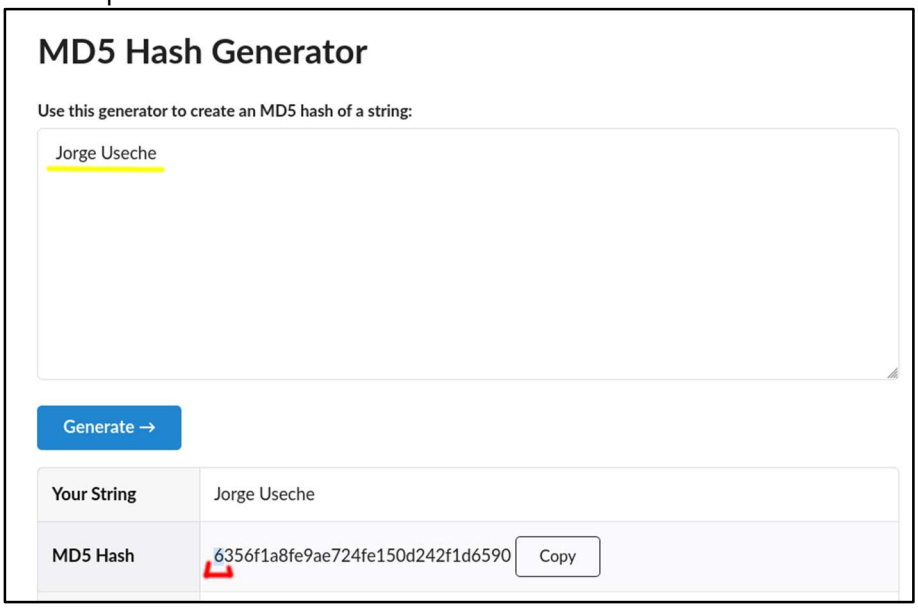
FullStack developer candidates tech challenge for Junior

Created by: Jorge Useche
Reviewers: Eduardo Arias, Camilla Arroyo
Version: 4.4

Para presentarlo debes compartir un **documento** con todos los **enlaces** a la persona de talento humano que te haya contactado, el código debe ser subido a **github, gitlab o bitbucket**, en la Kata se te preguntará por el código que realizaste.

Please open the link https://www.md5hashgenerator.com or https://codebeautify.org/md5-hash-generator and write your name, then select the first number between 1 and 9 of the MD5 Hash result. See an example below:



The number of the example is 6, but you will get any number from 1 to 9, if the Hash starts with a letter, please choose the first number, for example:

HASH -> Selected Number (n)
c20ad4d76fe97759aa27a0c99bff6710 -> 2
ce9e8dc8a961356d7624f1f463edafb5 -> 9
01a6ed5cb97b464996f7cd2c491f7431 -> 1

This number will be mencion in the next two challenges as “S”. The challenges can be resolve in javascript, typescript, java or python programming language.
In some cases you can get a hash with letters only, in this cases you can use this output as input of the hash algorithm (reapply).

1. Code challenge one

Having a list of **n** numbers with digits in range **[0, S]**, where **n <= 100**, switch all list positions in **O(n)** time.
If the input number contains a digit greater or equal than S, you will delete the digit from the number, for example with S=6, 61 becomes 1, and 6 will be deleted from the array. **The result should be printed in console/termina**l. **Please, don't use built-in sort of your language.**

Examples with S=6:
> [1, 2, 3, 4, 5, 6] -> [5, 4, 3, 2, 1]
> [10, 20, 30, 40] -> [40, 30, 20, 10]
> [6] -> []
> [66] -> []
> [65] -> [5]
> [6, 2, 1] -> [1 , 2]
> [60, 6, 5, 4, 3, 2, 7, 7, 29, 1] -> [1, 2, 2, 3, 4, 5, 0]

2. Code challenge two

Write a function that takes in a non-empty array of integers sorted in ascending order and returns a new array of the same length with the squares of the original integers also sorted in ascending order. If the output number is out of the range [0, SS] (for S=6 the range will be [0, 66]), you will delete it of the output array. **Please, don't use built-in sort of your language.**

Examples with S=6:
> {"array": [1, 2, 3, 5, 6, 8, 9]} -> [1, 4, 9, 25, 36, 64]
> {"array": [-2, -1]} -> [1, 4]
> {"array": [-6, -5, 0, 5, 6]} -> [0, 25, 25, 36, 36]
> {"array": [-10, 10]} -> []

3. Code challenge three

Given an array of positive integers representing the values of coins in your possession, write a function that returns the minimum amount of change (the minimum sum of money) that you CANNOT give change. The given coins can have any positive integer value and aren't necessarily unique (i.e., you can have multiple coins of the same value). **You can use built-in sort of your language.**

Hint 1
One approach to solve this problem is to attempt to create every single amount of change, starting at 1 and going up until you eventually can't create an amount. While this approach works, there is a better one.
Hint 2
Start by sorting the input array. Since you're trying to find the minimum amount of change that you can't create, it makes sense to consider the smallest coins first.

Hint 3

To understand the trick to this problem, consider the following example: coins = [1, 2, 4]. With this set of coins, we can create 1, 2, 3, 4, 5, 6, 7 cents worth of change. Now, if we were to add a coin of value 9 to this set, we would not be able to create 8 cents. However, if we were to add a coin of value 7, we would be able to create 8 cents, and we would also be able to create all values of change from 1 to 15. Why is this the case?

Examples:

> {"coins": [5, 7, 1, 1, 2, 3, 22]} -> 20
> {"coins": [1, 1, 1, 1, 1]} -> 6
> {"coins": [1, 5, 1, 1, 1, 10, 15, 20, 100]} -> 55