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); (print 10)
    document.write(a); (print 8)
   var f = function(a, b, c) {
        b = a;
        document.write(b); (print 8)
        b = c;
        var x = 5;
    }
f(a,b,c);
document.write(b); (print 9)
 }
 c(8,9,10);
document.write(b); (print 10)
 document.write(x); (print 1)
```

Result : 10 8 8 9 10 1

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

   In Javascript the method is a property containing a function definition of object. If we call property with () it can be access object, otherwise, without () it will be return function definition. Function is a piece of code that value can be returned.

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

     In Java this keyword is used to refer the current instance of object in the class method on which it is used. It cannot be used outside the method.

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

In Javascript this refers to the "owner" of the function. This is the global object in a function invocation. The global object is determined by the execution environment. It is the window object in a web browser. This is undefined in a function invocation in strict mode. "this" refers to the current object when used in a Javascript method.

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

"this" refers to the object the function belongs to, or the window object if the function belongs to no object.

6.What is a free variable in JavaScript?

In Javascript free variables are used locally, but defined in an enclosing scope.

7. 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.

```
function Object(name, major, num1, num2) {
                    this.name = name.toString();
                    this.major = major.toString();
                    this.property = function(num1, num2) {
                        return num1 > num2 ? num2: num1==num2? : num1*num2 ? num1; }
}
```