Introduction to JavaScript

JavaScript is used in millions of Web pages to improve the design, validate forms, and much more. JavaScript was developed by Netscape and is the most popular scripting language on the internet.

JavaScript is a script based programming language that supports the development of both client and server components of Web based applications. The process of reading the HTML file and identifying the elements contained in the file is referred as parsing. When a script is encountered during parsing, the browser executes the script before continuing with further parsing. The script can perform action, such as generating HTML code that affects the display of the browser window. Characteristic features of JavaScript are listed as follows:

- JavaScript was designed to add interactivity to HTML pages
- JavaScript is a scripting language - a scripting language is a lightweight programming language
- A JavaScript is lines of executable computer code
- A JavaScript is usually embedded directly in HTML pages. **JavaScript can put dynamic text into an HTML page -** A JavaScript statement like :

  document.write("<h1>" + name + "</h1>")

  can write a variable text into an HTML page
- JavaScript is an interpreted language (means that scripts execute without preliminary compilation)
- Everyone can use JavaScript without purchasing a license
- JavaScript is supported by all major browsers, like Netscape and Internet Explorer
- **JavaScript can react to events -** A JavaScript can be set to execute when something happens, like when a page has finished loading or when a user clicks on an HTML element
- **JavaScript can read and write HTML elements -** A JavaScript can read and change the content of an HTML element
- **JavaScript can be used to validate data -** A JavaScript can be used to validate form data before it is submitted to a server, this will save the server from extra processing

Embedding JavaScript in HTML

JavaScript statement can be included in HTML document by enclosing the statement between container element <SCRIPT>. <SCRIPT> element has the following attributes

- LANGUAGE attribute is set to JavaScript to identify the script being JavaScript. It can be some other scripting language , such as Visual Basic Script.

  <SCRIPT LANGUAGE="JavaScript">

  or it can be written as

  <SCRIPT TYPE="text/javascript">

- Scripts in the <BODY> section will be executed WHILE the page loads, while the Scripts in the <HEAD> section will be executed when CALLED.
- SRC attribute is used to specify a file containing JavaScript statements. The file could have been named anything, but it should end with the .js extension such as
  <SCRIPT LANGUAGE=”JavaScript” SRC=”jsfile.js”>

Example 11.1

```
<HTML>
    <HEAD>
        <TITLE>
            WEB Page 11.1: Welcome to the world of JavaScript
        </TITLE>
    </HEAD>
    <BODY BGCOLOR="#FFCCFF" TEXT=blue>
        <SCRIPT LANGUAGE="JavaScript">
            document.write("Welcome to the world of JavaScript !!")
            document.write("<H1>Welcome JavaScript !!</H1>")
        </SCRIPT>
</BODY>
</HTML>
```

The script has a single statement, document.write(), that writes the string to the body of the current document object. The text written by the script becomes part of the HTML document displayed by the browser. In Java each statement ends with a semicolon (;), but in JavaScript it is not so.

Handling Browsers that don’t handle JavaScript

Some browsers can’t handle JavaScript, so they ignore <SCRIPT> tag, which means script will appear directly in the browser as plain text. To avoid this, HTML provides a method to conceal JavaScript statements from such JavaScript-challenged browsers. There are two ways to do so:

1. Comment Tags

Combination of HTML comment tags and JavaScript comment tags takes care of hiding the script from the browser.

- The <!-- begins the HTML comment and the -> tag ends the comment.
- In JavaScript
  (i) The // string identifies a single line comment
  (ii) /* with */ strings are used for multiline comment

Use HTML comment tags to surround the JavaScript statements. Make the closing comment tag a JavaScript comment by starting it with two forward slashesor even JavaScript-enabled browsers will ignore this script.

```
<!—Begin hiding JavaSScript
    JavaScript statements
```

```
            // End hiding JavaScript statements -->
```

2. NOSCRIPT element

The above technique takes care of hiding the script from the browser, but to let the users know about JavaScript challenged browser they are using the <NOSCRIPT> element is used. JavaScript enabled browsers will ignore this statemement

Example 11.1 (revised)

```
<HTML>
        <HEAD>
                <TITLE>
                        WEB Page 11.1: Comment and NOSCRIPT
                </TITLE>
        </HEAD>
        <BODY BGCOLOR="#FFCCFF" TEXT=blue>
                <SCRIPT LANGUAGE="JavaScript">
                <!-- hide JavaScript statements
                        document.write("Welcome to the world of JavaScript !!")
                        document.write("<H1>Welcome JavaScript !!</H1>")
                // end hiding JavaScript -->
                </SCRIPT>
                <NOSCRIPT>
                        Sorry browser you are using does not support JavaScript
                </NOSCRIPT>
        </BODY>
</HTML>
```