Lab session #1

Introduction to Python

Dr Zied M'NASRI

Introduction

After reading the Python tutorial attached to the material of this lab session:

1. Open the Command Prompt cmd.exe (on your PC or on VMware Horizon) and check the python version

   ```
   c\Users\your name>python --version
   ```

2. Open the Notepad program and save the file as `fileName.py`
3. For each of the following exercises, write the code in a `fileName.py`, then run it under the Command Prompt using

   ```
   c\Users\your name\path to your file>python filename.py
   ```

Note! When you write your code, **respect the indentation rule!**

Exercise #1: Calculate the multiplication and sum of two numbers

Given two integer numbers, write a Python code to return their product only if the product is equal to or lower than 1000. Otherwise, return their sum, e.g.

Example of input and expected output:

a=40, b=20 → The output is 800

a=50, b=30 → The output is 80

Hints

- Create a function that will take two numbers as parameters
- Next, Inside a function, multiply two numbers and save their product in a product variable
- Next, use the if condition to check if the product >1000. If yes, return the product
- Otherwise, use the else block to calculate the sum of two numbers and return it.

Exercise #2: Print the Sum of a Current Number and a Previous number

Write a Python code to iterate the first 10 numbers, and in each iteration, print the sum of the current and previous number.

Example:

Current number 0 Previous number 0 Sum: 0

….

Current number 9 Previous number 8 Sum: 17

Hints

- Create a variable called previous_num and assigning it to 0
- Next, iterate through the first 10 numbers using the for loop and `range()` function
- Next, display the current number (i), the previous number, and the addition of both numbers in each iteration of a loop
- Finally, you need to update the previous_num for the next iteration. To do this, assign the value of the current number to the previous number (previous_num = i).

Exercise #3: Print characters present at an even index number

Write a Python code to accept a string from the user and display characters present at an even index number.

For example, str = "PYTHON". so your code should display ‘P’, ‘T’, ‘O’.

Hints

- Use the Python `input()` function to accept a string from a user.
- Calculate the length of the string using the `len()` function
- Next, iterate characters of a string using the for loop and `range()` function.
- Use start = 0, stop = len(s)-1, and step =2. the step is 2 because we want only even index numbers
- In each iteration of the loop, use s[i] to print characters present at the current even index number

Exercise #4: Remove first n characters from a string

Write a Python code to remove characters from a string from 0 to n and return a new string.

Example: If the input string is "PYTHON" and n=4, the output must be “ON”.

Note: n must be less than the length of the string!

Hint

Use string slicing to get a substring. For example, remove the first n characters using s[n:].

Exercise #5: Check if the first and last numbers of a list are the same

Write a code to return True if the list’s first and last numbers are the same. If the numbers are different, return False.

Example:

If the given list is l=[1, 5, 10, 1] → Output= True

If the given list is l=1,5,10, 20] → Output = False

Exercise #6: Display the numbers divisible by 5 from a list

Write a Python code to display only the numbers divisible by 5 from a list

Example: if the list l=[1,10,13,15], the output will be 10, 15

Hint

Check if the current number is divisible by using the condition (num % 5 = = 0)

Exercise #7: Find the number of occurrences of a substring in a string

Write a Python code to find how often a substring that you enter, e.g. “name”, appears in the given string.

Example:

If the input string is ”John Smith is the cousin of John Davis”

The output is: Jhon appeared 2 times

Hint

Use string method count().

Exercise #8: Print the following pattern

```
1
22
333
4444
55555
```

Hint:

Print Pattern using two imbricated for loops

Exercise 9: Check Palindrome Number

Write a Python code to check if the given number is palindrome. A palindrome number is a number that is the same after reverse.

For example, 121 is a palindrome number, whereas 123 is not.

Hint

- Reverse the given number and save it in a different variable
- Use the if condition to check if the original and reverse numbers are identical. If yes, return True.

Exercise #10: Merge two lists using the following condition

Given two lists of numbers, write a Python code to create a new list such that the latest list should contain odd numbers from the first list and even numbers from the second list.

Example

List1=[10,20,25,30,35]

List2=40,45,60,75,80]

result list=[25,35,40,60,80]

Hint

- Create an empty list named result_list
- Iterate the first list using a for loop
- In each iteration check if the current number is odd using the num % 2 != 0 formula. If the current number is odd, add it to the result list
- Now, Iterate the second list using a loop.

- In each iteration check if the current number is even using the num % 2 = = 0 formula. If the current number is even, add it to the result list.
- Print the result list.