

Scripting Languages

Unit - II

WEB TECHNOLOGY

17UCTE61

Outline

- Overview of Scripting Languages
- Different Scripting Languages
- JavaScript (A Client-side Scripting Language)
- ASP.NET (A Server-side Scripting Language)
- Conclusion

Scripting Language

- A new style of programming language different from system programming languages
- Designed as glue language or system integration language
- A single statement can execute huge number of machine instructions
- Are normally 'typeless'
 - Build complex algorithms and data structures..
- Can create dynamic webpages
 - Change based on user input

Types of Scripting Languages

□ Server-side Scripting Language

- Can use huge resources of the server
- Complete all processing in the server and send plain pages to the client
- Reduces client-side computation overhead

□ Client-side Scripting Language

- Does not involve server processing
- Complete application is downloaded to the client browser
- Client browser executes it locally
- Are normally used to add functionality to web pages e.g. different menu styles, graphic displays or dynamic advertisements

Different Scripting Languages

- Active Server Pages (ASP)
 - Server side scripting language
 - Developed by Microsoft
 - Good at connecting to Microsoft databases
 - Runs only on Microsoft servers
- Perl
 - Old UNIX language
 - Found on all Windows and Linux servers
 - Can handle text manipulation tasks
 - Excellent web scripting language

Different Scripting Languages

□ P H P (HypertextPre-Processor)

- Especially good at connecting to MySQL
- Very popular language
- Runs on UNIX and Windows
- HTML-embedded scripting language
- Syntax looks like C, JAVA, and PERL
- Generate Dynamic content and good User Interface
- Server side execution

□ J S P (Java ServerPages)

- Developed by Sun
- Uses Java
- Provide server-specific framework like Microsoft's ASP

Different Scripting Languages

□ CGI (Common Gateway Interface)

- Server-side solution
- Needs to launch separate instance of application for each web request
- Allows direct interaction with users

□ ASP.NET

- Server-side technology to create faster, reliable and dynamic web pages
- Supports .NET framework languages (C#, VB.NET, JScript.NET)
- Provides flexibility to designers and developers to work separately

Different Scripting Languages

□ VBScript

- Microsoft's scripting language
- Client side Scripting language
- Very easy to learn
- Includes the functionality of Visual Basic

□ JavaScript

- Client-side Scripting language
- Easy to use programming language
- Enhance dynamics and interactive features of a web page
- Allows to perform calculation, write interactive games, add special effects, customize graphic selections, create security passwords



JavaScript

A Client-side Scripting Language

JavaScript

- JavaScript is designed
 - to add interactivity to HTML pages
- JavaScript
 - consists of lines of interpretable computer code
 - gives HTML designers a programming tool
 - is usually embedded directly into HTML pages.
 - allows to put dynamic text into an HTML page
- Java and JavaScript are two completely different languages in both concept and design
- JavaScript's official name is ECMAScript.

JavaScript

- JavaScript is used in millions of web pages
 - to improve the design
 - to validate forms
 - to detect browsers
 - to create cookies
- JavaScript can react to events and can be used to validate data and to create cookies
- Is the most popular scripting language in all major browsers e.g.
 - Internet Explorer
 - Mozilla
 - Firefox
 - Netscape
 - Opera

JavaScript and HTML page

```
<html>
<body>
<script type="text/javascript">
document.write("Hello World!");
</script>
</body>
</html>
```

Tells where the JavaScript starts

Commands for writing output to a page

Tells where the JavaScript ends

This code produce the output on an HTML page:
Hello World!

JavaScript and HTML page

```
<html>
```

```
  <head>
```

```
    <script src="xyz.js"> </script>
```

```
  </head>
```

```
<body>
```

```
</body>
```

```
</html>
```

A separate file



Statements and Comments

- JavaScript statements
 - are codes to be executed by the browser
 - tells the browser what to do
 - commands to the browser
 - add semicolons at the end
 - can be grouped together into blocks using curly brackets
 - try...catch statement allows to test a block of code for errors
- JavaScript comments make the code more readable
 - Single line comments start with `//`
 - Multi line comments start with `/*` and end with `*/`

JavaScript Variables

□ JavaScript Variables

- are containers for storing information e.g. `x=15; length=60.10;`
- hold values or expressions
- can hold a text value like `inname="multimedia"`
- **var statement** can declare JavaScript variables: **`var x; var name;`**

□ Variable names

- are case sensitive i.e. `"myVar"` is not the same as `"myvar"`
- must begin with a letter or the underscore character

JavaScript Operators

- Arithmetic Operators:
 - perform arithmetic operations between the values of the variables
 - Addition (+) , Subtraction (-),
 - Multiplication (*), Division (/), Modulus (%),
 - Increment (+ +), Decrement (- -)
- Assignment Operators:
 - assign values to variables
 - =, +=, -=, *=, /=, %=
- Comparison Operators:
 - determines equality or difference between variables or values
 - Equal to (==), Exactly equal to (===),
 - Not equal (!=), Greater than (>), Less than (<),
 - Greater than or equal to (>=), Less than or equal to (<=)
- Logical Operators:
 - impose the logic between variables or values
 - AND (&&), OR (||), NOT (!)
- Conditional Operator:
 - assign value to a variable based on some conditions
 - ?:

JavaScript Conditional Statements

- **if statement** - to execute some code only if a specified condition is true
- **if...else statement** - to execute some code if the condition is true and another code if the condition is false
- **if...else if...else statement** - to select one of many blocks of code to be executed
- **switch statement** - to select one of many blocks of code to be executed

JavaScript Looping

- JavaScript looping
 - Executes the same block of codes
 - Executes a specified number of times
 - Execution can be controlled by some control logic
 - uses **for**, **while**, **do...while** statements
 - uses **for...in** to iterate through the elements of an array
- **Break** breaks the loop and follows the code after the loop
- **Continue** breaks the loop and continues with next value.

JavaScript Functions and Events

□ JavaScript Functions

- Can be called with the functionname
- Can also be executed by an event
- Can have parameters and return statement

□ Events

- are actions that can be detected e.g. OnMouseOver, onMouseOut etc.
- are normally associated with functions
- `<input type="text" size="30" id="email" onChange="checkEmail()">`

JavaScript: Events

- Javascript actions may be triggered from events, e.g. changes on form fields or a submit button being clicked:
 - onfocus = Form field gets focus (validation)
 - onblur= Form field loses focus (validation)
 - onchange= Content of a field changes (validation)
 - onselect= Text is selected
 - onmouseover= Mouse moves over a link (animated buttons)
 - onmouseout= Mouse moves out of a link (animated ...)
 - onclick= Mouse clicks an object
 - onload= Page is finished loading (initial actions, info,)
 - onSubmit= Submit button is clicked (validation etc.)

JavaScript Popup boxes

- JavaScript can create:
 - Alert box: to make sure information comes through to the user.
 - Confirm box: to verify or accept something
 - Prompt box: the user to input a value before entering a page

JavaScript and OOP

- JavaScript
 - is an Object Oriented Programming language
 - contains built-in JavaScript objects
 - String
 - Date
 - Array
 - Boolean
 - Math
 - RegExp
 - Window
 - Navigator
 - Screen
 - Location
 - History etc.
 - also allows to define new objects
 - objects contain Properties and Methods
 - objects can be used as variable types

JavaScript: DOM

- To access the data in the HTML page
 - needs some data structures to access the HTML page.
- Many browser implement an interface to what is called the Document Object Model (DOM)
 - It allows to output the document in the changed form to the browser.
- DOM is a representation of the document in an object form, accessible from JavaScript programs

JavaScript HTML DOM Objects

□ DOM Objects

- **Document** Represents the entire HTML document and can be used to access all elements in a page
- **Anchor** Represents an <a> element
- **Area** Represents an <area> element inside an image-map
- **Base** Represents a <base> element
- **Body** Represents the <body> element
- **Button** Represents a <button> element
- **Event** Represents the state of an event
- **Form** Represents a <form> element
- **Frame** Represents a <frame> element
- **Frameset** Represents a <frameset> element
- **Iframe** Represents an <iframe> element

JavaScript HTML DOM Objects

□ DOM Objects

- **Image** Represents an element
- **Input button** Represents a button in an HTML form
- **Input checkbox** Represents a checkbox in an HTML form
- **Input file** Represents a fileupload in an HTML form
- **Input hidden** Represents a hidden field in an HTML form
- **Input password** Represents a password field in an HTML form
- **Input radio** Represents a radio button in an HTML form
- **Input reset** Represents a reset button in an HTML form
- **Input submit** Represents a submit button in an HTML form
- **Input text** Represents a text-input field in an HTML form
- **Link** Represents a <link> element
- **Meta** Represents a <meta> element
- **Option** Represents an <option> element
- **Select** Represents a selection list in an HTML form
- **Style** Represents an individual style statement
- **Table** Represents a <table> element
- **TableData** Represents a <td> element
- **TableRow** Represents a <tr> element
- **Textarea** Represents a <textarea> element

Dynamic JavaScript

- JavaScript can be used to directly change the DOM model and thereby to change the document.
- The DOM model can also be used to manage XML content (and, for instance, to generate HTML from it).
- It can also use XMLHttpRequest objects to request data from the server without loading the whole page again. This provides possibilities to load new content to the page without re-loading the page.
 - Using this technology in combination of DOM and the basic HTML/CSS (or XHTML/CSS) is sometimes called Ajax.



ASP.NET

A Server-side Scripting Language

ASP.NET

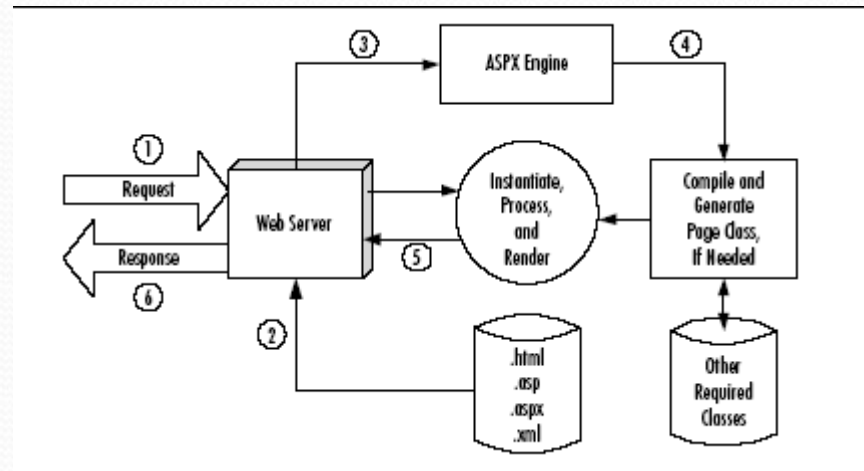
□ ASP.NET

- Is the latest version of Microsoft's Active Server Pages (ASP) technology
- Is a part of the Microsoft .NET framework
- Is a powerful tool for web development
- Is a program that runs inside IIS (Internet Information Services)
- Can contain HTML, XML and other scripts
- File extension is ".aspx"

How does ASP.NET Work?

- A browser requests an HTML file, the server returns the file
- A browser requests an ASP.NET file, IIS passes the request to the ASP.NET engine on the server
- ASP.NET engine reads the file, line by line, and executes the scripts in the file
- ASP.NET file is returned to the browser as plain HTML

How does ASP.NET Work?



ASP.NET - Server Controls

- Server controls are tags that are understood by the server
- ASP.NET has solved the "spaghetti-code" problem of Classic ASP with Server Controls
- There are three kinds of server controls:
 - HTML Server Controls - Traditional HTML tags
 - Web Server Controls - New ASP.NET tags
 - Validation Server Controls - For input validation

ASP.NET Features

- ASP.NET Event Handler
 - is a subroutine that executes code for a given event
 - Page_Load event is triggered when a page loads, and ASP.NET will automatically call the subroutine Page_Load, and execute the code inside it
- ASP.NET Web Forms
 - All server controls must appear within a <form> tag, and the <form> tag must contain the runat="server" attribute. The runat="server" attribute indicates that the form should be processed on the server.
- ASP.NET Data Binding
 - The following controls are list controls which support data binding:
 - asp:RadioButtonList
 - asp:CheckBoxList
 - asp:DropDownList
 - asp:Listbox
- ADO.NET is also a part of the .NET Framework. ADO.NET is used to handle data access. With ADO.NET you can work with databases.

ASP.NET Features

- Master Pages, Themes
- Standard controls for navigation
- Standard controls for security
- Roles, personalization, and internationalization services
- Improved and simplified data access controls
- Full support for XML standards like, XHTML, XML, and WSDL
- Improved compilation and deployment (installation)
- Improved site management
- New and improved development tools

Conclusion

- Scripting languages make the web development work
 - Easier
 - Faster
 - Best utilization of resources
- It is the programmer's choice which one will suite the best considering
 - Knowledge
 - Practice
 - Resouce



Thank you