Javascript Scope Exercises

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);
}
```

R – 10889101

2. What is the difference between a method and function?
R – A method, like a function, is a set of instructions that perform a task. The difference is that a method is associated with an object, while a function is not.

3. What does 'this' refer to when used in a Java method?
R – 'this' refers to the current object in Java.

4. What does 'this' refer to when used in a JavaScript method?
R – In a method, 'this' refers to the owners object.

5. What does 'this' refer to when used in a JavaScript constructor function?
R – 'this' refers to the current 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?
R – 'this' refers to the current object, which means object y.

7. What is a free variable in JavaScript?
R – The term free variable refers to variables used in a function that are not local variables nor parameters of that function.

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.

```
R –
const obj = {
    name: 'fred',
    major: 'music',
    smallestOrSquare: function (x, y) {
        if (x === y) return x * x;
        if (x > y) return y;
        else return x;
    }
}
```

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.

```
R –
const Employee = function (name, salary, position) {
    this.name = name;
    this.salary = salary;
    this.position = position;
}

const e1 = new Employee('Ray', 15000, 'Frontend Developer');
const e2 = new Employee('Raymond', 25000, 'Backend Developer');
const e3 = new Employee('Jean', 35000, 'Software Developer');
```

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

```
R –
function multi(...args) {
    return args.reduce((a, b) => a * b);
   }
```

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

```
R –
const max = (x, y, z) => {
    return Math.max(x, y, z);
}
```