**your whole job of programming is identifying the problem and break it down into small pieces as possible. You solve the small problem and get the small solution, and combine the small solutions together to solve the big problem**

**we can connect information, we can update information, we can pull back information from the document object**

Web development = front end + back end

Web design = focus on design but need WD to fulfil it

Front end:

1) HTML: content and structure of website;

2) CSS: decorate and arrange website’s content and structure;

3) JS: make website interactive;

Primitive Data Type

1.you can not get any further down in JS;

2.There are 5 types, everything else is object, array is a special object;

1. Number; 2) String; 3) Boolean; 4) Null; 5) Undefined;
2. In JavaScript null is "nothing". It is supposed to be something that doesn't exist.

Variable

1. Function:1) Assign value赋值; 2) Store the value to recall it later储存数据，之后召回;

2. Use var to declare variable;

3. Rules of Value: case sensitive; start with letter, \_ , or $ , no spaces;

4. Data type of variable: determined by the data type of value;

Console 控制台 (In browser)

devTool: 快捷键

1.Use cmd+opt+i = to open devTools in inspect Element mode

2.Use cmd+opt+j = to open devTools and bring focus on the console

3. Use cmd+shift + c = to open deTools in inspect Element mode or toggle Inspect Element mode if the devTools are already open

4. console.dir(document)=the map (the whole content) of the website

Built-in Method

1.var computer = “scissors”; computer.charAt(0): what is the character at position 0 of computer? retrun, s

2. toUpperCase: change to uppercase

3. alert; typeof (data type);prompt

Prompt (within script)

1. Pop up a typing window;

2. Capturing user’s response;

3. Responses can be dynamic values;

4. Returns string;

eg. var question = "What is your name?”;

var myName = prompt(question);

output=user’s response;

Arithmetic operators 计算方法

operator represents some actions; operator operates values;

1. modulus returns remainder (余数): 11%5; output = 1;% percentage line

2. shorter format

3. increment b = b++; decrement a = a- -;

4. assignment operators: a += b = a + b; c /= e = c / e;

5. comparison operators:

a == b (If a is equal to b? );

a !== b (If a is not equal to b?);

a === b (If a is absolutely equal to b?);

a >= b (If a is greater or equal to b?);

a <= b (If a is less than or equal to b?);

A != B (A is not equal to B)

6. logical operators:

&& = and;

| | = or;

? (explanation mark) = flip the boolean value;

7. the return to typeof is a string. a = 4; return of typeof a is “number”;

Array

JavaScript arrays are used to store multiple values in a single variable. An array is a special variable, which can hold more than one value at a time.

1. Built-in method:

1) myArray.length: The **length** property of an array returns the length of an array (the number of array elements); the index of the last item of an array is (myArray.length-1]);

2) myArray.isArray(myArray), returns true;

3) array.push(): add an item at the end of an array;

4) array.pop(): remove the last item from the end of an array;

5) array.shift: remove the first item from the beginning of an array;

6) array.unshift: add an item at the beginning of an array;

7) array.splice(x, y): remove items from the position of index of x and finish at the position of y;

8) array.slice(x, y): pull out items from the array, the starting position is index of x, finishes at the position of y. slice a part an array and return that information back;

9) array.reverse(): reverse the order of an array;

10) array.sort(): sort item by value in alphabetical order;

11) array.sort().reverse(): chain two different built-in methods;

12) array[10]=“x”; array.fill(“y”) ; array.length=10+1; fill all the undefined places as y;

13) array.indexof(“x”); output would be the position of x in the array;

14) array.includes(“x”): if item x exits in the array;

15)The split( ) method is used to split a string into an array of substrings, and returns the new array;

2. The best way to loop through an array, is using a "for" loop;

3. new: this is a new object;

4. (literal array)var myArray=[ ]; var myArray=new Array( );

5. you can return back the second item which has an index of 2.

6. multi-dimensional arrays = nested array

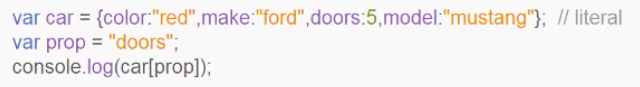
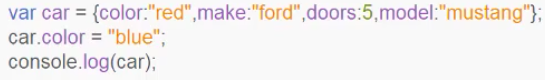
eg. set up an array which has multiple arrays:

var array = [[“cat”, “dog”, “snake”]];

array[0][2]=snake;

Object

DOM = document object model

1. object has a **paired value** (name:”value”,); array has items;
2. literal method? construction method?
3. . dot notation; [ ] bracket notation
4. Retrieve value: 1) object.name; 2) object[“name”]; 3) var prop=“x”;console.log(obj[prop]);
5. Update data of object: simply use car.color = “blue”; or car[“color”] = “blue”;
6. Delete a data of object: delete piano.name;
7. Add a data of object: piano.name = “xiaohua”;

**Loop**

**1) while loop: do while loop is a special while loop; no ideas of the times of loop; basic eg. “how many rounds do you want to run?”; practice eg.**

**for loop; know the times of loop;**

1. do while loop

do while loop is a special form of while loop;

do while loop put the condition after the action, so the action will be done at least once and then check whether fulfil the condition or not; while loop put the condition before the action;

As long as you can use while loop, you always can use do loop;

four parts:

1) initial value/ starting point:x=0;

2) action: increment= x++;

(1) if statement: action x++ should be added outside of if statement brackets;

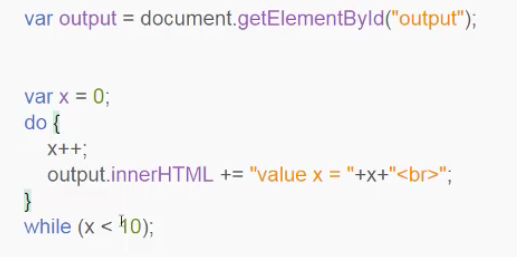
(2) if (x % 2 != 0) // if x % 2 = 0, it will also continue to loop through the function, but there is no output.

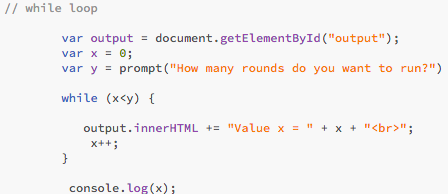
3) breaking point: condition to finish the loop: while (x<10);

4) final expression.

eg.你想跑几轮？

2. While loop and do loop are used when you don’t know how many times you are going to use the loop; if you do know the times of the loop usually for loop is used.

3. for loop: for (x=0;x<10;x++); for(initial value; condition to break out the loop; increment)



4. if (x = “red”) {continue;} skip item with string of red, and continue the loop; break:jump out from the loop.

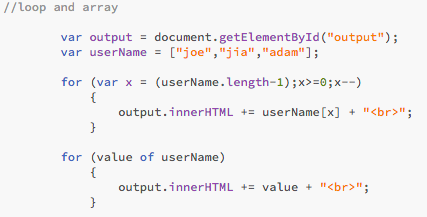
5. nested loop:

6. loop and array: var value of userNames

7. loop and object: var x in userNames

**Function**

function is used to run a piece of code over and over again simply by calling that piece of code;

1. Keyword function: use function keyword to declare that you are going to run a function;

2. Round brackets:

1) you need to write round brackets after your-named function to run the function; for example: myFunction();

2) within () you set multiple arguments or parameters;

3) within {} you can set all the statements;

3. Set a function doesn’t mean the program is going to run the code, you have to call it:

<script>

myFunction(3,5,6) // call the function; you pass the values to the function, then you will pick them up within the function as abc, and you utilise, represent and use them in the function.

function myFunction(a,b,c) {

alert (“Hello”);

} // set up the function

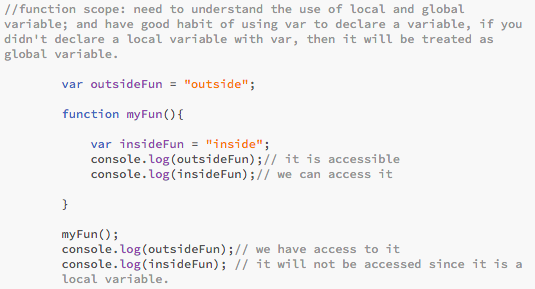
</script>

Scope function

1. it explains how variables are treated

2. local value and global value

1) local value: the variable you set within a function; once you go out of the function the local values would be deleted and you need to set the value again if you need; It means that the variable is only accessible locally within the function, and you can not access the same variable inside of another function;

2) global value:the variable you set at the top of <script> which is also outside of the function; It means that you can access it within our functions and outside our functions as long as we are not going to redeclare it locally within our functions.

3. if you don’t use var to declare the variable which is inside a function, the variable will be regarded as global variable, but you also need to call it first outside of the function.

Normal Professional expressions:

1. I am going to create and declare a variable called “insideFun()” and give it a value of “inside”;
2. you can access this variable;
3. it is accessible;
4. we have access to it;

4. no argument:

1) new solution: assign a value to the argument;

2) old solution: use conditional statement to check the typeof == “undefined”, then assign a value for it.

5. return:

1) return value shows in console not in browser;

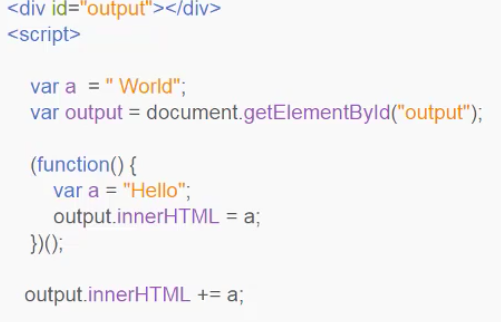
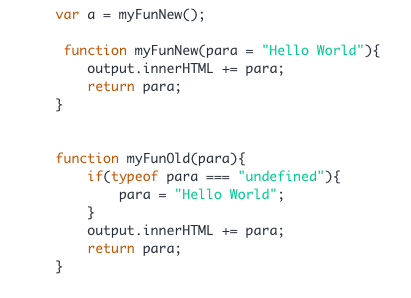
2) return value represents the “value of function”.

6. function value: var a = myFun1()

parameter=形参(formal parameter)， argument=实参(actual parameter)。

if (a>b) {return “H”;} : if the condition is true, function will break out the loop, otherwise execute other if;

6. “named function” exist any where in program;

7. “anonymous function” only exit after they‘ve been called by the program to execute it.

8. take a parameter; pass the value “” to;

9. self executing function: why?

dynamic interaction

dynamic web page content

dynamically pull back value

10. Remove duplicates

Javescript

1. Some basic Javascript can be added within the elements as an attribute.
2. onclick: holding the button for a long time. This is action is going to happen if this button gets clicked.
3. render 递交 decimal places/point 小数点位置 fraction 分数 dynamic=动态 index value =索引值
4. backslash: “My \“English\’s”\”
5. pull back, preprogram, built in

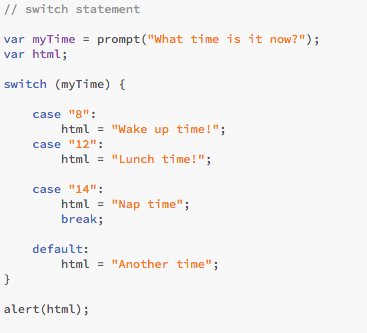
**Questions:**

Q1. why do we need to write console.log within script:

A: you can check the value

Q2. why at media query, some class style can not be recognised?

A: 因为你把520px写在了最后面，所以无论如何都会满足这个查询条件，把数值大的条件放在前面就好了。min-width顺序相反。设置不同属性的时候，从新写media query

1. Don’t understand the programming process of Rock Paper Scissors!
2. Why need to write +=

4. What’s difference between do loop and while loop?

5. what is ternary operators?三目操作符 what is the function of ternary operator?

**Javascript Logic Statement**

**If statement**

1. if 后面的条件必须在圆括号round brackets内.

**Switch Statement**

Recursion function 回归函数

Call a function within a function

Callback 回调函数

a function can be assigned to a variable

you are passing a function as an argument to another function

callback (which is a function) is passed as argument into other functions