Javascript Exercises

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

setTimeout(booyah, 2000);

setTimeout(booyah(), 2000);

Ans.

setTimeout(booyah, 2000); 🡺 the booyah function will be called after 2000ms.

setTimeout(booyah(), 2000); 🡺 it will call the booyah function immediately, rather than waiting the 2000ms.

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));

Ans.

var x = myfunc(2, 3); 🡺 6

var y = myfunc; 🡺 6

2 alerts will display 6.

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);

Ans.

function booyah1() {

alert("BOOYAH!");

}

function booyah2() {

setTimeout(function() {

alert("BOOYAH!")

}, 2000);

}

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

Unobtrusive JavaScript is a general approach to the use of [client-side](https://en.wikipedia.org/wiki/Client-side) [JavaScript](https://en.wikipedia.org/wiki/JavaScript) in [web pages](https://en.wikipedia.org/wiki/Web_page) so that if JavaScript features are partially or fully absent in a user's [web browser](https://en.wikipedia.org/wiki/Web_browser), then the user notices as little as possible any lack of the web page's JavaScript functionality.

We can write unobtrusive JavaScript code

* HTML with minimal JavaScript inside
* uses the DOM to attach and execute all JavaScript functions

It allows separation of web site into 3 major categories:

* content (HTML) - what is it?
* presentation (CSS) - how does it look?
* behavior (JavaScript) - how does it respond to user interaction?

The practical application of Unobtrusive Javascript is “[Decorate My Text](http://mumstudents.org/cs472/2021-06-BL/Labs/Lab06-decoratemytext.php)” lab. In that lab, it separates

* decoratemytext.html
* decoratemytext.css
* decoratemytext.js

The reasons for using it are the following.

* [Usability](https://en.wikipedia.org/wiki/Web_usability): An unobtrusive DOM script does not draw the attention of the user—"visitors just use it without thinking about it."
* [Graceful degradation](https://en.wikipedia.org/wiki/Graceful_degradation): Unobtrusive DOM scripts never generate error messages, in any browser, even when they fail. If features cannot be presented properly, they silently disappear.
* [Accessibility](https://en.wikipedia.org/wiki/Web_accessibility): If any script fails, the page still delivers its core functions and information via the markup, stylesheets and/or server-side scripting.
* [Separation](https://en.wikipedia.org/wiki/Separation_of_concerns): For the benefit of other and future web developers, all JavaScript code is maintained separately, without impacting other files of script, markup or code.