## Problem

We have seen that lists are mutable (they can be changed), and tuples are immutable (they cannot be changed).

Let's try to understand this with an example.

You are given an immutable string, and you want to make changes to it.

**Example**

>>> string = "abracadabra"

You can access an index by:

>>> print string[5]

a

What if you would like to assign a value?

>>> string[5] = 'k'

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: 'str' object does not support item assignment

How would you approach this?

* One solution is to convert the string to a list and then change the value.

**Example**

>>> string = "abracadabra"

>>> l = list(string)

>>> l[5] = 'k'

>>> string = ''.join(l)

>>> print string

abrackdabra

* Another approach is to slice the string and join it back.

Example

>>> string = string[:5] + "k" + string[6:]

>>> print string

abrackdabra

**Task**  
Read a given string, change the character at a given index and then print the modified string.

**Input Format**  
The first line contains a string, S.  
The next line contains an integer i, denoting the index location and a character c separated by a space.

**Output Format**  
Using any of the methods explained above, replace the character at index i with character c.

**Sample Input**

abracadabra

5 k

**Sample Output**

abrackdabra

## Solution

def mutate\_string(string, position, character):

    l = list(string)

    l[position] = character

    string2 = ''.join(l)

    return string2

if \_\_name\_\_ == '\_\_main\_\_':

    s = input()

    i, c = input().split()

    s\_new = mutate\_string(s, int(i), c)

    print(s\_new)