**Js Quest03**

Remember to git add && git commit && git push each exercise!

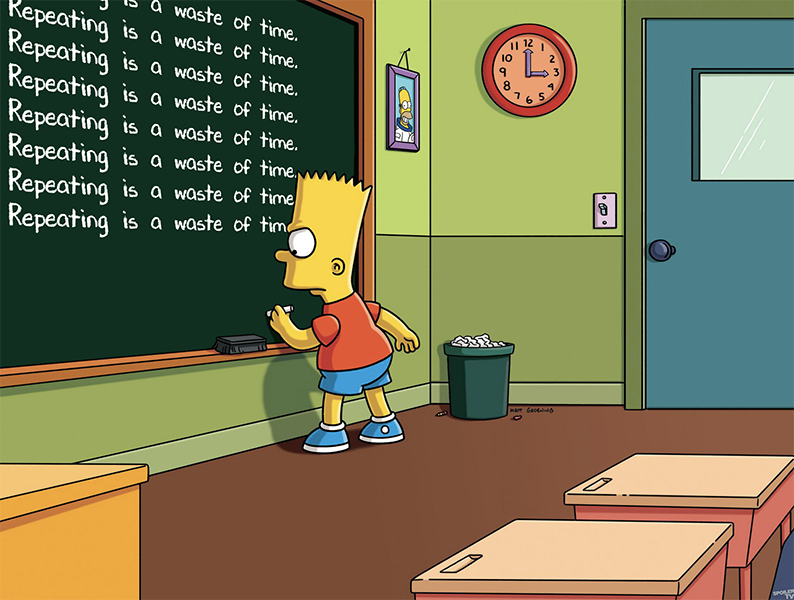
We will execute your function with our test(s), please DO NOT PROVIDE ANY TEST(S) in your file

For each exercise, you will have to create a folder and in this folder, you will have additional files that contain your work. Folder names are provided at the beginning of each exercise under submit directory and specific file names for each exercise are also provided at the beginning of each exercise under submit file(s).

**Introduction**

We've seen variable, with different types. Integer, char, string.

But let's remember, why we started to code? Create something? Automation? Both? :D



Let's loop in :D

| **Js Quest03** | **My Html Journey First While** |
| --- | --- |
| Submit directory | ex00 |
| Submit file | index.html |

**Description**

Last part of automation is loop.

Repeating is annoying? what is we could create a program for it? :)

Let's talk about our assignment:

Create an index.html file with this content:

<!DOCTYPE html>

<html>

<body>

</body>

<script>

index = 0;

while (XX) {

console.log("I want to code");

XX // <- you should increment

}

</script>

</html>

Implemente a while loop to print 100 times "I want to code". (Don't forget to increment the index ;-))

Replace/Complete the following code. (XX is what you need to replace)

To display your page you will need a static html page renderer (see annex at the bottom).

*Tip* Google access browser debugger console

| **Js Quest03** | **My Html Journey First Function** |
| --- | --- |
| Submit directory | ex01 |
| Submit file | index.html |

**Description**

Our code will start to be messy with a lot of loop and if statements.

Another concept is abstraction and how to do this? A short answer is with functions.

What is a function? A function is a body of code that returns a value. The value returned may depend on arguments provided to the function.

Let's talk about our assignment:

Create an index.html file with this content:

<!DOCTYPE html>

<html>

<body>

</body>

<script>

// function will print with console.log("my\_first\_function")

XXXXXX

XXXXXX

XXXXXX

my\_first\_function();

</script>

</html>

Replace the XX by the necessary code to create a function called my\_first\_function.

To display your page you will need a static html page renderer (see annex at the bottom).

*Tip* Google access browser debugger console

| **Js Quest03** | **My Html Journey First Function With Params** |
| --- | --- |
| Submit directory | ex02 |
| Submit file | index.html |

**Description**

We continue our journey with function. Function can be very complex.

A function is a flow, with an input and an output. In the middle it's doing something.

Let's talk about our assignment:

Create an index.html file with this content:

<!DOCTYPE html>

<html>

<body>

</body>

<script>

// function will console.log("detonation in... "+seconds\_left+" secondes.")

timer = 10;

while (XX) {

detonation\_in(timer);

XX

}

</script>

</html>

Function accepts parameters, let's send an integer to our function and print it!

Implemente a while loop to call a function detonation in...X secondes. Your loop will stop a 0. 10 included, 0 is not. (Don't forget to decrement the index ;-))

Replace/Complete the following code. (XX is what you need to replace)

To display your page you will need a static html page renderer (see annex at the bottom).

*Tip* Google access browser debugger console

| **Js Quest03** | **My Html Journey First Function With Return** |
| --- | --- |
| Submit directory | ex03 |
| Submit file | index.html |

**Description**

We continue our journey with function. We talked about input (parameters).

Let's talk about the output. (return)

Let's talk about our assignment:

Create an index.html file with this content:

<!DOCTYPE html>

<html>

<body>

</body>

<script>

// function my\_get\_seven() will return 7

console.log(my\_get\_seven());

</script>

</html>

Function returns a value, let's print it!

Implemente a function which return a number (7)

Replace/Complete the following code. (XX is what you need to replace)

To display your page you will need a static html page renderer (see annex at the bottom).

*Tip* Google access browser debugger console

| **Js Quest03** | **My Moving Box** |
| --- | --- |
| Submit directory | ex04 |
| Submit file | index.html |

**Description**

Complete an index.html file with the missing javascript code in order to move the "div" with the id my\_box to the coordinate of: bottom: 0and right 0

You cannot change the value of the html, moving the box needs to be done using javascript.

$>cat index.html

<html>

<div id="my\_box" style= "background-color: red; position: absolute; right: 70; bottom: 70; min-width: 100px; min-height: 100px"></div>

<script type="text/javascript">

// YOUR CODE

</script>

</html>

$>

To display your page you will need a static html page renderer (see annex at the bottom).

*Tips* Google Javascript document getElementById Google Javascript change css position