Js Quest02

Subject   1 Solution   Additional Resources (1)

Control Center

Group formation

In Progress

Submitted

Test review

Finished: approved

Js Quest02

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)` .

🖵   Go To DoCode

Access:
READ   WRITE

◈   Go To Gitea

💬   Keep Working On This Solution

Introduction

Syntax is the very first things we need to dive in before the "creation".
We need to know how to know the alphabet and grammar before being able to write an essai.

Coding is the same. And rules are simpler. It's a language and you need to learn it.
No, you need to assimilate it!

What is a computer language?

Computer languages are systems of communication with a computer. Such languages are used to create computer code or program code, the set of instructions forming a computer program which is executed by the computer.

Blahblah, let's make it simpler: variable, loop and if-else statement.

Let's dive in.

Js Quest02	My Html Journey Variable Char
Submit directory	ex00
Submit file	index.html

Description

We've seen variable can have different `type` . Let's continue with another `type` : character.

Same as before it will need to be inside a `script` tag.

Let's talk about our assignment:

Create an index.html file with this content:

```
<!DOCTYPE html>
<html>
    <body>
    </body>
    <script>
        // Put your variable here
    </script>
</html>
```

create a variable named `letter` initialized to the value `'c'` Notice the "'", this is where a character is different from an integer. Letter are surrounding by this `'`.

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

Tip
Google access browser debugger console

Js Quest02	My Html Journey Variable String
Submit directory	ex01
Submit file	index.html

Description

We've seen variable can have different `type` . Let's continue with another `type` : string.

A string is a `word` , it's just `multiple` characters. There are defined as an `array` but we will see this later on.

Same as before it will need to be inside a `script` tag.

Let's talk about our assignment:

Create an index.html file with this content:

Type
Project

Group Size
1 Participant

Review system
Test Review (Gandalf)

Difficulty
Initiation

Average duration
1 Week

Project's Metadata

Project

id: 58

name: js-quest02

visible: True

```
<!DOCTYPE html>
<html>
    <body>
    </body>
    <script>
        // Put your variable here
    </script>
</html>
```

Create a variable named `my_string` initialized to the value `"Learning is growing"`
Notice the `'`, this is where a string is different from an integer. String are surrounding by this `'`.

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

Tip
Google access browser debugger console

Js Quest02	My Html Journey Multiple Variable Type
Submit directory	ex02
Submit file	index.html

Description

We've seen, integer, character and string. Let's combine all of them and `print` it with a console.log.

Let's talk about our assignment:

Create an index.html file with this content:

```
<!DOCTYPE html>
<html>
    <body>
    </body>
    <script>
        // Put your variable here
    </script>
</html>
```

We will create 3 variables, one my_age, one my_name and one my_comma.

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

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

`renderer` (see annex at the bottom).

Tip
Google access browser debugger console

Js Quest02	My Html Journey First Incrementation
Submit directory	ex03
Submit file	index.html

Description

Automation is the real goal of coding. How does that translate to?
A lot of loop and if statements. :-)

One very important tool is: incrementation and decrementation.

This is a way to add (or remove) 1 from a integer variable.

Syntax is: `my_variable++` (and `my_variable--` to decrement)

Let's use them inside a page.

Create an index.html file with this content:

```
<!DOCTYPE html>
<html>
    <body>
    </body>
    <script>
      // Put your variable here
    </script>
</html>
```

We will a variable my_index initialize to 0.
increment once
print
then decrement twice
print
then increment three times.
and print again

```
my_index = 0;
// replace this comment with an increment
console.log(my_index);
// replace this comment with an decrement
// replace this comment with an decrement
console.log(my_index);
// replace this comment with an increment
// replace this comment with an increment
// replace this comment with an increment
console.log(my_index);
```

Copy it inside the script tag and complete the code above (replace the comment with the right incrementation or decrementation).

Notice lines with `//` are comments.

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

Tip
Google access browser debugger console

Js Quest02	My Html Journey First If Else
Submit directory	ex04
Submit file	index.html

Description

2nd step into automation: if (and else) statements.

In computer science, conditional statements, conditional expressions and conditional constructs are features of a programming language, which perform different computations or actions depending on whether a programmer-specified boolean condition evaluates to true or false. Apart from the case of branch predication, this is always achieved by selectively altering the control flow based on some condition.

Let's talk about our assignment:

Create an index.html file with this content:

```
<!DOCTYPE html>
<html>
    <body>
    </body>
    <script>
      nbr = 10;

      if (XX) {
         console.log("nbr is greater than
20");
      }
      else {
         console.log("nbr is less than
20");
      }
    </script>
</html>
```

Replace the XX by the necessary comparaison to make it print the correct value.

Notice, `;` at the end of the line, we highly suggest you always add them :)

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

Tip
Google access browser debugger console

Js Quest02	My Html Journey First If Multiple Conditions
Submit directory	ex05
Submit file	index.html

Description

2nd step into automation: if (and else) statements part II.

`If` are usually more complicated than the previous assignment. :-)

Let's talk about our assignment:

Create an index.html file with this content:

```html
<!DOCTYPE html>
<html>
    <body>
    </body>
    <script>
      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");
      }
    </script>
</html>
```

Replace the XX by the necessary comparaison to make it print the correct value.

Notice, `;` at the end of the line, we highly suggest you always add them :)

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

Tip
Google access browser debugger console

Js Quest02	My Html Journey Change Box Color With Js
Submit directory	ex06
Submit file	index.html

Description

Updating a page in a `static` way is cool, but can we update it dynamically using Javascript? :-)

Let's take our `my_box` :

Create an index.html file with this content:

```html
<!DOCTYPE html>
<html>
    <body>
        <div id="my_box" style="height:
200px; width: 200px; border: solid 1px
black; background-color: #FF0000">
            Box
        </div>
    </body>
    <script>
        my_box =
document.getElementById("my_box");
        my_box.style.backgroundColor =
XXXXXXX;
    </script>
</html>
```

You will modify this page so the div box has a green (#00FF00) background.

Notice we are `selecting` the box with a function called: `getElementById` and we are passing the `id` of `my_box` .

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

Tip
Google access browser debugger console