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

a-setTimeout(booyah, 2000);

b-setTimeout(booyah(), 2000);

**answer**:

a-the first one is a callback function so it will call boyaah function after two seconds.

b-The second one is a direct call to the function, will call it immediately.

## 2. 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));

**answer:**

both alerts will show 6

3. 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);

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

function booyah1() {
    alert("BOOYAH! 1");
```

```
}

function booyah2() {;
    setTimeout(function() { alert("BOOYAH! 2"); }, 2000);


}
```

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

It is the technique of keeping HTML files separated from javascript files. In another way is to separate the content from behavior. Also, it means the progressive enhancement which means that the webpage will work fine without showing error if the user disable javascript in his browse or he use an old browser that doesn't support javascript.

The major benefits is to avoid the problems of traditional JavaScript programming (such as browser inconsistencies and lack of scalability)

"Unobtrusive JavaScript promotes separation of web page content into 3 different concerns: content (HTML), presentation (CSS), and behavior(JS) (ala MVC, knower, known, process of knowing"

Prof. Bruce Lester