LESSON 4

98-363 Web Development Fundamentals

4.1 Understand Client-side Scripting

4.2 Understand AJAX Concepts

MTA Web Development Fundamentals 4 Test

Microsoft

LESSON 4.1

98-363 Web Development Fundamentals

Understand Client-side Scripting

Lesson Overview

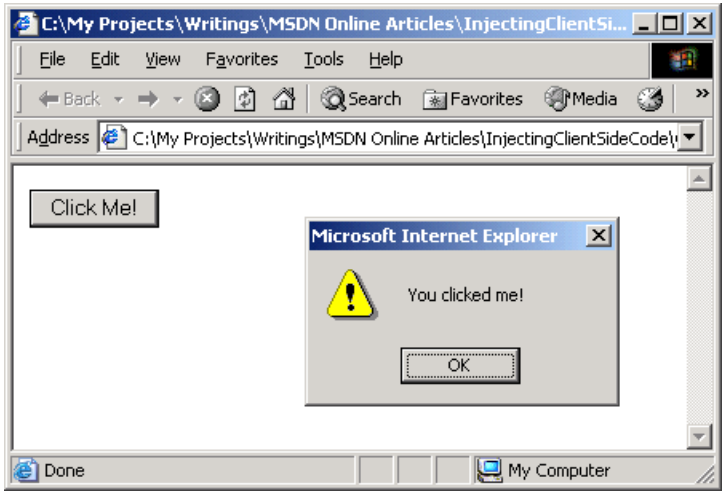
- Client-side scripts run on the client's computer, as opposed to the server.
- After a Web page is downloaded to the client computer, and if the browser is enabled to run scripts, the client-side scripts will run in the browser.
- Although script languages are simpler than programming languages, scripts can add sophisticated logic to Web pages and increase interactivity.

How do you implement client-side scripting?

- You can implement client-side scripting in a Web Part in one of two ways:
  - — Linking a script file to a Web Part
  - — Embedding a script in a Web Part

Example of a Client-side Script

```
<script language="JavaScript">
    <!-- function doClick() { alert("You clicked me!"); }// -->
</script>
```



Linking a Script File to a Web Part

- Write the script in a separate file and place the file on a server running Microsoft Windows Server 2003 with Microsoft Windows SharePoint Services.

- The first time the script file is referenced, the file is fetched from the server and placed in the Web browser's cache.
  - — Subsequent references to the script file only require that the script be fetched from the Web browser's cache.

Linking a Script File to a Web Part (continued)

Advantages:

- More efficient for different Web Parts and Web Part pages to share scripts placed in a common script file.
  - — Fetched from the server only the first time it is referenced.
  - — Subsequent references fetch the script from the Web browser's cache.
- The benefit is more obvious if the script is long.
- More practical and easier to maintain organizationally than embedding the entire script in the Web Part class definition.

Embedding Script in a Web Part

- Specify your script in a Web Part explicitly and load the script only once for all instances of the same Web Part on a Web Part Page.

Advantages:

- Similar to linking to a script file, the script is loaded only once for all instances of the same Web Part on that Web Part page.

- For short scripts, clarity and convenience outweigh performance gains.

Assignment

- Complete student activity 4.1

Lesson Review

- Scripts are blocks of code that are inserted into a Web page and are interpreted at run time.
- Just as a Web browser interprets Hypertext Markup Language (HTML) on a Web page to determine what to display, a Web browser also interprets scripts to determine what actions to take when an event occurs (for example, what happens when a user clicks a button on a Web page).
- Client-side scripts run on the client as opposed to the server.
  - After a Web page is downloaded to the client computer and if the browser is enabled to run scripts, the client-side scripts will run in the browser.

LESSON 4.2

98-363 Web Development Fundamentals

Understand AJAX Concepts

LESSON 4.2

98-363 Web Development Fundamentals

Lesson Overview

In this lesson, you will learn about:

- AJAX
- Asynchronous JavaScript and Hypertext Markup Language (HTML)
- Ajax and ASP.NET AJAX technologies
- Architecture of ASP.NET AJAX

Why AJAX?

- Typically, servers get the data they need and then construct a Web page, which is then sent to the client.
- If the user requests other data, then a new Web page is constructed, which means sending back all kinds of information that the client already has, such as the visual design of the page.
- AJAX allows a page to get an update of the data alone, and then reconstruct the portion of the page that is affected.
- It drastically reduces the amount of information sent over each time the user makes a request for more data, resulting in a much better user experience.

AJAX Technologies

- The *XMLHttpRequest* object
- HTML, Extensible Hypertext Markup Language (XHTML), and Cascading Style Sheets (CSS)
- Extensible Markup Language (XML) and other data transfer formats such as JavaScript Object Notation (JSON)
- Document Object Model (DOM)

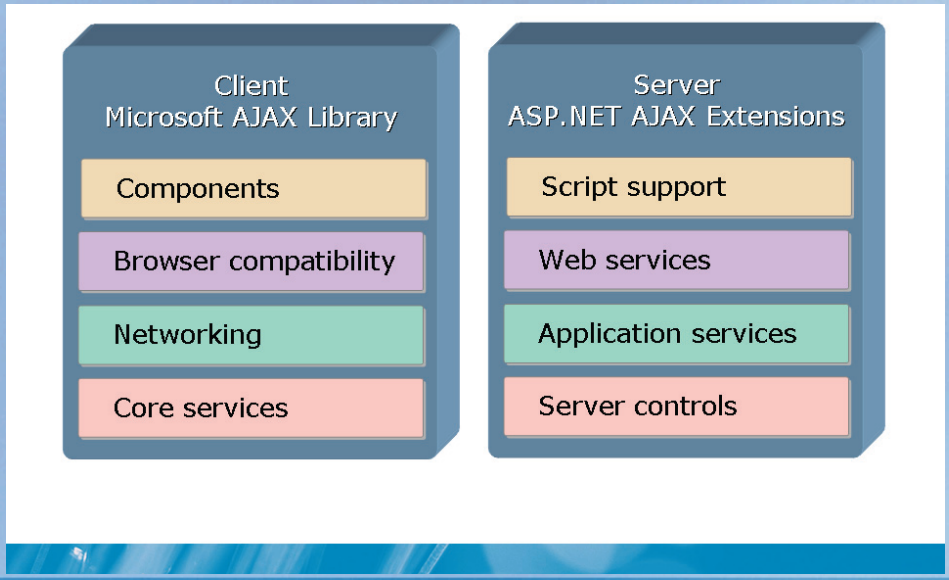
AJAX Provides

- Partial-page updates
- Data integration using Web services
- User-interface (UI) elements that are familiar to users
- Integration with ASP.NET security features
- Increased efficiency means the bulk of processing is done by browser
- Supports the most commonly used browsers

Client and Server Architecture

• The client architecture of ASP.NET AJAX is made of client-script libraries that are composed of JavaScript files.

• The server architecture of ASP.NET AJAX consists of ASP.NET Web server controls and components that are used to create the UI and implement the functionality of an application.



ASP.NET Server Controls

- **ScriptManager**
  - Manages script resources for client components
- **UpdatePanel**
  - Enables a refresh of part of a Web page
- **UpdateProgress**
  - Provides the status of a partial-page update
- **Timer**
  - Performs postbacks at defined intervals

Managing Partial-page Updates

- When partial-page rendering is implemented, an instance of the *PageRequestManager* class becomes available to the page.
- Use JavaScript in a Web page to handle the events raised by the *PageRequestManager* class.
- Get a reference to the current instance of the *PageRequestManager* class by calling the *getInstance* method and then create a function to handle the required event.

Assignment

- Complete student activity 4.2

LESSON 4

98-363 Web Development Fundamentals

Complete Quia Test:

MTA WebFund4 Test