



Js Quest04

Subject

1 Solution

Additional Resources
(2)

Js Quest04

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

From Hollywood's movies:

"Give me a pdf, so I will connect my ssh to my RSA and I will be able to access the md5."

Most of the computer on earth are not running any Graphical User Interface (GUI) and the closest thing we have from it, is the terminal.

Control Center



Group formation



Progress



Submitted



Test review



Finished: approved



[Go To DoCode](#)



Access:

READ

WRITE



[Go To Gitea](#)



[Keep Working On This Solution](#)

Looking for a group

But more important, you can do more things with the terminal than with the GUI.

We will pass some time here, to write some scripts.

At the end of the bootcamp you will be one of us :-)



Js Quest04	My First Script
Submit directory	ex00
Submit file	my_first_script.js

Description

Write your first script, printing `Hello World!`.

Create a file `my_first_script.js`.

Add a printing function (see tips)

Example 00 (In Javascript)

```
$>node my_first_script.js
Hello World!
$>
```

Example 01 (In Python)

```
$>python my_first_script.py
Hello World!
$>
```

Example 02 (In Ruby)

No body is looking for a partner at the moment

Also working on the project

   
[iron_j](#) [jamolid](#) [toychib](#) [xolxoja](#)
[d_b](#) [o_a](#) [y_f](#)









[atoyev_u](#)

Just finished

   
[aripovaakhmet](#) [kozhe](#) [vitale_d](#)
[_b](#) [o_sa](#) [mra_d](#)

   
[peralta](#) [bennet](#) [champ](#) [musha](#)
[_r](#) [t_k](#) [eno_s](#) [rba_j](#)

   
[thomas](#) [zucker](#) [hwang_](#) [ji_j](#)
[on_b](#) [ma_r](#) [d](#)

 
[chun_k](#) [hanna_e](#)

Type

Project

Group
Size

1
Participant

```
$>ruby my_first_script.rb
Hello World!
$>
```

Tip

(In Javascript)

It will contain `console.log("Hello World!")`

Js Quest04	My Multiple Variables Multiple Type
Submit directory	ex01
Submit file	my_multiple_variables_multiple_type.js

Description

Replace/Complete the following code. Create multiple variables with (if needed) the right type.

(The XX is what you need to replace)

Function prototype (javascript)

```
XX = 34;
XX = "Luke";
XX = ', ';
```



```
console.log("Hello " + my_name + my_comma
+ " I'm " + my_age + " years old.");
```

Example 00

Review
system

Test Review (Gandalf)

Difficult

y

Initiation

Average

e

duration

n

1
Week

Project's Metadata

Project

id: 60

name: js-quest04

visible: True

Input:

Output: Hello Luke, I'm 34 years old.

Return Value: nil

Js Quest04	My First If Multiple Conditions
Submit directory	ex02
Submit file	my_first_if_multiple_conditions.js

Description

`if conditions` are linked to `else` statements and writing the correct condition can be quite complicated :D.

Your assignment is to write the correct conditions inside the if statements below in order to render the 2 print statements true!

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

Function prototype (javascript)

```
a = 10
b = 9
c = 11
d = 10
y = 9
z = 11

if (XX) {
  console.log("a is bigger than b AND
smaller than c AND equal to d");
}
if (XX) {
  console.log("z OR y are bigger than
a");
}
```

Example 00

Input:

Output: a is bigger than b AND smaller
than c AND equal to d
z OR y are bigger than a

Return Value: nil

Js Quest04	My First Function
Submit directory	ex03
Submit file	my_first_function.js

Description

The syntax is only a small part of what you need to learn to write quality code.

Software Architecture (Designing Software) is really the core part of each project and being a good engineer.

In order to "organize" your code, functions are the key. Let's dive in to functions!

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

Function prototype (javascript)

```
// Following XXXXXX will be a function  
that will print using  
console.log("my_first_function")  
XXXXXX  
XXXXXX  
XXXXXX  
  
my_first_function();
```

Example 00

Input:
Output: my_first_function

Return Value: nil

