Web Technology
(KCS-602)
Unit 3
Introduction to JavaScript Documents

Prepared By

Abhishek Kesharwani

Assistant Professor,UCER Naini,Allahabad

Index

JavaScript Introduction

- JavaScript was designed to add interactivity to HTML pages
- JavaScript is a scripting language
- A scripting language is a lightweight programming language
- JavaScript is usually embedded directly into HTML pages
- JavaScript is an interpreted language (means that scripts execute without preliminary compilation)

What can a JavaScript do?

- JavaScript gives HTML designers a programming tool -
- JavaScript can put dynamic text into an HTML page
- JavaScript can react to events -
- JavaScript can read and write HTML elements
- JavaScript can be used to validate data.

Limitations with JavaScript

- Client-side JavaScript does not allow the reading or writing of files. This has been kept for security reason.
- JavaScript can not be used for Networking applications because there is no such support available.
- JavaScript doesn't have any multithreading or multiprocess capabilities.

JavaScript Vs Java

- Java is purely an object-oriented language, JavaScript is just a scripting tool
- JavaScript is not compiled and executed; the client directly interprets it
- Object references must exist at compile-time in Java (static binding) whereas they are checked only at runtime in JavaScript (dynamic binding)

Advantages of JavaScript

- Less server interaction:
- Immediate feedback to the visitors:
- Increased interactivity:
- Richer interfaces:
- Form validation in the client's place itself, thereby reducing the burden on the server
- Javascript timer on the client to check how much time he/she takes to fill a form.

JavaScript Syntax

JavaScript in <head>

- JavaScript function is placed in the <head> section of an HTML page.

JavaScript in <body>

- JavaScript function is placed in the <body> section of an HTML page.

External JavaScript

- Scripts can also be placed in external files.External scripts are practical when the same code is used in many different web pages.
- JavaScript files have the file extension **.js**.

JavaScript in <head></head> section

```
<html>
<head>
<script type="text/javascript">
 function sayHello()
 {
 alert("Hello World")
 }</script>
</head>
<body><input type="button" onclick="sayHello()"
    value="Say Hello“ > </body> </html>
```

JavaScript in <body></body> section:

```
<html>
<head> </head>
<body>
<script type="text/javascript“>
 document.write("Hello World“)
</script> <p>This is web page body </p>
 </body>
</html>
```

JavaScript in External File

```
<html> <head>
<script type="text/javascript" src="filename.js”>
</script>
</head>
 <body> ..... </body>
 </html>
```

JavaScript Variable

A **JavaScript variable** is simply a name of storage location.

There are two types of variables in JavaScript : local variable and global variable.

There are some rules while declaring a JavaScript variable (also known as identifiers).

- Name must start with a letter (a to z or A to Z), underscore(_), or dollar($) sign.
- After first letter we can use digits (0 to 9), for example value1.
- JavaScript variables are case sensitive

JavaScript local variable

A JavaScript local variable is declared inside block or function. It is accessible within the function or block only.

For example:

```
<script>
function abc(){
var x=10;//local variable
}
</script>
```

JavaScript global variable

A **JavaScript global variable** is accessible from any function. A variable i.e. declared outside the function or declared with window object is known as global variable.

For example:

```
<script>
var data=200;//gloabal variable
function a(){
document.writeln(data);
}
function b(){
document.writeln(data);
}
a();//calling JavaScript function
b();
</script>
```

Document Object Model

- The **document object** represents the whole html document.
- When html document is loaded in the browser, it becomes a document object.
- It is the **root element** that represents the html document.
- it has properties and methods. By the help of document object, we can add dynamic content to our web page.

window.document

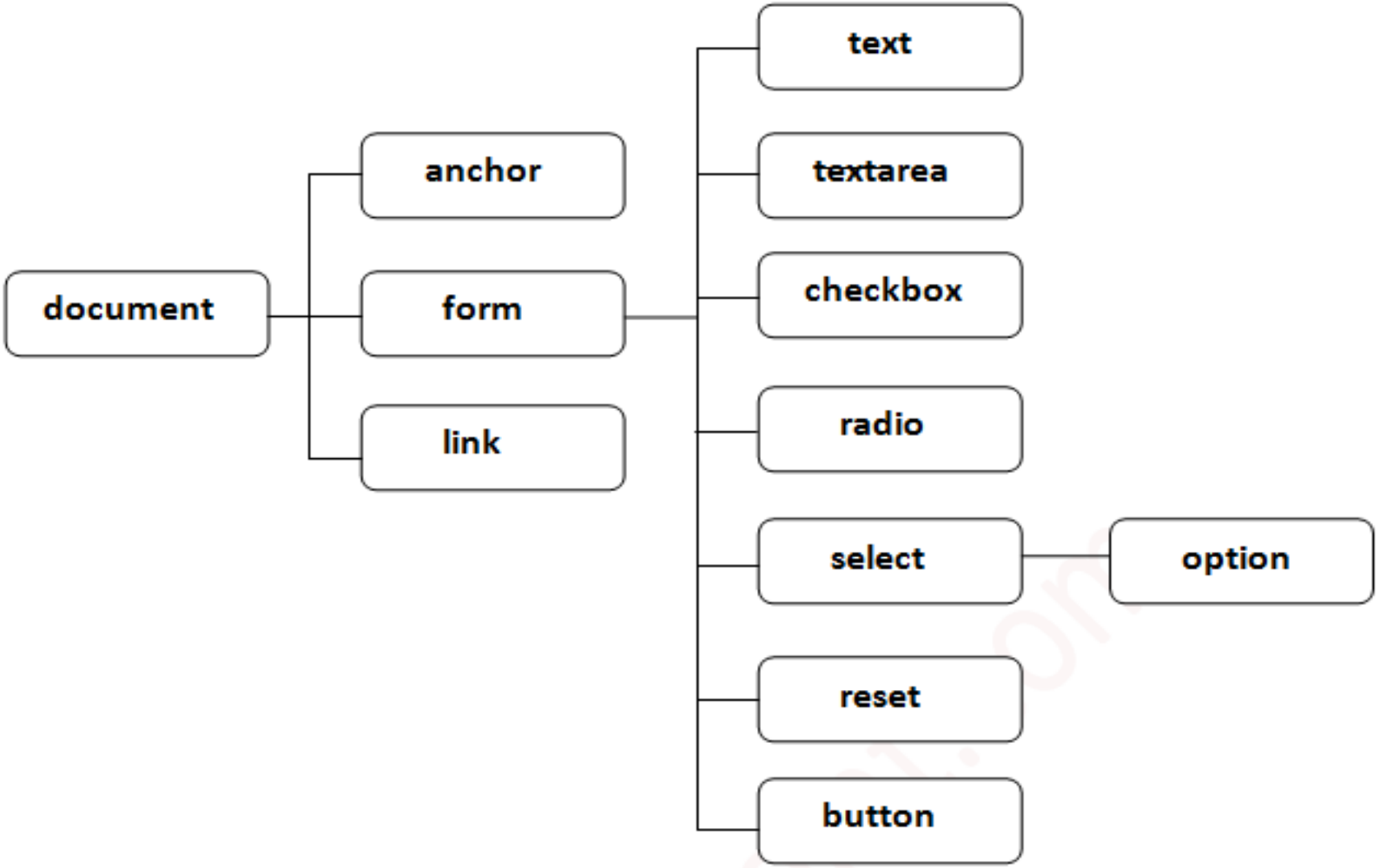
Is same as

document

DOM

- According to W3C - *"The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document."*

Properties of document object



Methods of document object

Method	Description
write("string")	writes the given string on the doucment.
writeln("string")	writes the given string on the doucment with newline character at the end.
getElementById()	returns the element having the given id value.
getElementsByName()	returns all the elements having the given name value.

Example

```
<script type="text/javascript">
function printvalue(){
var nm=document.form1.n1.value;
alert("Welcome: "+nm);
}
</script>

<form name="form1">
Enter Name:<input type="text" name="n1"/>
<input type="button" onclick="printvalue()" value="print
   name"/>
</form>
```

document.getElementById() method

The **document.getElementById()** method returns the element of specified id.

```
<script type="text/javascript">
function getcube(){
var number=document.getElementById("number").value;
alert(number*number*number);
}
</script>
<form>
Enter No:<input type="text" id="number" name="number"/>
<br/>
<input type="button" value="cube" onclick="getcube()"/>
</form>
```

Javascript - innerHTML

- The **innerHTML** property can be used to write the dynamic html on the html document.
- It is used mostly in the web pages to generate the dynamic html such as registration form, comment form, links etc.

Javascript - innerText

- The **innerText** property can be used to write the dynamic text on the html document. Here, text will not be interpreted as html text but a normal text.
- It is used mostly in the web pages to generate the dynamic content such as writing the validation message, password strength etc.

```
<script type="text/javascript" >
function validate() {
var msg;
if(document.myForm.userPass.value.length>5){
msg="good";
}
else{
msg="poor";
}
document.getElementById('mylocation').innerText=msg;
 }

</script>
<form name="myForm">
<input type="password" value="" name="userPass" onkeyup=
"validate()">
Strength:<span id="mylocation">no strength</span>
</form>
```

Interview Questions

- What is the difference between Java and JavaScript.
- Why do you think JavaScript plays important role in web designing?