns the first matching CSS element in the document.
9	innerHTML() Property	Used to get or manipulate content within an HTML document using JavaScript. It is a part of the DOM.
10	getElementById() Method	Used to identify and return an element that has the ID attribute applied to it within the HTML document.
11	getElementsByClassName() Method	Used to identify and return a collection of HTML elements that have the class attribute applied to them within the HTML document.
12	Document Object	References the current HTML file.

JavaScript and HTML

Introduction
Overview

JavaScript can be used to edit and add content in real-time within the HTML document, in the same manner, it is applied in the browser Console. External JS can store complex scripts that can be applied to multiple websites by linking the external document to the HTML files, similar to external CSS documents. The flexibility and efficiency of JS make it an excellent skill set to complement HTML and CSS development skills.

Outcomes

In this lab, you will learn to:

1. Use JavaScript to style HTML elements.
2. Use JavaScript to insert HTML elements.
3. Apply external JavaScript to an HTML file.

	Key Term	Description
1	External JavaScript	JavaScript that is coded in a separate document and linked to HTML files.
2	createElement() Method	Creates a new HTML element in the DOM.
3	appendChild() Method	Adds a node to the list of nodes within a parent node, as in a list item to a list or a form element to a form.
4	onMouseOver Event	Occurs when the user's mouse is placed over an element within the website page; used to trigger an action.
5	onMouseOut Event	Occurs when the user's mouse is moved off of an element within the website page; usually used to return the element to its normal state.
6	Internal JavaScript	Coded within the HTML document between <script> and </script> tags; usually placed at the end of the HTML document.

Website Debugging

Introduction
Overview

The ability to debug code is a valuable skill for any developer. Everyone makes errors, even the most experienced developers. Therefore, understanding how to identify and correct errors is important to ensure the quality, functionality, and accessibility of website files. Often, developers will work with others, which can make it difficult if there are errors in the code or the code is not well formatted within the documents. Manually debugging the code is an important first step. Testing accessibility and validation are necessary to ensure accuracy and conformity of standards.

Outcomes

In this lab, you will learn to:

1. Practice manually debugging HTML syntax
2. Practice manually debugging CSS syntax
3. Validate website files

	Key Term	Description
1	Debug	The process of identifying syntax and keystroke errors within code and correcting the errors to meet industry standards and syntax rules.
2	Accessibility	Ensuring website content is available and usable for all users who visit a website, regardless of bandwidth, network connection speed, device, or disability.
3	Validation	The process of checking website code and content for accuracy and ensuring the code meets a set of standard guidelines.