Java Script

H S Rana

CIT,UPES

April 21, 2014

Java Script

- JavaScript is a client-side scripting language that runs entirely inside the web browser.
- JavaScript first appeared in the Netscape Navigator browser in 1995
- we place it between opening <script> and closing </script> HTML tags.
- You can place an unlimited number of scripts in an HTML document.
- Scripts can be in the <body> or in the <head> section of HTML, and/or in both.
- It is a common practice to put functions in the <head> section, or at the bottom of the page.

Example

Let us consider an example

```
<html>
    <head><title>Hello World</title></head>
    <body>
        <script type="text/javascript">
            document.write("Hello World")
        </script>
        <noscript>
            Your browser doesn't support or has disabled JavaSc
        </noscript>
    </body>
</html>
```

Including JavaScript file

- In addition to writing JavaScript code directly in HTML documents, you can include files of JavaScript code either from your website or from anywhere on the Internet.

```
The syntax for this is:
<script type="text/javascript" src="script.js"></script>
to pull a file in from the Internet, use:
<script type="text/javascript"

src="http://someserver.com/script.js"> </script>
```

Semicolons

- Unlike PHP, semicolons are not generally required by JavaScript if you have only one statement on a line.
  - x+=10
- However, when you wish to place more than one statement on a line, they must be separated with semicolons, like this
  - x += 10; y -= 5; z = 0

Variables

- A variable may include only the letters a-z, A-Z, 0-9, the $ symbol, and the underscore (_).
- No other characters such as spaces or punctuation are allowed in a variable name.
- The first character of a variable name can be only a-z, A-Z, $, or _ (no numbers).
- Names are case-sensitive. Count, count, and COUNT are all different variables.
- There is no set limit on variable name lengths.

Example

Let consider following example

```
String Variables
================
greeting = "Hello there"
warning  = 'Be careful'
Numeric Variables
=================
count       = 42
temperature = 98.4
Arrays
=======
toys = ['bat', 'ball', 'whistle', 'puzzle', 'doll']
face =
[
     ['R', 'G', 'Y'],['W', 'R', 'O'],
     ['Y', 'W', 'G']
]
```

Table 14-2. Arithmetic operators

Operator	Description	Example
+	Addition	`j + 12`
-	Subtraction	`j - 22`
*	Multiplication	`j * 7`
/	Division	`j / 3.14`
%	Modulus (division remainder)	`j % 6`
++	Increment	`++j`
--	Decrement	`--j`

Table 14-3. Assignment operators

Operator	Example	Equivalent to
=	j = 99	j = 99
+=	j += 2	j = j + 2
+=	j += 'string'	j = j + 'string'
-=	j -= 12	j = j - 12
*=	j *= 2	j = j * 2
/=	j /= 6	j = j / 6
%=	j %= 7	j = j % 7

Table 14-4. Comparison operators

Operator	Description	Example
==	Is **equal** to	j == 42
!=	Is **not equal** to	j != 17
>	Is **greater than**	j > 0
<	Is **less than**	j < 100
>=	Is **greater than or equal** to	j >= 23
<=	Is **less than or equal** to	j <= 13
===	Is **equal** to (and of the same type)	j === 56
!==	Is **not equal** to (and of the same type)	j !== '1'

Operators

Table 14-5. Logical operators

Operator	Description	Example
&&	And	`j == 1 && k == 2`
||	Or	`j < 100 || j > 0`
!	Not	`! (j == k)`

String Concatenation

- JavaScript string concatenation is like Python and Java and uses "+" versus PHP which uses "."
- Like the PHP "." operator, it automatically converts non- string values to strings as needed

```
>>> x = 12
12
>>> y = 'Hello ' + x + ' people'
"Hello 12 people"
```

Variable Typing

JavaScript is a loosely typed language and does automatic type conversion when doing expressions

```
>>> x = "123" + 10;
"12310"
>>> x = ("123" * 1) + 10;
133
>>> x = ("fred" * 1 )
NaN
>>> x = x + 1
NaN
```

Function

JavaScript functions are used to separate out sections of code that perform a particular task.

```
<script>
function product(a, b)
{
    return a*b
}
</script>
```

Global Variables

Global variables are ones defined outside of any functions (or within functions, but defined without the var keyword)
for example

- a = 123 // Global scope
- var b = 456 // Global scope
- if (a == 123) var c = 789 // Global scope

Local Variables

- Parameters passed to a function automatically have local scope
- Variable defined inside the function var keyword
- there is one exception. Arrays are passed to a function by reference, so if you modify any elements in an array parameter, the elements of the original array will be modified.

```
<script>
function test()
{   a = 123                  // Global scope
    var b = 456                  // Local scope
    if (a == 123) var c = 789 // Local scope}
</script>
```