Javascript 2

- 10 September 2019

Was Webstorm a software tool students have installed from previous classes, is this video guide or just a recommendation for an IDE to use or is it required to complete this class? I don't have this installed and it seems it requires a license (after 30 days) if it is not a recommendation and I will need a license for further course use where can I get the license? Are there any other software tools that will be used in this class that I should know about so that I can get them installed. It looks like Virtual Studio Code will work with some plug-ins for live web page development.

- WebStorm Setup - https://youtu.be/QziYHoZa--Y
- Notes
  - inline - in the .htlm file
  - external - external .js file
  - js script render blocking (solved in modern versions (HTTP/2)
  - case sensitive
  - “calls a constructor”
  - naming conventions
    - variables (camelCase)
    - objects and classes (first letter Cap)
    - constants (ALLCAPS)
  - does not care about whitespace
  - each statement on a separate line
  - use comments

- object containers
  - variable (let) let = aVariable;
- data types
  - numeric - (Int) var number = 1
  - string - use ‘’ or “” use escape var sampleStr = "Alan said, \"Peter is learning JavaScript\"."
  - bool var thisBool = true
  - null var thisEmptyVar = null
  - undefined var undefined;
  - symbol
- Arithmetic Operators (= + - * / ++ --)
  - PEMDAS in effect
  - “+” the only operator here that is also a string operator
    - use of “+” to add a mix of numbers and strings will result in concatenation of the variables as a string
    - use of = - * / ++ — with a set of mixed variables (sting and number) will result in strings that contain only numbers being treated automatically as a number if tried NaN will be displayed
- Conditional if Statement
  - Logical operators
    - ==
    - === (to avoid acceptance of strings)
    - >
    - <
    - <=
    - >=
    - !=
    - !== (to avoid acceptance of strings)
    - && (and)
    - || (or)

        - If a = b & c = d
          if (a == b && c == d) {}
        - If a = b or c = d (or both)
          if (a == b || c == d) {}
        - If a = b xor c = d (but not both)
          if ((a == b) || (c == d)) && ((a == b) != (c == d)) {}
    - ternary operators (for if/else statements)
    - a == b ? console.log("Match") : console.log("No Match);
    - bitwise operators

Operator	Description	Example	Same as	Result	Decimal
&	AND	5 & 1	0101 & 0001	0001	1
|	OR	5 | 1	0101 | 0001	0101	5
~	NOT	~ 5	~0101	1010	10
^	XOR	5 ^ 1	0101 ^ 0001	0100	4
<<	Zero fill left shift	5 << 1	0101 << 1	1010	10
>>	Signed right shift	5 >> 1	0101 >> 1	0010	2
>>>	Zero fill right shift	5 >>> 1	0101 >>> 1	0010	2

  - Arrays
    - var declaringAnArray
    - declaringAnArray = new Array ["same", "effect_as_the", "Shorthand", 3];
    - declaringAnArray = ["one", 2, true, false, "a few words"];
    - console.log(declaringAnArray[1]
    - var arrayString = declaringAnArray =pens.join(", ") - outputting an array

        - as string list (with ‘,’ delimitation)
    - BOM – Browser Object Model
        - DOM – Document Object Model (a property in the windows object that contains the webpage documents)
    - Functions – mini programs
        - name
        - anonymous functions
        - immediately invoked functions expressions
        - define functions before it is called in the script
        - // Regular function, called explicitly by name:

```
function multiply() {
    var result = 3 * 4;
    console.log("3 multiplied by 4 is ",
result);
}
multiply();
```

        - // Anonymous function stored in variable.
        - // Invoked by calling the variable as a function:

```
var divided = function() {
    var result = 3 / 4;
    console.log("3 divided by 4 is ", result);
}
divided();
```

        - // Immediately Invoked Function Expression.
        - // Runs as soon as the browser finds it:

```
(function() {
    var result = 12 / 0.75;
    console.log("12 divided by 0.75 is ",
result);
```

}())

- Objects
  - objects are data models that allow us to combine properties and methods for a specific data set in a structured way.
  - https://www.w3schools.com/js/js_objects.asp