CT053-3-1
Fundamentals of Web Design & Development

Introduction to JavaScript

Why JavaScript?

JavaScript is one of the 3 languages all web developers must learn:

1. HTML to define the content of web pages
2. CSS to specify the layout of web pages
3. JavaScript to program the behavior of web pages

Java vs. JavaScript

- JavaScript and Java are completely different languages, both in concept and design.
  - JavaScript was invented by Brendan Eich in 1995, and became an ECMA standard in 1997.
  - ECMA-262 is the official name of the standard. ECMAScript is the official name of the language.
- Java is a general purpose programming language, and JavaScript is used on websites to make them animated and interactive.

JavaScript Introduction

A·P·U
ASIA PACIFIC UNIVERSITY
OF TECHNOLOGY & INNOVATION

- What JavaScript can do?
  - JavaScript can change html content
  - JavaScript can change HTML attributes
  - JavaScript can change HTML styles (CSS)
  - JavaScript can hide HTML elements
  - JavaScript can show HTML elements
  - and many more!

Where To include JavaScript?

- The <script> Tag
  - In HTML, JavaScript code must be inserted between <script> and </script> tags.
- JavaScript Functions and Events
  - A JavaScript function is a block of JavaScript code, that can be executed when "called" for.
  - For example, a function can be called when an event occurs, like when the user clicks a button.

Will be covered in separate chapter

JavaScript in <head> or <body>

- You can place any number of scripts in an HTML document.
- Scripts can be placed in the <body>, or in the <head> section of an HTML page, or in both.
- 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.
  - To use an external script, put the name of the script file in the src (source) attribute of a <script> tag.

```
<script src="myScript.js"></script>
```

External JavaScript Advantages

- Placing scripts in external files has some advantages:
  - It separates HTML and code
  - It makes HTML and JavaScript easier to read and maintain
  - Cached JavaScript files can speed up page loads
  - To add several script files to one page - use several script tags:

```
<script src="myScript1.js"></script>
<script src="myScript2.js"></script>
```

JavaScript Output

- JavaScript can "display" data in different ways:
  - Writing into an alert box, using **window.alert()**.
  - Writing into the HTML output using **document.write()**.
  - Writing into an HTML element, using **innerHTML**.
  - Writing into the browser console, using **console.log()**.

Example – window.alert()

```
<!DOCTYPE html>
<html>
<body>
    <script>
    window.alert(5 + 6);
    </script>
</body>
</html>
```

Example – document.write()

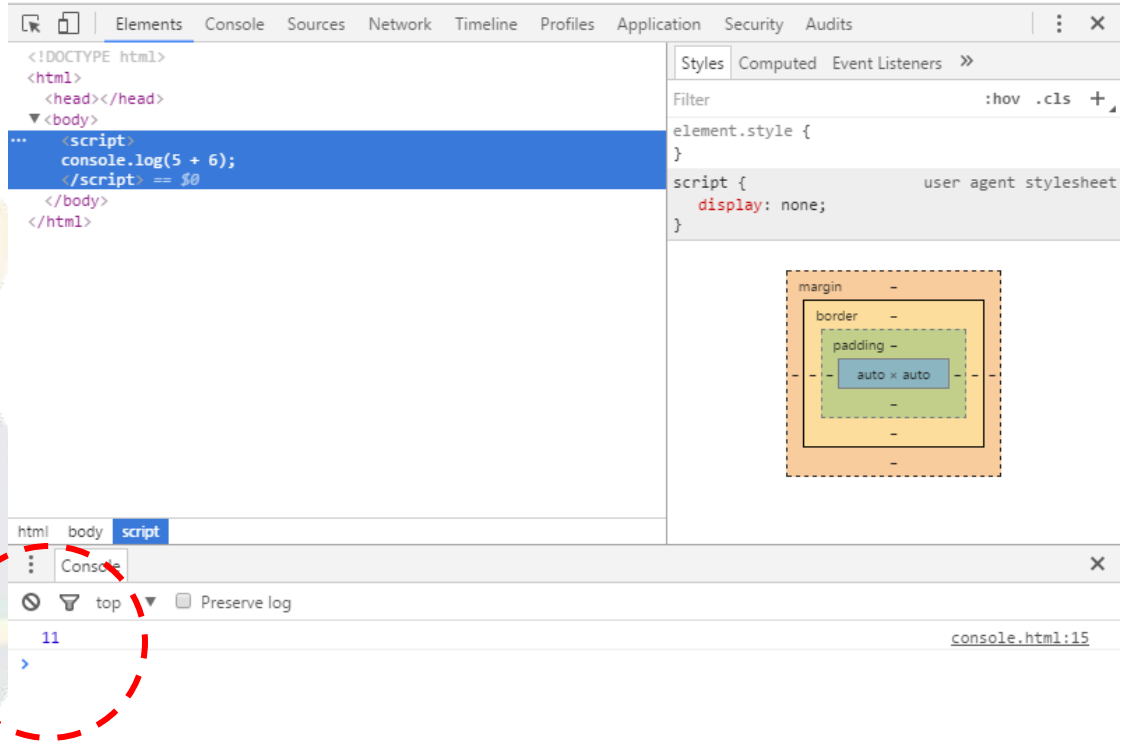
```
<!DOCTYPE html>
<html>
<body>

    <button onclick="document.write(5 + 6)">Try it</button>

</body>
</html>
```

Example - innerHTML

```
<!DOCTYPE html>
<html>
<body>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = 5 + 6;
</script>

</body>
</html>
```

Example – console.log()

```
<!DOCTYPE html>
<html>
<body>

    <script>
    console.log(5 + 6);
    </script>

</body>
</html>
```



JavaScript Syntax

- JavaScript syntax is the set of rules, how JavaScript programs are constructed.
- JavaScript Programs
  - A computer program is a list of "instructions" to be "executed" by the computer.
  - In a programming language, these program instructions are called statements.
  - JavaScript is a programming language.
  - JavaScript statements are separated by semicolons.

JavaScript Statements

- JavaScript statements are composed of:
  - Values
  - Operators
  - Expressions
  - Keywords
  - Comments
- The statements are executed, one by one, in the same order as they are written.

JavaScript Operators

ASIA PACIFIC UNIVERSITY OF TECHNOLOGY & INNOVATION

JavaScript Arithmetic Operators

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
++	Increment
--	Decrement

JavaScript Operators

ASIA PACIFIC UNIVERSITY OF TECHNOLOGY & INNOVATION

JavaScript Assignment Operators

Operator	Example	Same As
=	x = y	x = y
+=	x += y	x = x + y
-=	x -= y	x = x - y
*=	x *= y	x = x * y
/=	x /= y	x = x / y
%=	x %= y	x = x % y

JavaScript Operators

JavaScript Comparison Operators

Operator	Description
==	equal to
===	equal value and equal type
!=	not equal
!==	not equal value or not equal type
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
?	ternary operator

JavaScript Operators

JavaScript Type Operators

Operator	Description
typeof	Returns the type of a variable
instanceof	Returns true if an object is an instance of an object type

JavaScript Comments

- JavaScript comments can be used to explain JavaScript code, and to make it more readable.
- JavaScript comments can also be used to prevent execution, when testing alternative code.
- Single line comments start with **//**.
- Multi-line comments start with **/*** and end with ***/**.

JavaScript Variables

- JavaScript variables are containers for storing data values.
- In this example, x, y, and z, are variables

```
var x = 5;
var y = 6;
var z = x + y;
```

- From the example above, you can expect:
  - x stores the value 5
  - y stores the value 6
  - z stores the value 11

JavaScript Identifiers

- All JavaScript variables must be identified with unique names.
- These unique names are called identifiers.
- Identifiers can be short names (like x and y) or more descriptive names (age, sum, totalVolume).

JavaScript Identifiers

- The general rules for constructing names for variables (unique identifiers) are:
  - Names can contain letters, digits, underscores, and dollar signs.
  - Names must begin with a letter
  - Names can also begin with $ and _ (but we will not use it in this tutorial)
  - Names are case sensitive (y and Y are different variables)
  - Reserved words (like JavaScript keywords) cannot be used as names
  - JavaScript identifiers are case-sensitive.

Declaring JavaScript Variables

- Creating a variable in JavaScript is called "declaring" a variable.
- You declare a JavaScript variable with the **var** keyword:

```
var carName;
```

- After the declaration, the variable has no value. (Technically it has the value of undefined)
- To assign a value to the variable, use the equal sign:

```
carName = "Volvo";
```

- You can also assign a value to the variable when you declare it:

```
var carName = "Volvo";
```

JavaScript Data Types

- JavaScript variables can hold many data types: numbers, strings, objects and more:

```
var length = 16; // Number

var lastName = "Johnson"; // String

var x = {firstName:"John", lastName:"Doe"}; // Object
```

JavaScript Strings

- A string (or a text string) is a series of characters like "John Doe".
- Strings are written with quotes. You can use single or double quotes.
- You can use quotes inside a string, as long as they don't match the quotes surrounding the string.

```
var answer = "It's alright";            // Single quote inside double quotes
var answer = "He is called 'Johnny'";   // Single quotes inside double quotes
var answer = 'He is called "Johnny"';   // Double quotes inside single quotes
```

JavaScript Numbers

- JavaScript has only one type of numbers.
- Numbers can be written with, or without decimals:

```
var x1 = 34.00;     // Written with decimals
var x2 = 34;        // Written without decimals
```

- Extra large or extra small numbers can be written with scientific (exponential) notation:

```
var y = 123e5;      // 12300000
var z = 123e-5;     // 0.00123
```

JavaScript Numbers

- All the input that received from the user will be treated as string even though it is a number, hence it can't be calculated.
- The string input need to be parsed to return a number by using one of these global functions.

Number()	Converts an object's value to a number
parseFloat()	Parses a string and returns a floating point number
parseInt()	Parses a string and returns an integer

JavaScript Arrays

- JavaScript arrays are written with square brackets.
- Array items are separated by commas.
- The following code declares (creates) an array called cars, containing three items (car names):

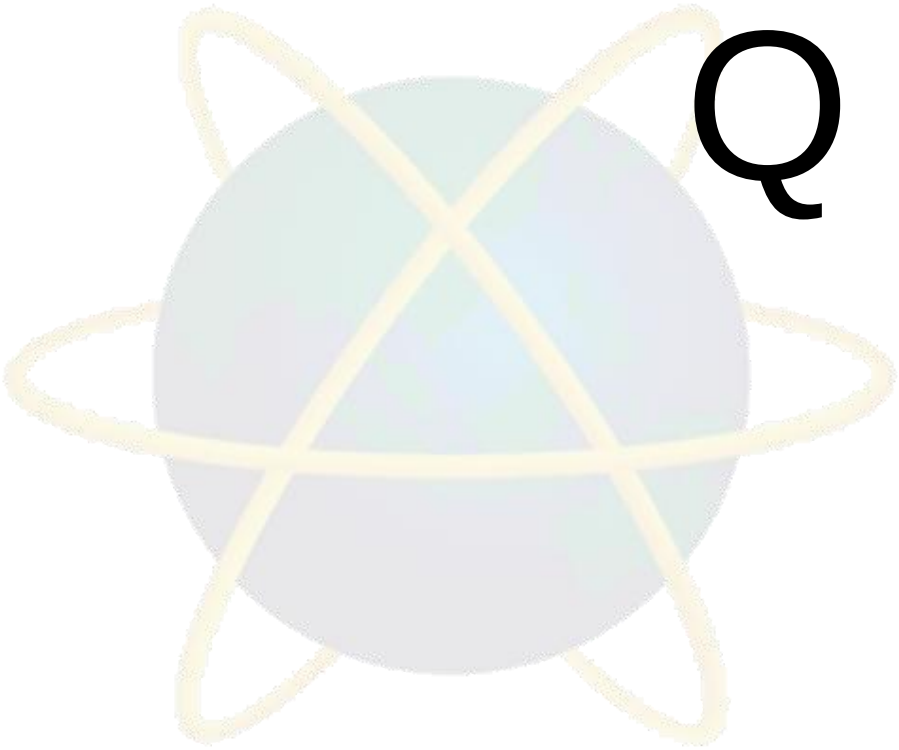
```
var cars = ["Saab", "Volvo", "BMW"];
```

JavaScript Objects

- JavaScript objects are written with curly braces.
- Object properties are written as name:value pairs, separated by commas.

```
var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};
```

- The object (person) in the example above has 4 properties: firstName, lastName, age, and eyeColor.



Q & A