Javascript Reference

* Javascript is a very loose, forgiving language – Easy to jump into for beginners but with plenty of nuance for advanced programmers.
* Those with experience in C, C++, Java will find many familiar language constructs

***If Statements:***

if(conditional == true){

…

}

else{

…

}

***For Loops:***

for(i = 0; i < 100; ++i){

…

}

***Functions:***

function pointlessAdd(a, b){

var c;

c = a + b;

return c;

}

***Variable Declaration***

No need to give a variable type – ALL variables are declared with *var*.

var a; //this variable has no type - undefined

var b = 5; //this variable is a number

var c = true; // this variable is a boolean

var d = “Hello!”; // this variable is a string

Javascript also handles type conversions seamlessly

b = d; //No error! b is now a string

***Declaring Arrays***

*Where variables store a single value, arrays are used to store many values. Elements are accessed using square brackets [ ]. Arrays in Javascript are actually objects, hence using new on lines 1 and 3*

1 var empty = new Array(); //creates an array of size 0

2 var empty2 = [ ]; //empty arrays have *undefined values*

*Lines 1 and 2 are functionally the same – both create empty array of size 0*

3 var empty3 = new Array(5); //creates an empty array of size 5

4 var notEmpty = [5]; //creates an array of size 1 with a value of 5

*Lines 3 and 4 are NOT the same – one creates an array of size 5, the other creates an array of size 1 with a value of 5*

5 var notEmpty2 = [1, 2, 3, 4, 5]; //array of size 5 with values 1 - 5

6

7 console.log(empty[0]) // accesses element 0 of empty - prints *undefined*

8 console.log(notEmpty[0]); //accesses element 0 of notEmpty - prints 5

9 console.log(notEmpty2[3]); //accesses element 3 of notEmpty2 - prints 4

***Declaring Objects***

Just about everything in Javascript can be an object. Objects are variables with properties, or values (name, weight, height), and methods, or functions / actions (run(), drive(), start()).

1 var empty = {}; //creates a new empty object

2 var empty2 = new Object(); //same as above

*Lines 1 and 2 are functionally the same – both create an empty object*

3 var car = {};

4 car.make = “Nissan”; //adding properties

5 car.model = “Maxima;

6 car.year = “1998”;

7

8 var car = { make: “Nissan”,

9 model: “Maxima”,

10 year: “1998”

11 };

*Lines 3 – 6 and 8 – 11 are also the same – both are ways to create an object and add elements to it*

12 car.start = function(){ … }; //adding methods

13 car.adjustMirrors = function() {…};

14 car.drive = function(){ … };

Properties and methods are accessed using the dot . operator. We will use this *frequently* in D3.

15 console.log(car.make) //prints Nissan

16 console.log(car.year) //prints 1998

17 car.start() //starts car

18 car.adjustMirrors() //adjusts car mirrors