

Javascript Scope Exercises-II

1. Determine what this Javascript code will print out (without running it):

```
x = 1;
var a = 5;
var b = 10;
var c = function(a, b, c) {
    var x = 10;
    document.write(x);
    document.write(a);
    var f = function(a, b, c) {
        b = a;
        document.write(b);
        b = c;
        var x = 5;

        f(a,b,c);
        document.write(b);
    }
    c(8,9,10);
    document.write(b);
    document.write(x);
}
```

Answer: x-->10 , a-->8, b-->8, b-->9, b-->10, x-->1

2. What is the difference between a **method** and **function**?

Answer: Functions and methods both are functions in JavaScript. A method is just a function which is a property of an object. In simple terms a function is one which is declared outside a class and a method is something which is declared inside a class and is called using instance of the class. While function are those which we can call without creating objects.

3. What does 'this' refer to when used in a Java method?

Answer: It is used to refer to a particular instance variable within that class.

4. What does 'this' refer to when used in a JavaScript method?

Answer: In general, this keyword refers to the object it belongs to. It has different values depending on where it is used. For instance, in a method, this refers to the owner object which is defined given block.

5. What does '**this**' refer to when used in a JavaScript constructor function?

Answer: the keyword this inside the constructor function points to the newly created object.

6. Assume object x is the prototype for object y in Javascript. Object x has a method f() containing keyword 'this'. When f is called by x.f(), what does 'this' refer to?

Answer: object is another widely encapsulations mechanism in JavaScript. they are easily created with object literals. they can dynamically add new properties; behave like associative arrays. Here x is inheriting the object y but during calling x.f() this refer to only the x object.

7. What is a free variable in JavaScript?

Answer: Free variables are simply the variables that are neither locally declared nor passed as parameter. We know about scope in JavaScript, We create private variables to enable encapsulation. To write Function or variables created in a function has local scope and not accessible out from that function blocks.

8. Create an object that has properties with name = "fred" and major="music" and a property that is a function that takes 2 numbers and returns the smallest of the two, or the square of the two if they are equal.

Answer

```

var object1= function NumberObject() {
    name: "fred";
    major: " music"
    function manumulatTwoNumber( n1, n2){
        if( n1<n2) return n2;
        else if (n1>n2) return n1;
        else return n1*n1;}
    }
    Eg. Object1. manumulatTwoNumber( 10, 10) //→ 100

```

9. Write Javascript code for creating three Employee objects using the **"new"** keyword and a constructor function. Employee objects have the following fields: name, salary, position.

Answer:

```

function employee( name, salary, position){
    this. name;
    this.salary= name;
    this. position= name;
}
var e1= new employee(" employee 1", 10000, "Admin");
var e2= new employee(" employee 1", 10000, "Admin");
var e3= new employee(" employee 1", 10000, "Admin");

```

10. Write a Javascript function that takes any number of input arguments and returns the product of the arguments.

Answer:

```

function sum(x1, x2... more)
{
    var total= x*y;
    for ( var i=0; i< more.lenght; i++) {
        total*= more[i];}
    return total;
}

```

11. Write an arrow function that returns the maximum of its three input arguments.

```

funcation findMax( n1, n2, n3){
    (n1, n2, n3)=> { var localMax= Math.max(n1, n2);
        retrun max(Math.max(localamx, n3);
    }
}

```