Python Bootcamp {2020}

[BASICS - WEEK 01 CHALLENGE]

Kirirom Institute of Technology

Challenge 01

Description
Create a python program that prints this message in the console: "Hello World!"

Requirements	
Program NAME	c01_hello.py
Program PATH	username/week01/c01_hello.py

USAGE
python3.7 c01_hello.py
EXPECTED OUTPUT
Hello World!

HINTS / SUGGESTIONS	
print (built-in function)	Prints to the standard output device

WARNING

This exercise is easy but to be able to pass the test you have to make sure that ALL the requirements below are respected:

- YOUR PROGRAM NAME IS CORRECT : c01_hello.py
- YOUR PROGRAM PATH IS CORRECT : username/week01/c01_hello.py
- THE OUTPUT IS CORRECT (no extra line, no extra or missing characters)

IF any of the requirements is wrong you will fail the challenge.

!NOTE: THESE RULES MUST BE RESPECTED FOR ALL THE CHALLENGES

Challenge 02

Description
Create a python program that prints the message below in the console: "Welcome to the Bootcamp Good luck and have fun!"

Requirements	
Program NAME	c02_welcome.py
Program PATH	username/week01/c02_welcome.py

USAGE
python3.7 c02_welcome.py
EXPECTED OUTPUT
Welcome to the Bootcamp Good luck and have fun!

HINTS / SUGGESTIONS	
print (built-in function)	Prints to the standard output device
\n (newline)	Escape Sequence

WARNING
⚠ For this challenge you are **FORBIDDEN** to use print more than 1 time.

Challenge 03

Description
Create a python program that prints the same message as EXPECTED OUTPUT.

Requirements	
Program NAME	c03_rules.py
Program PATH	username/week01/c03_rules.py

USAGE
python3.7 c03_rules.py
EXPECTED OUTPUT
BOOTCAMP RULES: #1 Before getting started read the instructions TWICE! #2 IF you have any QUESTIONS... ASK! #3 When you create a file: DOUBLE check the filename. #4 When you are done with a program: CHECK AND TEST IT AGAIN! #5 IF one CHARACTER is different from instructions: YOU FAIL!

HINTS / SUGGESTIONS	
print (built-in function)	Prints to the standard output device
\n (newline)	Escape Sequence
''' (triple quote)	Multi-line Print

WARNING
⚠ For this challenge you are FORBIDDEN to use print more than 1 time. ⚠ THE MAXIMUM LENGTH FOR 1 LINE IS 79 CHARACTERS (PEP 8 Style Guide).

Challenge 04

Description
Create a python program that ask the user to enter his/her name: "My name is BOT! What is your name? " and then print: "Nice to meet you <NAME>! Good luck for the Bootcamp!" !NOTE: <NAME> will be replaced by the name that the user typed.

Requirements	
Program NAME	c04_chatbot.py
Program PATH	username/week01/c04_chatbot.py

USAGE
python3.7 c04_chatbot.py
EXPECTED OUTPUT (EXAMPLE 01)
My name is BOT! What is your name? Guido van Rossum Nice to meet you Guido van Rossum! Good luck for the Bootcamp!
EXPECTED OUTPUT (EXAMPLE 02)
My name is BOT! What is your name? Kevin Nice to meet you Kevin! Good luck for the Bootcamp!

HINTS / SUGGESTIONS	
print (built-in function)	Prints to the standard output device
input (built-in function)	Allowing user input

Challenge 05

Description
Create a python program that asks the user for a number: "Enter a number: " and then print "ODD" or "EVEN" based on the given input. ODD/EVEN EXPLANATION: https://www.mathsisfun.com/numbers/even-odd.html

Requirements	
Program NAME	c05_odd_even.py
Program PATH	username/week01/c05_odd_even.py

USAGE
python3.7 c05_odd_even.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter a number: 123 ODD
EXPECTED OUTPUT (EXAMPLE 02)
Enter a number: 888 EVEN

HINTS / SUGGESTIONS	
input (built-in function)	Allowing user input
int (built-in function)	Return an integer number
if/else and conditions	IF statements and logical conditions

Challenge 06

Description
Create a python program that ask the user for input: "Enter something: " Then the program will print the first character. IF the user input is empty, the program will print: "Nothing to display."

Requirements	
Program NAME	c06_first.py
Program PATH	username/week01/c06_first.py

USAGE
python3.7 c06_first.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter something: Python P
EXPECTED OUTPUT (EXAMPLE 02)
Enter something: Nothing to display.

HINTS / SUGGESTIONS	
input (built-in function)	Allowing user input
if/else and conditions	IF statements and logical conditions
[] (square brackets)	Accessing character by index

Challenge 07

Description
Create a python program that ask the user for input: "Enter something: " Then the program will print the last character. IF the user input is empty, the program will print: "Nothing to display."

Requirements	
Program NAME	c07_last.py
Program PATH	username/week01/c07_last.py

USAGE
python3.7 c07_last.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter something: Python n
EXPECTED OUTPUT (EXAMPLE 02)
Enter something: Nothing to display.

HINTS / SUGGESTIONS	
input (built-in function)	Allowing user input
if/else and **conditions**	IF statements and logical conditions
[] (square brackets)	Accessing character by index

Challenge 08

Description
Create a python program that ask the user for input: "Enter something: " Then the program will print the string in lowercase. IF the user input is empty, the program will print: "Nothing to display."

Requirements	
Program NAME	c08_lower.py
Program PATH	username/week01/c08_lower.py

USAGE
python3.7 c08_lower.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter something: Python 101 python 101
EXPECTED OUTPUT (EXAMPLE 02)
Enter something: Nothing to display.

HINTS / SUGGESTIONS	
input (built-in function)	Allowing user input
if/else and conditions	IF statements and logical conditions
Python string method	NO MORE HINTS FOR THIS CHALLENGE! :)

Challenge 09

Description
Create a python program that ask the user for input: "Enter something: " Then the program will print the string in uppercase. IF the user input is empty, the program will print: "Nothing to display."

Requirements	
Program NAME	c09_upper.py
Program PATH	username/week01/c09_upper.py

USAGE
python3.7 c09_upper.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter something: Python 101 PYTHON 101
EXPECTED OUTPUT (EXAMPLE 02)
Enter something: Nothing to display.

HINTS / SUGGESTIONS	
input (built-in function)	Allowing user input
if/else and **conditions**	IF statements and logical conditions
Python string method	NO MORE HINTS FOR THIS CHALLENGE! :)

Challenge 10

Description
Create a python program that ask the user for input: "Enter something: " Then the program will print the string reversed. IF the user input is empty, the program will print: "Nothing to display."

Requirements	
Program NAME	c10_reversed.py
Program PATH	username/week01/c10_reversed.py

USAGE
python3.7 c10_reversed.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter something: Python 101 101 nohtyP
EXPECTED OUTPUT (EXAMPLE 02)
Enter something: Nothing to display.

HINTS / SUGGESTIONS	
input (built-in function)	Allowing user input
if/else and **conditions**	IF statements and logical conditions
Python string method	NO MORE HINTS FOR THIS CHALLENGE! :)

Challenge 11

Description
Create a python program that ask the user to enter a number: "Enter a number: " and then print "Hello World!" N times. IF the user input is a negative number or 0 you will not print anything.

Requirements	
Program NAME	c11_hello_loop.py
Program PATH	username/week01/c11_hello_loop.py

USAGE
python3.7 c11_hello_loop.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter a number: 5 Hello World! Hello World! Hello World! Hello World! Hello World!
EXPECTED OUTPUT (EXAMPLE 02)
Enter a number: 1 Hello World!
EXPECTED OUTPUT (EXAMPLE 03)
Enter a number: -1
EXPECTED OUTPUT (EXAMPLE 04)
Enter a number:

Challenge 12

Description
Create a python program that ask the user to enter a number: "Enter a number: " then print a countdown from the given number to 1 and finally display "BOOM!" (watch the examples below for more information)

Requirements	
Program NAME	c12_countdown.py
Program PATH	username/week01/c12_countdown.py

USAGE
python3.7 c12_countdown.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter a number: 3 3 2 1 BOOM!
EXPECTED OUTPUT (EXAMPLE 02)
Enter a number: 1 1 BOOM!
EXPECTED OUTPUT (EXAMPLE 03)
Enter a number: 0 BOOM!
EXPECTED OUTPUT (EXAMPLE 04)
Enter a number: -1

Challenge 13

Description
Create a python program that ask the user for input: "Enter a sentence: " Then print the sentence with replacing few terms: "OOP" ⇒ "Object Oriented Programming" "FP" ⇒ "Functional Programming" "AI" ⇒ "Artificial Intelligence"

Requirements	
Program NAME	c13_acronym.py
Program PATH	username/week01/c13_acronym.py

USAGE
python3.7 c13_acronym.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter a sentence: Python supports both OOP and FP. Python supports both Object Oriented Programming and Functional Programming.
EXPECTED OUTPUT (EXAMPLE 02)
Enter a sentence: AI is FUN! Artificial Intelligence is FUN!
EXPECTED OUTPUT (EXAMPLE 03)
Enter a sentence: Random sentence. Random sentence.

Challenge 14

Description
Create a python program that prints a RANDOM number from 1 to 6.

Requirements	
Program NAME	c14_random.py
Program PATH	username/week01/c14_random.py

USAGE
python3.7 c14_random.py
EXPECTED OUTPUT (EXAMPLE 01)
3
EXPECTED OUTPUT (EXAMPLE 02)
2
EXPECTED OUTPUT (EXAMPLE 03)
6

HINTS / SUGGESTIONS	
random (python module)	Python module to generate random number

WARNING
⚠ Every time you will run your program, one RANDOM number is printed. ⚠ YOUR PROGRAM SHOULD PRINT NUMBER FROM 1 to 6 ONLY! [1,2,3,4,5 or 6]

Challenge 15

Description
Create a python program that prints RANDOMLY 0 or 1 N times. N correspond to the number passed in the input by the user (check the examples)

Requirements	
Program NAME	c15_random_loop.py
Program PATH	username/week01/c15_random_loop.py

USAGE
python3.7 c15_random_loop.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter a number: 3 0 1 1
EXPECTED OUTPUT (EXAMPLE 02)
Enter a number: 3 1 0 1
EXPECTED OUTPUT (EXAMPLE 03)
Enter a number: 0

WARNING
⚠ YOUR PROGRAM SHOULD RANDOMLY PRINT 0 or 1 ONLY! (N times)

Challenge 16

Description
Create a python program that ask the user for a title: "Enter a title: " Then print the title as HTML TITLE Style: Your title here IF the user did not enter value the program will print "Nothing to display."

Requirements	
Program NAME	c16_html_title.py
Program PATH	username/week01/c16_html_title.py

USAGE
python3.7 c16_html_title.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter a title: Hello World Hello World
EXPECTED OUTPUT (EXAMPLE 02)
Enter a title: Welcome to Python Bootcamp Welcome to Python Bootcamp
EXPECTED OUTPUT (EXAMPLE 03)
Enter a title: Nothing to display.

Challenge 17

Description
Create a python program that ask the user for a sentence: "Enter a sentence: " as long as the user does not enter GENERATE. Once the user type GENERATE the program will display all the sentences inside HTML PARAGRAPH Style. IF the user type GENERATE without any value before: "Nothing to display."

Requirements	
Program NAME	c17_html_para.py
Program PATH	username/week01/c17_html_para.py

USAGE
python3.7 c17_html_para.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter a sentence: Python is easy Enter a sentence: GENERATE Python is easy
EXPECTED OUTPUT (EXAMPLE 02)
Enter a sentence: Hello Enter a sentence: World Enter a sentence: GENERATE Hello   World
EXPECTED OUTPUT (EXAMPLE 03)
Enter a sentence: GENERATE Nothing to display.

Challenge 18

Description
Create a python program that asks the user: "Enter something: " Then display the sentence until the first # (hashtag) character. IF no hashtag is entered, you will just print the sentence. IF no value is entered, you will not print anything.

Requirements	
Program NAME	c18_str_hashtag.py
Program PATH	username/week01/c18_str_hashtag.py

USAGE
python3.7 c18_str_hashtag.py
EXPECTED OUTPUT (EXAMPLE 01)
Enter something: Hello World!#This is a comment Hello World!
EXPECTED OUTPUT (EXAMPLE 02)
Enter something: Easy#Medium#Hard Easy
EXPECTED OUTPUT (EXAMPLE 03)
Enter something: Easy Easy
EXPECTED OUTPUT (EXAMPLE 04)
Enter something:

Challenge 19

Description
Create a python program that asks the user: "Enter a word: " Then display the two first and last characters reversed. IF the length is 2 you will display the two first characters 2 times IF the length is 1 you will display the first character 4 times IF the length is 0 you will display 4 times. !NOTE: TEST and COMPARE your program with ALL the examples below!

Requirements	
Program NAME	c19_str_mix.py
Program PATH	username/week01/c19_str_mix.py

USAGE	
python3.7 c19_str_mix.py	
EXPECTED OUTPUT (EXAMPLE 01)	EXPECTED OUTPUT (EXAMPLE 05)
Enter a word: 123456789 2198	Enter a word: 123 2132
EXPECTED OUTPUT (EXAMPLE 02)	EXPECTED OUTPUT (EXAMPLE 06)
Enter a word: ABCDEF BAFE	Enter a word: 12 1212
EXPECTED OUTPUT (EXAMPLE 03)	EXPECTED OUTPUT (EXAMPLE 07)
Enter a word: ABCD BADC	Enter a word: 1 1111
EXPECTED OUTPUT (EXAMPLE 04)	EXPECTED OUTPUT (EXAMPLE 08)
Enter a word: item time	Enter a word: 0000

Challenge 20

Description
Create a python program that asks the user: "Enter something: " Then will print every letter N time separated with - (minus symbol). N will represent the order of the letter that is display. The first letter will be uppercase. IF there are space you will skip it. IF nothing is entered, you will print "EMPTY" !NOTE: TEST and COMPARE your program with ALL the examples below!

Requirements	
Program NAME	c20_str_serial.py
Program PATH	username/week01/c20_str_serial.py

USAGE	
python3.7 c20_str_serial.py	
EXPECTED OUTPUT (EXAMPLE 01)	EXPECTED OUTPUT (EXAMPLE 04)
Enter something: abcde A-Bb-Ccc-Dddd-Eeeee	Enter something: aAaA A-Aa-Aaa-Aaaa
EXPECTED OUTPUT (EXAMPLE 02)	EXPECTED OUTPUT (EXAMPLE 05)
Enter something: a b c d A-Bb-Ccc-Dddd	Enter a word: a A
EXPECTED OUTPUT (EXAMPLE 03)	EXPECTED OUTPUT (EXAMPLE 06)
Enter something: 1234 1-22-333-4444	Enter a word: EMPTY

File Verification

Please take time to review your work and check the list below:

CHALLENGE NO	PROGRAM NAME	PROGRAM PATH
CHALLENGE 01	**c01_hello.py**	username/week01/c01_hello.py
CHALLENGE 02	**c02_welcome.py**	username/week01/c02_welcome.py
CHALLENGE 03	**c03_rules.py**	username/week01/c03_rules.py
CHALLENGE 04	**c04_chatbot.py**	username/week01/c04_chatbot.py
CHALLENGE 05	**c05_odd_even.py**	username/week01/c05_odd_even.py
CHALLENGE 06	**c06_first.py**	username/week01/c06_first.py
CHALLENGE 07	**c07_last.py**	username/week01/c07_last.py
CHALLENGE 08	**c08_lower.py**	username/week01/c08_lower.py
CHALLENGE 09	**c09_upper.py**	username/week01/c09_upper.py
CHALLENGE 10	**c10_reversed.py**	username/week01/c10_reversed.py
CHALLENGE 11	**c11_hello_loop.py**	username/week01/c11_hello_loop.py
CHALLENGE 12	**c12_countdown.py**	username/week01/c12_countdown.py
CHALLENGE 13	**c13_acronym.py**	username/week01/c13_acronym.py
CHALLENGE 14	**c14_random.py**	username/week01/c14_random.py
CHALLENGE 15	**c15_random_loop.py**	username/week01/c15_random_loop.py
CHALLENGE 16	**c16_html_title.py**	username/week01/c16_html_title.py
CHALLENGE 17	**c17_html_para.py**	username/week01/c17_html_para.py
CHALLENGE 18	**c18_str_hashtag.py**	username/week01/c18_str_hashtag.py
CHALLENGE 19	**c19_str_mix.py**	username/week01/c19_str_mix.py
CHALLENGE 20	**c20_str_serial.py**	username/week01/c20_str_serial.py