Capital Robotics Club

Scan here or visit our website to register ->

http://www.CapitalRoboticsClub.com

How to Use CodingBat.com to Practice Python

Brought to you by VEX Robotics Team - VRC 99909A Rising Phoenix

This instruction is for first time user of CodingBat.com

Step 1: Create an account on CodingBat.com

- Go to CodingBat.com's website https://codingbat.com/python *and sign up for an user account.*
- *Sign into your account to work on Python exercises. Your progress will be tracked.*
- *CodingBat.com provides coding exercises for Both Python and Java. For now, we will stay with Python.*

Step 2: How to use CodingBat.com/Python

- CodingBat.com organizes all Python coding exercises into several topics.
  - Warmup 1, 2
  - String 1, 2
  - List 1, 2
  - Logic 1, 2
- All the exercises on CodingBat.com are presented in the form of functions. Each exercise presents the challenge and provides a function name and its signature (input parameters/arguments). Students are asked to complete the function body and return the desired outcome.

Example:

https://codingbat.com/prob/p115413

String-1 > hello_name

prev | next | chance

Given a string name, e.g. "Bob", return a greeting of the form "Hello Bob!".

hello_name('Bob') → 'Hello Bob!'
hello_name('Alice') → 'Hello Alice!'
hello_name('X') → 'Hello X!'

(Below is a text editor box for you to enter your code. The function name and parameters are already provided as follows:)

```
def hello_name(name):
```

(keyword "def" defines a function called "hello_name" with only one input parameter or argument called "name" in its signature - the parentheses.)

Sample solution:

```
def hello_name(name):

        a = 'Hello '

        b = '!'

        return a + name + b
```

In the above solution, 2 variables of String type were created and assigned with a String. The final result is returned on the last line with the keyword "return".

This exercise practice string concatenation => how to connect several shorter strings into a longer one.

In Python, you can simply use the "+" operator to connect strings.

Once the function is defined, you call that function by its name and supply all the necessary parameters like this:

```
hello_name('Bob')
```

The String “Bob” in the function signature will be passed into the function as the local variable “name” when the program is executed. The function will return “Hello Bob!” as specified in the challenge.

Another solution:

```
def hello_name(name):

        return “Hello “ + name + ”!”
```