1. What is the difference between the following 2 statements?

* setTimeout(booyah, 2000);
* setTimeout(booyah(), 2000);

First one passes function booyah as a parameter for setTimeout(), waits 2s and then execute function booyah(). Second one, executes function booyah() beforehand and then executes setTimeout()

1. What do the following 2 alerts display (answer without running the code)?

var myfunc = function(a, x) {

return a \* x;

};

var x = myfunc(2, 3);

var y = myfunc;

alert(x);

alert(y(2,3));

alert 1 displays 6, so alert 2.

1. Write functions booyah1 and booyah2 so that in both cases below, an alert box comes up after 2 seconds that says “BOOYAH!”

* setTimeout(booyah1, 2000);
* setTimeout(booyah2(), 2000);

function booyah2(){

alert('BOOYAH!');

}

var booyah1 = booyah2();

1. What is "Unobtrusive Javascript"? What is the practical application of Unobtrusive Javascript (and the reasons for using it)?

Unobtrusive is a way to write our code having our documents (HTML, CSS, JS) clearly separated and have a cleaner approach when having every "*concern*" on its own and only referencing between files so the modifications can be done by only modifying the corresponding document.

For this we use separated files, and for referencing we use classes, ids, tags, names, etc.