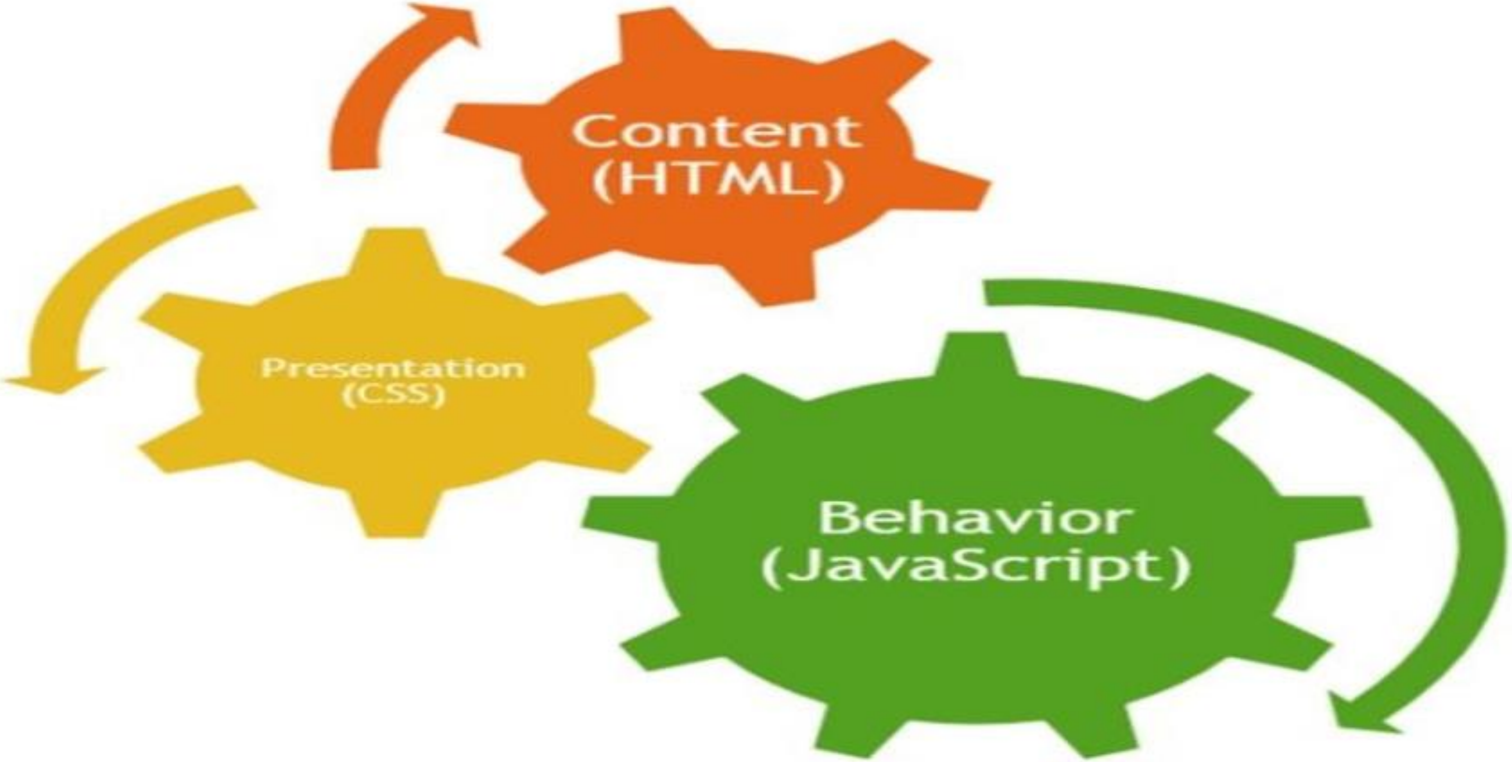
Introduction

JavaScript

- JS – What?
  - A programming language designed for web pages
  - It is embedded directly into HTML Pages
  - An interpreted, client-side, event based language
  - It is dynamic, lightweight and case-sensitive


Content
(HTML)
Presentation
(CSS)
Behavior
(JavaScript)

JavaScript History

● Created by Brendan Eich (co-founder of Mozilla) in 1995 during his time at Netscape Communications
● It was inspired by Java, Scheme and Self
● Originally called Mocha (a name chosen by Marc Andreessen)

JavaScript History

- In Sep 1995 the name was changed to LiveScript
- Then, in Dec 1995, upon receiving a trademark license from Sun, the name JavaScript was adopted
- The general-purpose JavaScript engine had been embedded in web browsers (Netscape, IE, etc..)

JavaScript History

● The ECMA-262 (European Computer Manufacturer’s Association) Specification defined a standard version of JavaScript language

● ECMAScript Edition 1 First standardized version of JavaScript

● ECMAScript Edition 2 is the second official standard

JavaScript History

● ECMAScript 3 (ES3) was released in December 1999. More advanced language, Includes regular expressions and exception handling

● ECMAScript 4 (ES4) A new standard includes features such as JSON (JavaScript Object Notation) & class based object-oriented programming

● ECMAScript 5 (ES5) was released in December 2009. It adds in getter and setter properties, introduces features for robust programming via a strict mode and JSON handling arrays

● ECMAScript 6 (ES6) was released in June 2015, and is the latest official
version of JavaScript

JavaScript Advantages

● JavaScript is executed on the client side
– The code is executed on the user's processor instead of the web server
– This means less load on server
● JavaScript is relatively easy language
– The JavaScript language is relatively easy to learn and comprises of syntax that is close to English

JavaScript Advantages

● Increased interactivity: We can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard
● Richer interfaces: You can use JavaScript to include such items as drag-and-drop components and sliders to give a rich interface to your site visitors

JavaScript Limitations

● JavaScript cannot write to files on the server without the help of a server side script

– Ajax, JavaScript can send a request to the server

● JavaScript cannot access back-end databases without a server side script

JavaScript Limitations

● JavaScript cannot read from or write to files in the client
– Even though JavaScript is running on the client computer the one where the web page is being viewed) it is not allowed to access anything outside of the web page itself
– This is done for reasons of security
● JavaScript does not have multi-threading
– JavaScript does not have multi-threading capabilities

JS - first program

JavaScript can be implemented using JavaScript statements that are placed within the <script>... </script> HTML tags in a web page

```
<script language="javascript" type="text/javascript">
. . . JavaScript code . . .
</script>
```

JS – HTML script element

- Attributes used in <script> tag
  - src
  - language (deprecated)
  - type
- Source attribute “src” is used to specify the source JavaScript file
- Language attribute is used to specify the JavaScript language
- Type attribute is used to specify that the file type is text file and contains JavaScript code

JS - Internal

-
  ```
  <!DOCTYPE html>
  <html>
  <head>
  <script> . . . JS code . . . </script>
  </head>
  <body>
  <script> . . . JS code . . . </script>
  </body>
  </html>
  ```

JS – First Program

-
  ```
  <body>
  <script>
  document.write("Hello JavaScript!!");
  </script>
  </body>
  ```

JS - External

● External (or linked script) JavaScript can be inserted using src attribute

Syntax :
<script src="URL"> </script>
Absolute URL : http://www.example.com/example.js
Relative URL : /script/example.js

JS – External

- **Example** 1 :
  <script src=“script.js”></script>
  Example 2 :
  <script src=“/js/script.js”></script>

JS – Linked Script

- ● Benefits :

  – Decoupling logic from presentation

  – Re-use across pages

  ● Disadvantage

  – Script may load slowly

  – Linking to remote script may pose security threat

Coding Conventions

- ● JavaScript Files
  – JavaScript programs should be stored in and delivered as .js files
  ● Comments
  // Single line comment
  /* Multi
  * Line comment
  */

Coding Conventions

● Variable Names

– Use camelCase for identifier names (variables and functions)

– The name should be descriptive and concise

– All names start with a letter

– Variable names are case sensitive

Coding Conventions

● Naming Conventions

– Variable and function names written as camelCase

– Global variables written in UPPERCASE

– Constants written in UPPERCASE

JavaScript with html and css

- **<html>**
- **<body>**
- **<h1>My First JavaScript</h1>**
- **<p id="demo">JavaScript can change the style of an HTML element.</p>**
- **<script>**
- **function myFunction() {**
- **document.getElementById("demo").style.fontSize = "25px";**
- **document.getElementById("demo").style.color = "red";**
- **document.getElementById("demo").style.backgroundColor = "yellow";**
- **}**
- **</script>**
- **<button type="button" onclick="myFunction()">Click Me!</button>**
- **</body>**
- **</html>**

JavaScript with html

- **<html>**
- **<body>**
- **Name: <input type="text" id="myText" value="Mickey">**
- **<p>Click the button to change the value of the text field.</p>**
- **<button onclick="myFunction()">Try it</button>**
- **<script>**
- **function myFunction() {**
- ** document.getElementById("myText").value = "Johnny Bravo";**
- **}**
- **</script>**
- **</body>**
- **</html>**