Javascript

Introduction

Lambda

What is JavaScript?

* An object-oriented computer programming language commonly used to create interactive effects within web browsers. (Google defn.)

* JavaScript is the programming language of HTML and the Web. (w3schools)

Hello World

```
alert(“Hello World!”);
```

How to include JS in a web page

* Inside <script> tags

```
<script>
   alert(“Hello World!”);
</script>
```

* Within external file

```
<script src=“page.js“></script>
```

Javascript Basics

Syntax

- Mostly like C and java.
- Everything is case-sensitive.
- Variables are loosely typed.
- End-of-line semicolons are optional.
- Comments are the same as in Java, C, and Perl.

Primitive values

- undefined
- null
- boolean
- number
- string

Use `typeof` keyword to check type of variable.

Examples

```
var q = “Hello world!”; // String
var w = 100; // Number
var e = 123.456; // Number
var r = true; // Boolean
var t = undefined;
var y = null;
var u = [ 1, 2, ”yay!”, [ 2.3, ”yohoo!” ], false ]; // Array
var i = { // JS Object
        “key1”: “I am a string”,
        “key2”: 1234,
        “key3”: {
                “key_x”: [ 1, 2, 3 ]
        }
};
```

Statements

- **The if statement**

  -
    ```
    if (condition) statement1 else statement2
    ```
- **do-while**

  -
    ```
    do { statements } while (condition )
    ```
- **While**

  -
    ```
    while (condition) { statements}
    ```
- **for**

  -
    ```
    for (initialization; condition; post-loop-expression)
    { statements}
    ```
- **for-in**

  ```
  for (property in expression) {statements}
  ```

Statements

Arithmetic: + - * / % ** ++ --

Bitwise: & | << >> >>> ^ ~

Logical: && || !

Relational: < > <= >= == != === !==

== vs ===

- 1 == “1” // true
- 1 === “1” // false
- 0 == false // true
- 0 === false // false
- null == undefined // true
- null === undefined // false

Function defination

```
function fnName(a){
        alert(a);
}

var fnName1 = function(a){
    return a*5;
};

var anObject = {
    “fnName”: function(a){
        return fnName1(a);
    }
};

var arr = [1 , fnName]
```

Calling Functions

```
fnName(“Hello”);

var b = fnName1(10);

arr[1](“Another function”);

var c = anObject.fnName(100);

var d = anObject[“fnName”](100);
```

Functions (contd.)

- Functions are also object, in fact everything is an object in JavaScript
- Functions can return any data type, ‘undefined’ if none specified.

Variables scope

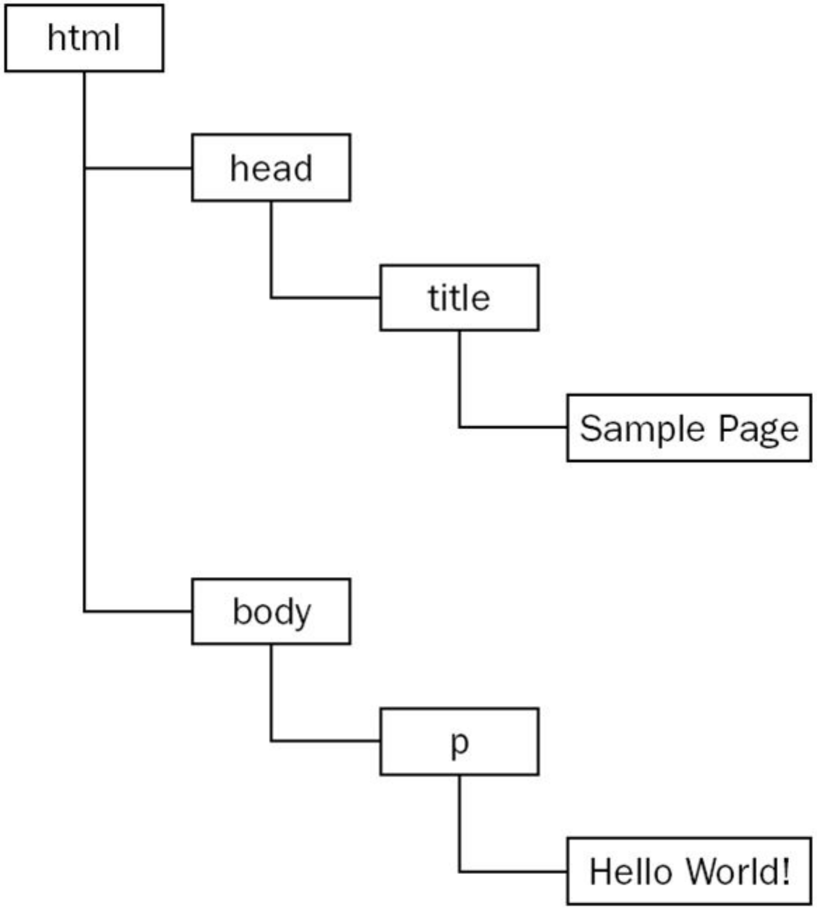
```
var x = 10; // global

function f() {
   var y = 20; // local
   function g(){
      var z = 30; // local
   }
   if (true) {
      var a = 5;
   }
   alert(a); // 5
}
```

The Document Object Model (DOM)

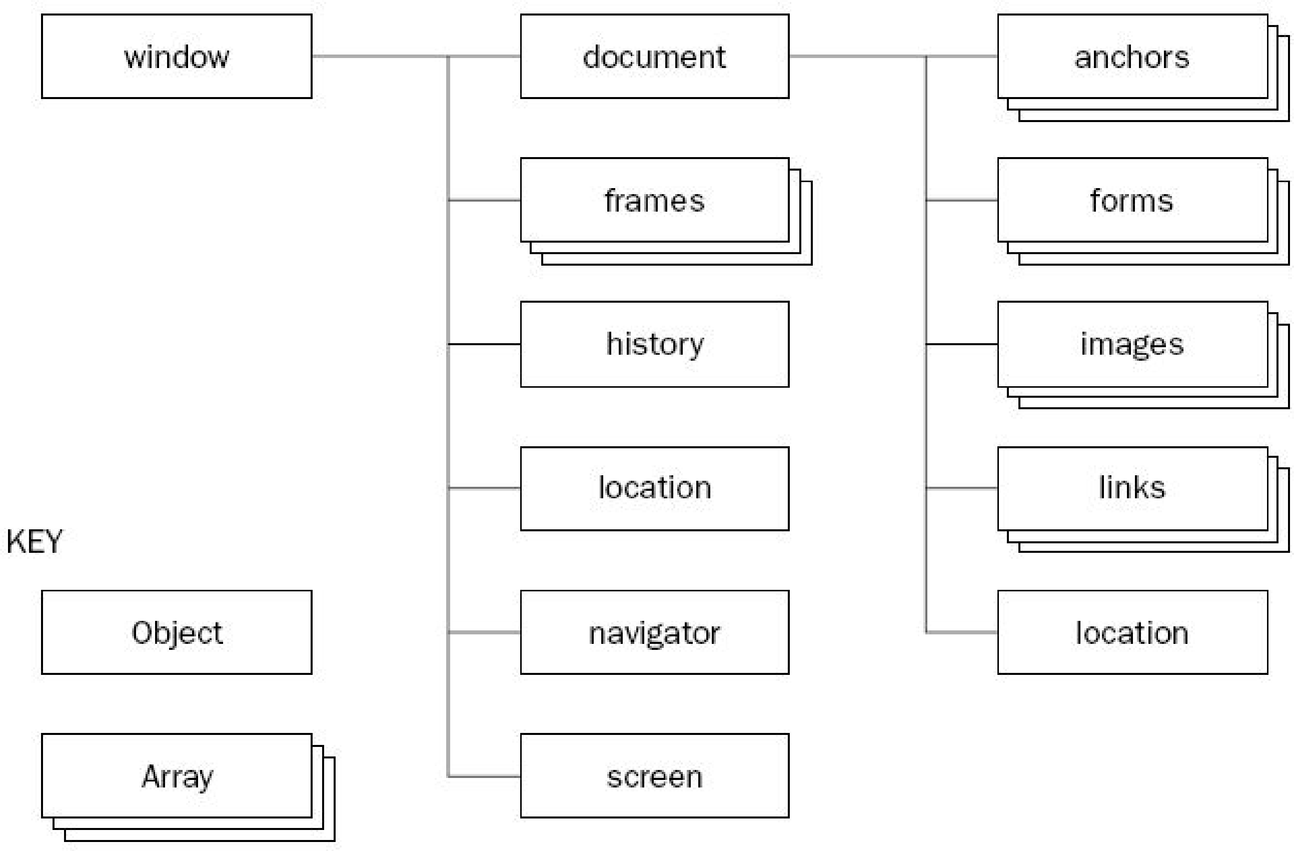
- The *Document Object Model* (DOM) is an application programming interface (API) for HTML as well as XML.

DOM

```
<html>
    <head>
        <title>Sample Page</title>
    </head>
    <body>
        <p>Hello World!</p>
    </body>
</html>
```



Browser Object Model (BOM)



DOM and JS World

```
var x = document.getElementById(“myDIv”);
x.innerHTML = “new html inside element”;
x.onclick = function(){
      alert(“Clicked!”);
}

var x = document.getElementsByTagName(“div”);
x[0].innerHTML = “new html inside div”;

var x = document.getElementsByClassName(“aClass”);
x[0].innerHTML = “new html inside the element”;
x[0].attribute = “new value”;
x[0].setAttribute( attribute , “new value” );
X[0].style.color = “#f00”;
```

Thank You!