**len(s)**

Return the length (the number of items) of an object. The argument may be a sequence (such as a string, bytes, tuple, list, or range) or a collection (such as a dictionary, set, or frozen set).

返回對象的長度（項目數）。該參數可以是一個序列（如字符串，字節，元組，列表或範圍）或集合（如字典，集合，或冷凍組）。

**range(start, stop[, step])**

Rather than being a function, range is actually an immutable sequence type, as documented in Ranges and Sequence Types — list, tuple, range.

而不是一個功能，範圍其實是一個不可變的序列類型，如記錄在範圍和序列類型 - 列表，元組的範圍。

**int(x=0)**

**int(x, base=10)**

**Convert a number or string x to an integer,** or return 0 if no arguments are given. If x is a number, return x.\_\_int\_\_(). For floating point numbers, this truncates towards zero.

If x is not a number or if base is given, then x must be a string, bytes, or bytearray instance representing an integer literal in radix base. Optionally, the literal can be preceded by + or - (with no space in between) and surrounded by whitespace. A base-n literal consists of the digits 0 to n-1, with a to z (or A to Z) having values 10 to 35. The default base is 10. The allowed values are 0 and 2-36. Base-2, -8, and -16 literals can be optionally prefixed with 0b/0B, 0o/0O, or 0x/0X, as with integer literals in code. Base 0 means to interpret exactly as a code literal, so that the actual base is 2, 8, 10, or 16, and so that int('010', 0) is not legal, while int('010') is, as well as int('010', 8).

The integer type is described in Numeric Types — int, float, complex.

Changed in version 3.4: If base is not an instance of int and the base object has a base.\_\_index\_\_ method, that method is called to obtain an integer for the base. Previous versions used base.\_\_int\_\_ instead of base.\_\_index\_\_.

**sys.exit([arg]) (需import sys)**

Exit from Python. This is implemented by raising the SystemExit exception, so cleanup actions specified by finally clauses of try statements are honored, and it is possible to intercept the exit attempt at an outer level.

The optional argument arg can be an integer giving the exit status (defaulting to zero), or another type of object. If it is an integer, zero is considered “successful termination” and any nonzero value is considered “abnormal termination” by shells and the like. Most systems require it to be in the range 0-127, and produce undefined results otherwise. Some systems have a convention for assigning specific meanings to specific exit codes, but these are generally underdeveloped; Unix programs generally use 2 for command line syntax errors and 1 for all other kind of errors. If another type of object is passed, None is equivalent to passing zero, and any other object is printed to stderr and results in an exit code of 1. In particular, sys.exit("some error message") is a quick way to exit a program when an error occurs.

Since exit() ultimately “only” raises an exception, it will only exit the process when called from the main thread, and the exception is not intercepted.

sys.exit（[參數]）

退出了Python。這是通過提高SystemExit例外實施，由try語句的finally子句中指定，以便清理行動很榮幸，這是可能的攔截企圖出口在外部層面。

可選的參數arg可以是一個整數，退出狀態（默認為零），或其他類型的對象。如果它是一個整數，零被認為是“成功的終止”和任何非零值被認為是“異常終止”由貝殼等。大多數系統都要求它在範圍0-127，並產生不確定的結果並非如此。一些系統分配特定的含義具體的退出代碼約定，但這些通常是不發達; Unix程序通常使用2個命令行的語法錯誤，1為所有其他類型的錯誤。如果其他類型的對象傳遞，無相當於傳遞零，任何其他對象被打印到標準錯誤，並導致在一特別的退出代碼，sys.exit（“一些錯誤信息”）是一種快速的方法來當發生錯誤時退出程序。

由於exit（）的最終“只”引發了異常，只會退出進程從主線程調用時，異常不攔截。

**str(object='')**

**str(object=b'', encoding='utf-8', errors='strict')**

Return a str version of object. See str() for details.

str is the built-in string class. For general information about strings, see Text Sequence Type — str.

**Python注意事項:**

i = 1

………………..

a = input("請設定要輸入的資料項有幾項: ")

………………..

while i < int(a):

if next\_step is '1':

i += 1

b = str(i)

text\_data\_next = input("請輸入第"+ b + "筆資料: ")

data\_set.append(text\_data\_next)

假設目前有3筆資料輸入，當while i < int(a):

結果:

1.

2.

3.

但如果while i <= int(a):

結果:

1.

2.

3.

4.

**如果判別運算符多加一個 “=”，會變成要多輸入一筆，要注意!**

"""

應有的功能:

指定set, list理面有多少資料項可以擺

可以決定是否要跳到下一筆資料輸入；或者留在原資料項繼續輸入；或者是要結束這項輸入作業、離開應用程式

用這個功能取代原先open資料後無法進行文字比對分析的缺憾，讓所有的資料一開始就跑進set或list中，但到時要記得把輸入的資料全部匯入指定的文字檔案裡

"""

**基礎程式碼**

import sys

#====================================================================data\_set = []

i = 1

#====================================================================a = input("請設定要輸入的資料項有幾項: ")

#====================================================================while i < int(a):

text\_data = input("請輸入資料: ")

data\_set.append(text\_data)

next\_step = input("接下來要繼續的動作為: (1)輸入下一筆資料 ；(2)繼續修改原本資料； (3)結束應用程式")

if next\_step is '1':

i += 1

b = str(i)

text\_data\_next = input("請輸入第"+ b + "筆資料: ")

data\_set.append(text\_data\_next)

elif next\_step is '2':

continue

elif next\_step is '3':

sys.exit()

#====================================================================for x in data\_set:

print(x)

**完整程式碼**

import sys

#====================================================================

data\_set = []

i = 1

#====================================================================

a = input("請設定要輸入的資料項有幾項: ")

#====================================================================

while i < int(a):

text\_data = input("請輸入資料: ")

data\_set.append(text\_data)

next\_step = input("接下來要繼續的動作為: (1)輸入下一筆資料 ；(2)繼續修改原本資料； (3)結束應用程式")

if next\_step is '1':

i += 1

b = str(i)

text\_data\_next = input("請輸入第"+ b + "筆資料: ")

data\_set.append(text\_data\_next)

elif next\_step is '2':

continue

elif next\_step is '3':

sys.exit()

#====================================================================

new\_file\_name = input("請輸入檔案格式與名稱: ")

with open("C://Users//Eric//PycharmProjects//data searching//save\_space//"+new\_file\_name, 'xt', encoding= "utf8", newline = '') as f:

for x in data\_set:

#將資料從set中一筆一筆寫入檔案裡

with open("C://Users//Eric//PycharmProjects//data searching//save\_space//"+new\_file\_name, 'at', encoding= "utf8", newline = '') as f:

f.write(x)