**JavaScript Syntax Sheet**

.length Identifies length of var.

Set variables: var email = ”sam.comber@xaxis.com”;

Console.log(email): Will take whatever is inside the parentheses and log itto the console below the code (otherwise known as printing out).

Confirm(\_\_\_\_) Boxes used to confirm things for users – e.g. unsaved changes.

Prompt(\_\_\_) – e.g. “Where are you from?” user inputs value: var feedback = prompt(“Rate the game from 1 to 10);

Strings – e.g. “DOGS GO WOOF”

Numbers – 123456789

Booleans: true or false values – e.g. 23 > 10 is true.

If statements: evaluates a condition – e.g. if (10 > 5) {console.log(“Greater Than”)}

Else statements: Evaluates whether first condition is true, if so then passes a second condition – e.g. else {console.log(“Nope.”)}

Modulo (%): When % is placed between two numbers, the computer will divide the first number by the second, and then return the remainder of that division – e.g. use modulos in comparison where 10 % 2 === 0 evaluates to true whereas 7 % 3 === 0 evaluates to false as there is 1 left over.

Substrings: For displaying just part of a string – subsetting begins at 0. Example = console.log(“wonderful”.substring(0,3));

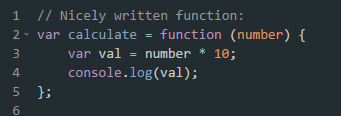
===: Check whether two values are equal.

Confirm: Requests user to confirm.

Functions: take an input, does something, and produces an output.

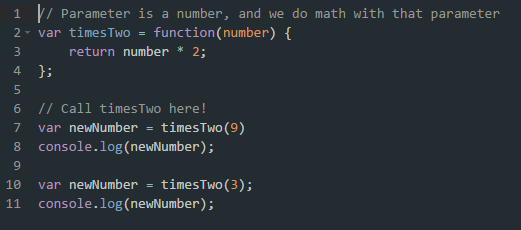
1. Declare a function using var, giving it the name sayHello.
2. Use function keyword to tell the computer you’re making a function.
3. Code in the parentheses is called a parameter, a placeholder word we give a specific value when we call the function.
4. Write reusable code between { }, with every line of code in this block ending with a ;.

To use the function, we call the function by just typing the function’s name and putting a parameter value inside parentheses after it. The computer will run the reusable code with the specific parameter value substituted into the code. Add period at end of each line in {} so computer knows where there are stopping points in the code.



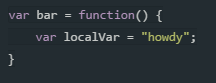
Concatenation: Joining of strings – e.g. (“HEY” + “ “ + “YOU”)

Return: Gives the programmer back the value that comes out of the function. So the function runs, and when the return keyword is used, the function will immediately stop running and return the value. When we call a function, its return value is just the result from running the function.



Functions with two parameters: var areaBox = function(length, width){return length \* width;};

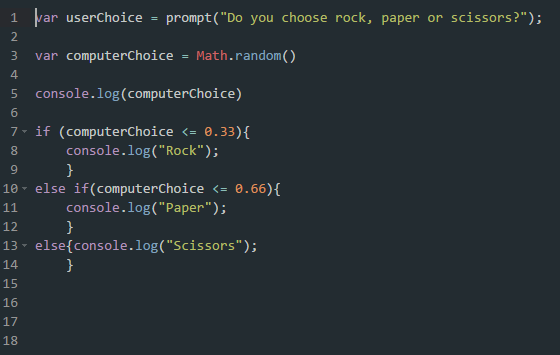
Scope: Variables defined outside a function are accessible anywhere once they have been declared. They are called global variables and their scope is global. Variables inside a function are local variables, they cannot be accessed outside of that function.



Math.random: If we declare a variable and make it equal to Math.random, that variable will equal a number between 0 and 1.

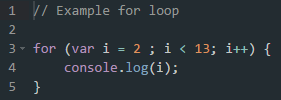
Else if: additional argument for else/if.

&&: and clause.

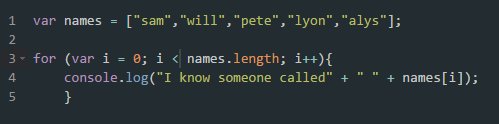


For loops: Every for loop makes use of a counting variable – e.g. first part of a loop tells the computer to start with a value of 1 for i. For loops only run when the condition is true.

1. i++ increments by 1, i— decrements by 1. I += means incrementing by any value e.g. i += 3 increments by 3s and i -= decrements by 3s.



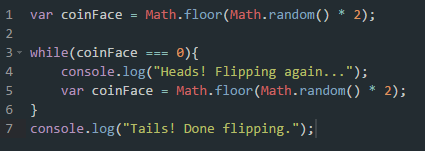
Arrays: Can store lists of data, store different data types at the same time and are ordered so the position of each piece of data is fixed = delineated by [ ] – e.g. console.log(starWars[3]).



Zero-based indexing: where we index the position of elements in an array starting from 0.

Counter variable i for For loops: The counter variable i starts at “start” and stops looping when it reaches “end”.

While loops: will re-run…



**EXAMPLE CODE:**

var age = prompt("What's your age?");

if(age >= 13){

console.log("You're old");

}

else{

console.log("You're young");

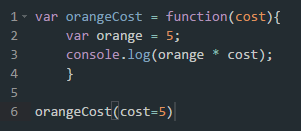
}

**FUNCTIONS:**

Var functionName = function(){

## CODE CODE CODE ##

}



Functions and IF/ELSE conditions:

