

YZV102E - Introduction to Programming for Data Science (Python)

Lab 5

27.03.2023

Res. Asst. Erhan Biçer (bicer21@itu.edu.tr)
Res. Asst. Uğur Önal (onalug@itu.edu.tr)
Res. Asst. Sümeyye Öztürk (ozturks20@itu.edu.tr)
Res. Asst. Barış Bilen (bilenb20@itu.edu.tr)
Res. Asst. Püren Tap (tap23@itu.edu.tr)

1 Exercise 1

In this part, you will complete the following tasks;

1. Define a function named *recursive_sum_of_digits_in_integer* that takes an integer as an argument. It calculates the sum of the digits in the integer in a recursive way. **Note:** The input integer will not be a negative value.
2. Get an integer named *num1* from the user.
3. Call the function *recursive_sum_of_digits_in_integer* with *num1* as an argument.
4. Print the result.
5. Test the function with the cases given in Table 1.

Table 1: Test Case for Example 1

Test Cases		
Test Case #	Input	Output
1	98756	35
2	65789	35
3	0	0
4	123	6

2 Exercise 2

In this part, you will complete the following tasks;

1. Define a function named *reverse_string* that takes a string as an argument. It reverses the given string in a recursive way.
2. Get a string from the user named *word*.
3. Call the function *reverse_string* with *word* as an argument.
4. Print the result.
5. Test the function with the cases given in Table 2.

Table 2: Test Case for Example 2

Test Cases		
Test Case #	Input	Output
1	pencil	licnep
2	knits	stink
3	exercise	esicrexe

3 Exercise 3

In this part, we will complete the following tasks;

1. Define a function named *is_polindrome* that takes a string as an argument. The function *is_polindrome* returns a Boolean value. If the string is a palindrome, it returns True; else, False. A **palindrome** is a word, number, phrase, or another sequence of characters that reads the same backward as forward, such as madam or racecar.
2. Test the function with the cases given in Table 3.

Table 3: Test Case for Example 3

Test Cases		
Test Case #	Input	Output
1	madam	True
2	Madam	True
3	butterfly	False

4 Exercise 4

In this part, you will complete the following tasks;

1. Define a function named *is_prime* that takes an integer as an argument. The function *is_prime* returns a Boolean value. If the number is prime, it returns True; else, False.
2. Test the function with the cases given in Table 4.

Table 4: Test Case for Example 4

Test Cases		
Test Case #	Input	Output
1	-4	False
2	1	False
3	2	True
4	3	True
5	4	False
6	17	True
7	21	False
8	53	True

5 Exercise 5

In this part, we will complete the following tasks;

1. Define a function named *sum_binaries* that takes two strings as input. The function *sum_binaries* calculates the sum of the binary inputs and returns the sum.
2. Test the function with the cases given in Table 5.

Table 5: Test Case for Example 5

Test Cases		
Test Case #	Input	Output
1	'11', '1'	'100'
2	'10', '10'	'100'
3	'111', '111'	'1110'
4	'1111111', '1'	'10000000'
5	'0101', '00011'	'1000'