

Exercise for Week 1

We will have some discussion on the exercise next week, but I do not expect everyone can finish it in a week, especially you all have to work at the same time. We say, for the exercise given in that week, the due day will be two weeks later.

Hopefully, it will not cause you too much pressure.

Of course, I will invite someone to share their idea or outcomes of the exercises at the next week after the assignment is given.

Homework for the lecture given at Feb 16, 2022.

1. REVIEW THE supplements given for you and other resources about python as much as you can.
2. Write a function to calculate the sum from 1 to a given number, e.g., 50, and return its outcome. Can you write the code with a given n instead of a fixed n (50)?

Other kind of problems

3. 寫一個字串的 `string_to_list` 函數:

輸入: 'I am a good man'

輸出: ['I', ' ', 'a', 'm', ' ', 'a', ' ', 'g', 'o', 'o', 'd', ' ', 'm', 'a', 'n']

4. 寫一個將 list 字元變回字串的函數。類似 `join`:

輸入: ['I', ' ', 'a', 'm', ' ', 'a', ' ', 'g', 'o', 'o', 'd', ' ', 'm', 'a', 'n']

輸出: 'I am a good man'

5. 請將 ['Company1', 'Company2', 'Company3'] 變成 ['Company_1', 'Company_2', 'Company_3'].

(這個題目很實用,許多時候我們拿到的資料會有空格,但變數命名不允許空格,可以用這個方法解決!)

For more experienced programmer with python

6. Find an alternative way to write a Fibonacci sequence function `calc_fib(n)` given in Chapter 1. The one given in the Chapter 1 causes too much calculation burden when the n is large (you may check if the outcome is returned at once. This exercise ask you to give some fixed n and return with its sequence; the second object is to use some other way to rewrite the code with better calculation speed.
7. To show all the prime numbers existed between 1 to 10. (A prime number is a positive natural number that can be divided by itself and 1 only)
8. Can the problem #4 be extended to some program that is given by fixed number N such that the program can show all the prime numbers in between?