**Ruby 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).

**My String Index**

* Submit directory: ex00
* Submit file: ["my\_string\_index.rb"]

Create a my\_string\_index function which takes 2 parameters (haystack and needle) and locates the first occurrence of the character needle in the string haystack and returns the position.

You can this function as: is there a L (character) in my string "helLo".

Objective is to build a loop and having a if statement when it matches the right character and returns its position.

**Function prototype** (ruby)

##

##

## QWASAR.IO -- my\_string\_index

##

##

## @param {String} param\_1

## @param {Character} param\_2

##

## @return {integer}

##

def my\_string\_index(param\_1, param\_2)

end

**Example 00**

Input: "hello" && "l"

Output:

Return Value: 2

**Example 01**

Input: "aaaaa" && "b"

Output:

Return Value: -1

**My Upcase**

* Submit directory: ex01
* Submit file: ["my\_upcase.rb"]

Create a my\_upcase function. Which takes a string as parameter and returns the uppercase version of it.

**Function prototype** (ruby)

##

##

## QWASAR.IO -- my\_upcase

##

##

## @param {String} param\_1

##

## @return {string}

##

def my\_upcase(param\_1)

end

**Example 00**

Input: "aBc"

Output:

Return Value: "ABC"

**Example 01**

Input: ""

Output:

Return Value: ""

*Tip* Google upcase string YOURCODINGLANGUAGE

**My Downcase**

* Submit directory: ex02
* Submit file: ["my\_downcase.rb"]

Create a my\_downcase function. Which takes a string as parameter and returns the lowercase version of it.

**Function prototype** (ruby)

##

##

## QWASAR.IO -- my\_downcase

##

##

## @param {String} param\_1

##

## @return {string}

##

def my\_downcase(param\_1)

end

**Example 00**

Input: "aBc"

Output:

Return Value: "abc"

**Example 01**

Input: ""

Output:

Return Value: ""

*Tip* Google downcase string YOURCODINGLANGUAGE

**My Size**

* Submit directory: ex03
* Submit file: ["my\_size.rb"]

Create a my\_size function. Which takes a string as parameter and returns the length of it.

**Function prototype** (ruby)

##

##

## QWASAR.IO -- my\_size

##

##

## @param {String} param\_1

##

## @return {integer}

##

def my\_size(param\_1)

end

**Example 00**

Input: "aBc"

Output:

Return Value: 3

**Example 01**

Input: ""

Output:

Return Value: 0

**Example 02**

Input: "AbcE Fgef1"

Output:

Return Value: 10