香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

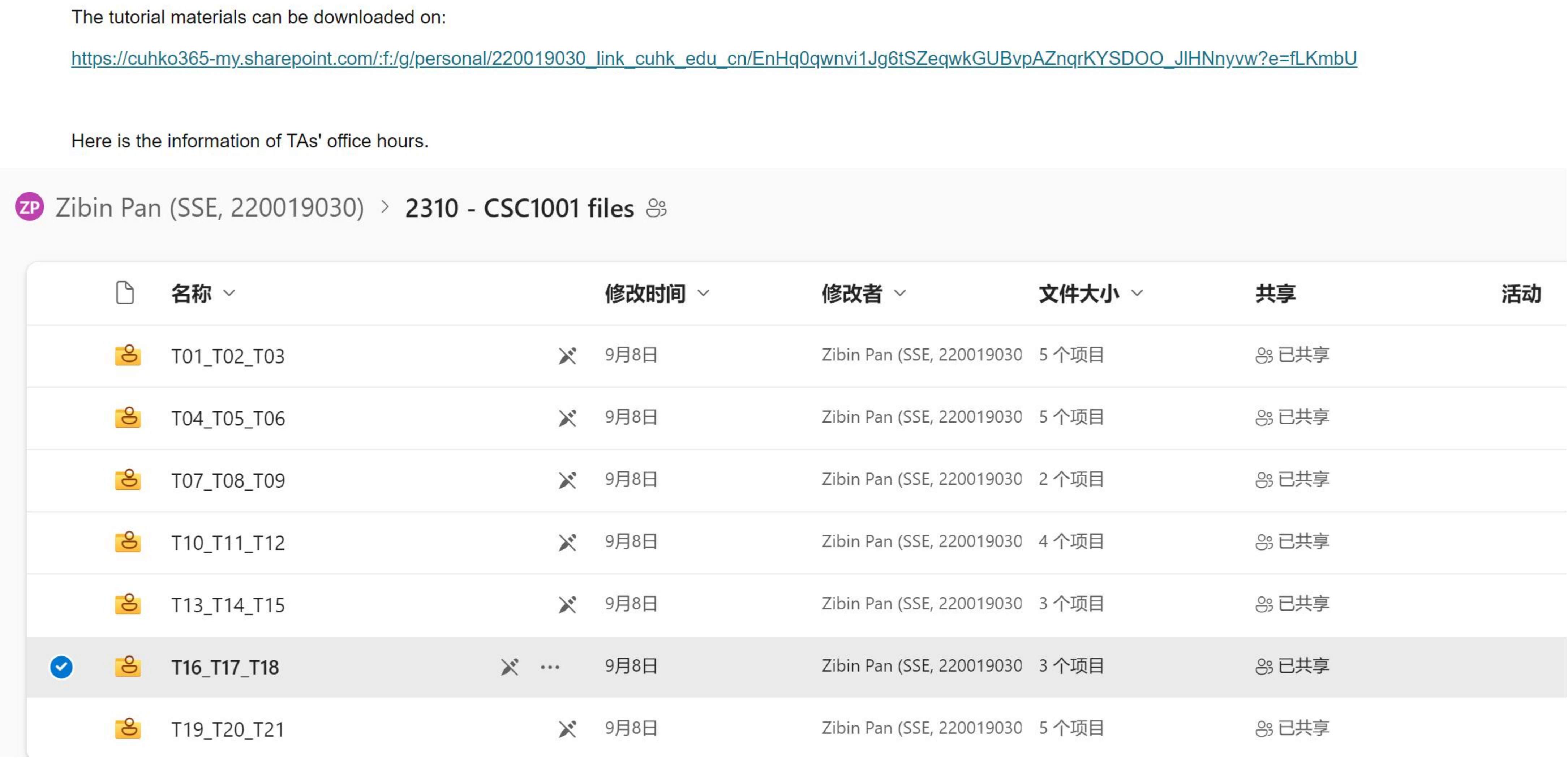
INTRODUCTION TO COMPUTER SCIENCE: PROGRAMMING METHODOLOGY

TUTORIAL 5

FUNCTIONS

朱翔宇
220019187@link.cuhk.edu.cn
Tutorial: 6-9pm Wed TD209
Office hour: 8-9am Wed Zhixin203B

- Download this slides and codes at:



The tutorial materials can be downloaded on:

https://cuhko365-my.sharepoint.com/:f:/g/personal/220019030_link_cuhk_edu_cn/EnHq0qwnvi1Jg6tSZeqwkGUBvpAZnqrKYSDOO_JIHNnyvw?e=fLKmbU

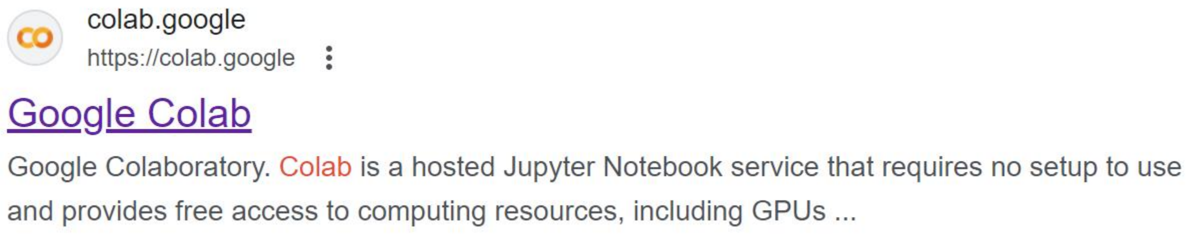
Here is the information of TAs' office hours.

ZP Zibin Pan (SSE, 220019030) > 2310 - CSC1001 files

名称	修改时间	修改者	文件大小	共享	活动
T01_T02_T03	9月8日	Zibin Pan (SSE, 220019030	5 个项目	已共享	
T04_T05_T06	9月8日	Zibin Pan (SSE, 220019030	5 个项目	已共享	
T07_T08_T09	9月8日	Zibin Pan (SSE, 220019030	2 个项目	已共享	
T10_T11_T12	9月8日	Zibin Pan (SSE, 220019030	4 个项目	已共享	
T13_T14_T15	9月8日	Zibin Pan (SSE, 220019030	3 个项目	已共享	
T16_T17_T18	9月8日	Zibin Pan (SSE, 220019030	3 个项目	已共享	
T19_T20_T21	9月8日	Zibin Pan (SSE, 220019030	5 个项目	已共享	

How to use .ipynb file?

Online: Google Colab

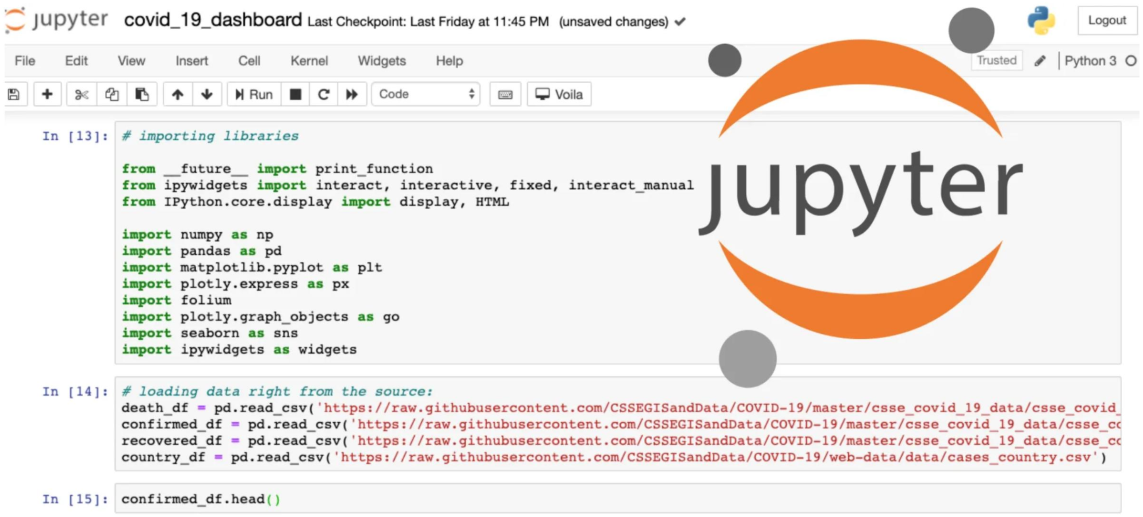


colab.google
https://colab.google

Google Colab

Google Colaboratory. Colab is a hosted Jupyter Notebook service that requires no setup to use and provides free access to computing resources, including GPUs ...

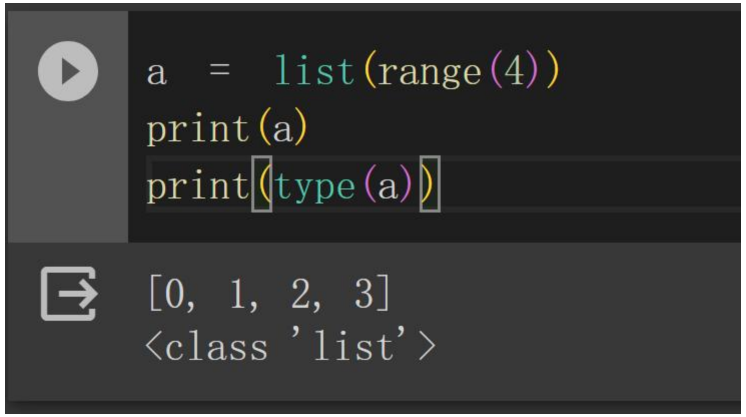
Offline: Jupyter Notebook(in Anaconda3)



```
In [13]: # importing libraries

         from __future__ import print_function
         from ipywidgets import interact, interactive, fixed, interact_manual
         from IPython.core.display import display, HTML

         import numpy as np
         import pandas as pd
         import matplotlib.pyplot as plt
         import plotly.express as px
         import folium
         import plotly.graph_objects as go
         import seaborn as sns
         import ipywidgets as widgets

In [14]: # loading data right from the source:
         death_df = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_
         confirmed_df = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_cc
         recovered_df = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_cc
         country_df = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/web-data/data/cases_country.csv')

In [15]: confirmed_df.head()
```

What to do in the tutorial .ipynb file?

- Print value
- Print type



```
a = list(range(4))
print(a)
print(type(a))
```

```
[0, 1, 2, 3]
<class ’list’>
```

- Try different inputs, modify codes and check output

What is function?

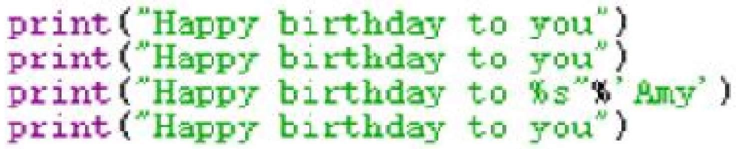
- Function is a package of code used to realize some specific functionalities or output some results of calculation.

◆ Elevator-
Functionality: lift a number of people up to a certain floor.

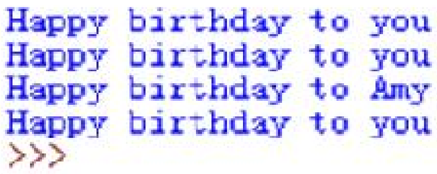
◆ Washing machine-
Input: dirty clothes
Ouput: clean clothes

When to use function?-I

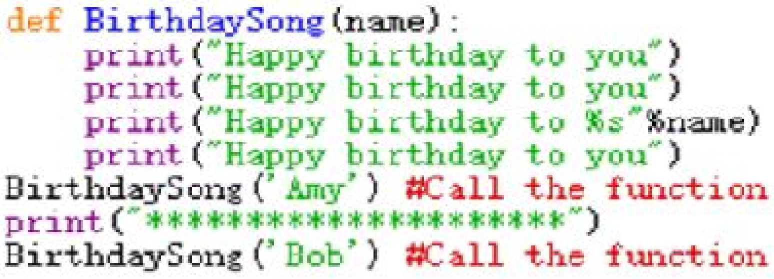
- Motivations: 1. Repeatedly use the code, make it more efficient.
  2. Realize specific functionalities.
  3. Make communication between programmers more convenient.
  4. Make the logic in your code more clear.
  …
- Example①:



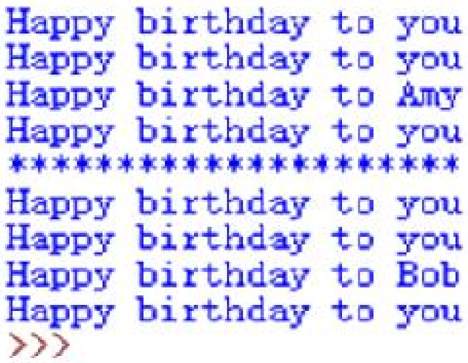
```
print("Happy birthday to you")
print("Happy birthday to you")
print("Happy birthday to %s"%'Amy')
print("Happy birthday to you")
```



```
Happy birthday to you
Happy birthday to you
Happy birthday to Amy
Happy birthday to you
>>>
```



```
def BirthdaySong(name):
    print("Happy birthday to you")
    print("Happy birthday to you")
    print("Happy birthday to %s"%name)
    print("Happy birthday to you")
BirthdaySong('Amy') #Call the function
print("********************")
BirthdaySong('Bob') #Call the function
```

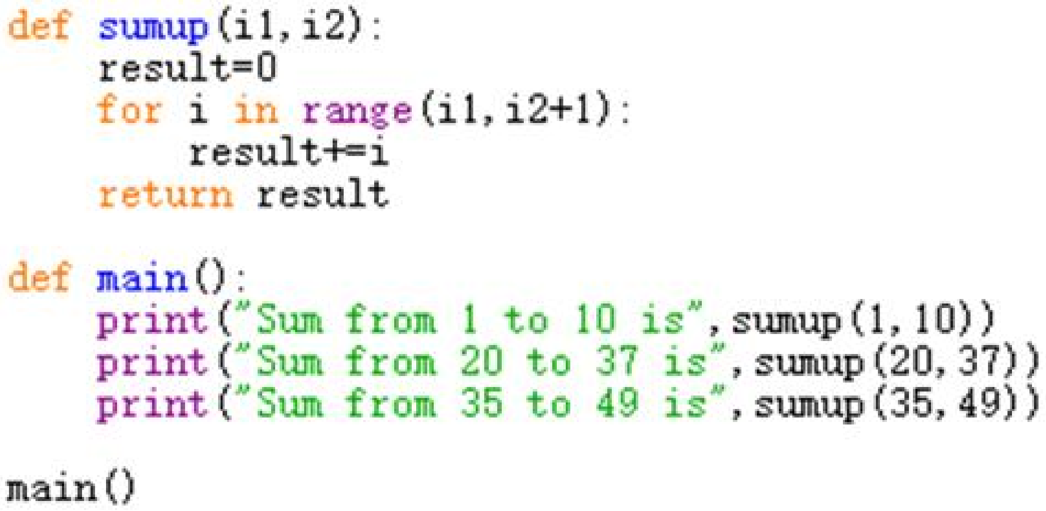


```
Happy birthday to you
Happy birthday to you
Happy birthday to Amy
Happy birthday to you
********************
Happy birthday to you
Happy birthday to you
Happy birthday to Bob
Happy birthday to you
>>>
```

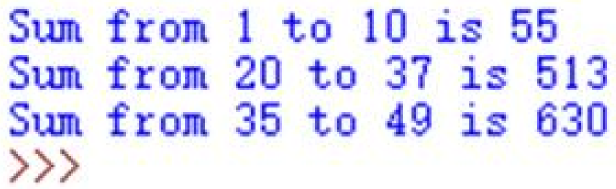
We can repeat the function body for many times.

When to use function?-II

- Example②:



```
def sumup(i1,i2):
    result=0
    for i in range(i1,i2+1):
        result+=i
    return result

def main():
    print("Sum from 1 to 10 is",sumup(1,10))
    print("Sum from 20 to 37 is",sumup(20,37))
    print("Sum from 35 to 49 is",sumup(35,49))

main()
```



```
Sum from 1 to 10 is 55
Sum from 20 to 37 is 513
Sum from 35 to 49 is 630
>>>
```

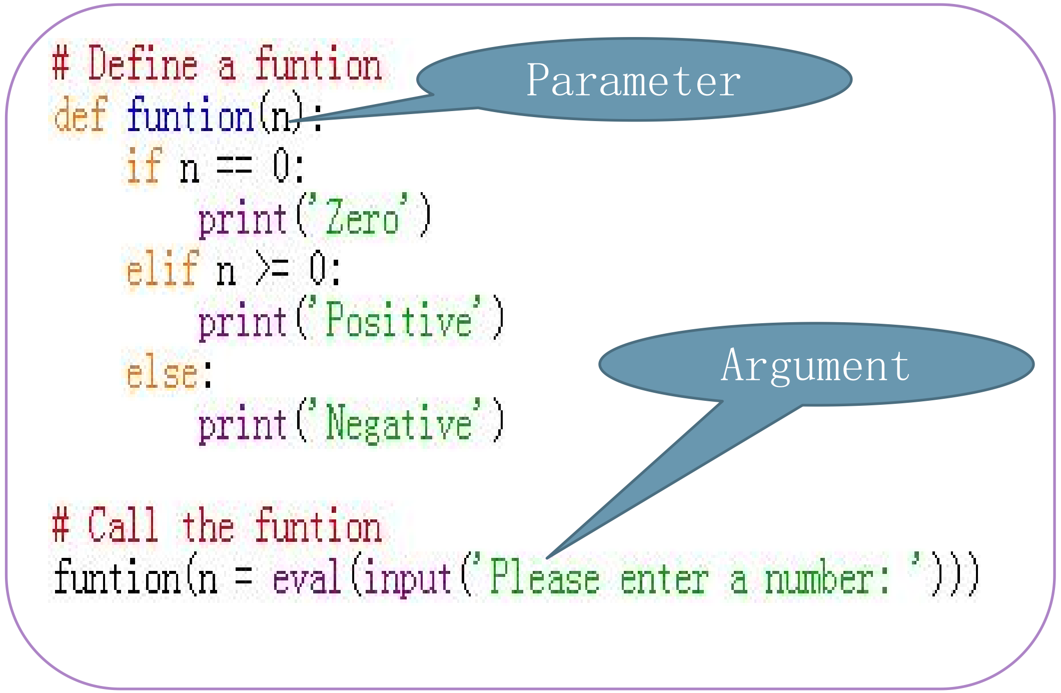
We can use functions to do calculation by using return key words.

Function Definition-I

Program

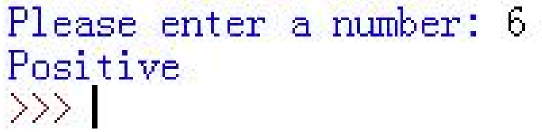
Output

Define a function:



```
# Define a funtion
def funtion(n):
    if n == 0:
        print('Zero')
    elif n >= 0:
        print('Positive')
    else:
        print('Negative')

# Call the funtion
funtion(n = eval(input('Please enter a number: ')))
```

Call/Invoke a function:



```
Please enter a number: 6
Positive
>>> |
```

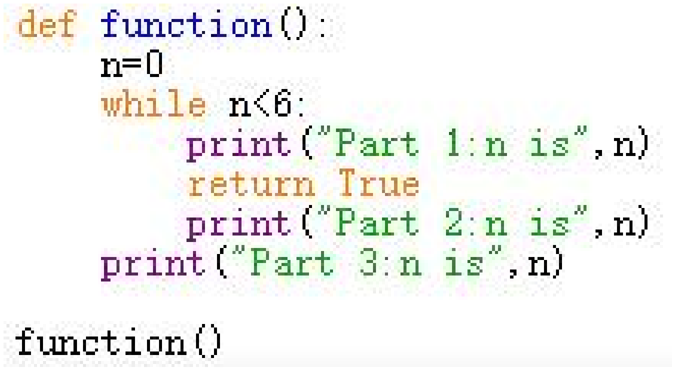
An **argument** is a value we pass into the function as its **input** when we **call** the function

Function Definition-II

- The **return** keyword-I *“A gift from function”*

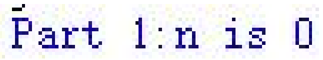
Return values

- ❑ A fruitful function is one that produces a result(or return value)
- ❑ The return statement ends the function execution and ‘sends back’ the result of the function



```python
def function():
    n=0
    while n<6:
        print("Part 1:n is",n)
        return True
        print("Part 2:n is",n)
    print("Part 3:n is",n)

function()
```

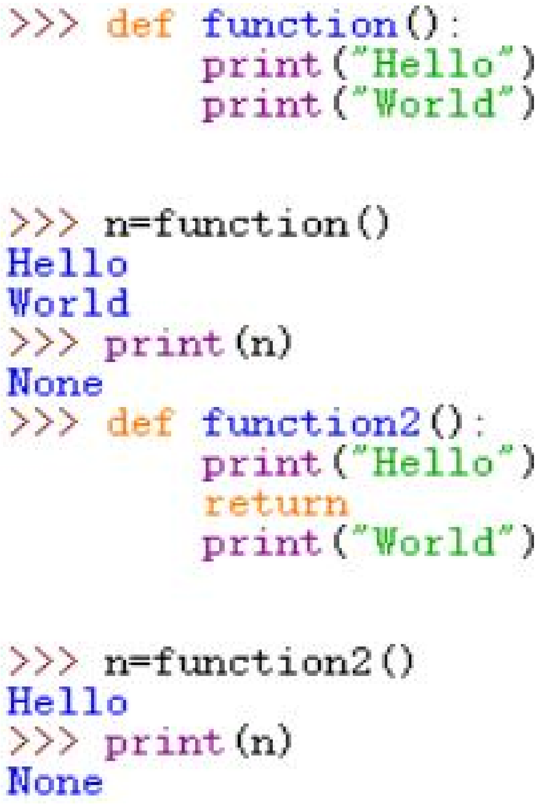
Output



```
Part 1:n is 0
```

Function Definition-III

- The **return** keyword-II



```
>>> def function():
        print("Hello")
        print("World")


>>> n=function()
Hello
World
>>> print(n)
None
>>> def function2():
        print("Hello")
        return
        print("World")


>>> n=function2()
Hello
>>> print(n)
None
```

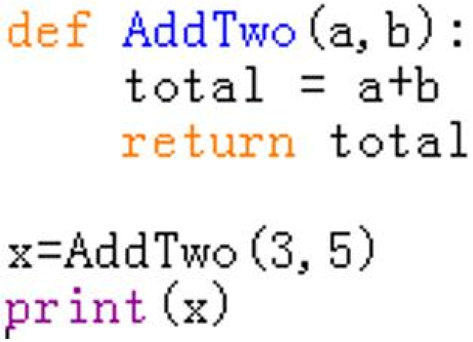
Void functions

- ❑ When a function does not return a value, it is called a “void” function

- ❑ When a function has no return statement, it will return None

Function Definition-IV

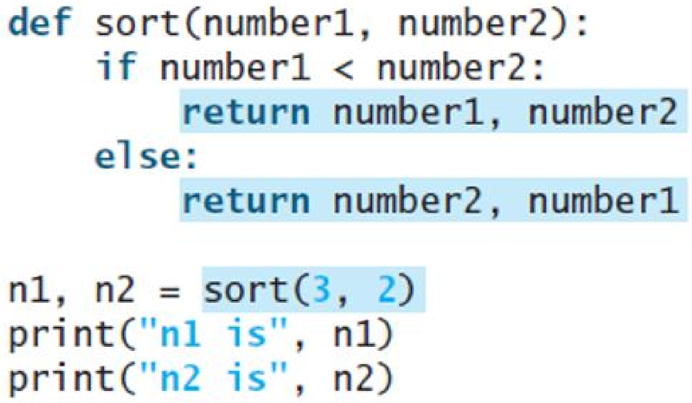
Multiple parameters/arguments



```
def AddTwo(a,b):
    total = a+b
    return total

x=AddTwo(3,5)
print(x)
```

- Match the number and order of arguments and parameters

Return multiple values

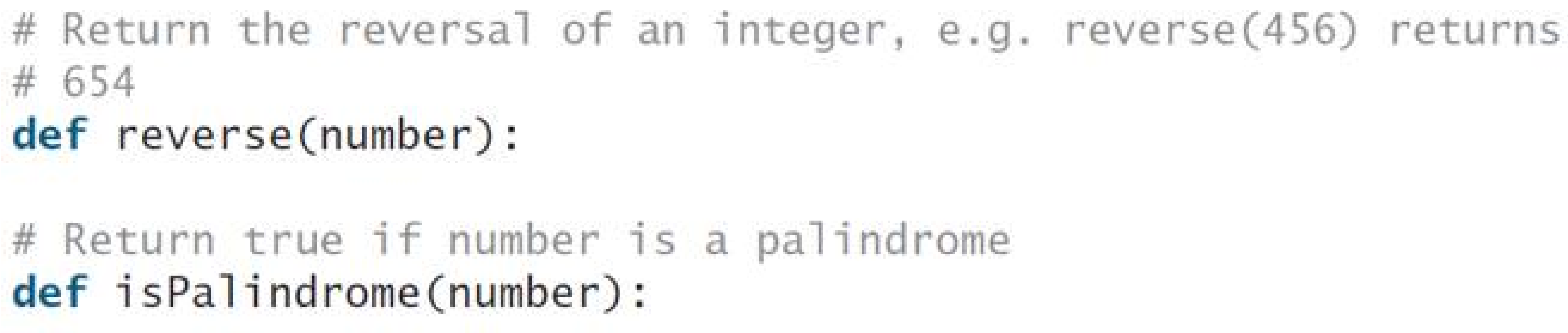


```
def sort(number1, number2):
    if number1 < number2:
        return number1, number2
    else:
        return number2, number1

n1, n2 = sort(3, 2)
print("n1 is", n1)
print("n2 is", n2)
```

- When it is invoked, you need to pass the returned values in a simultaneous assignment

Q1: Palindrome Integers

- Write the functions with the following headers:

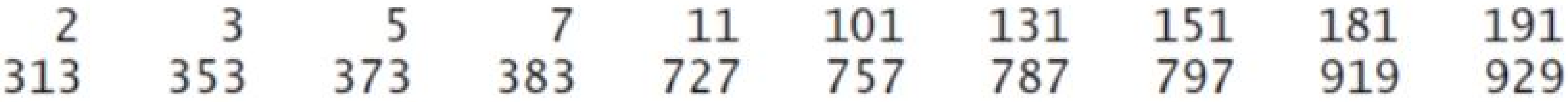


```
# Return the reversal of an integer, e.g. reverse(456) returns
# 654
def reverse(number):

# Return true if number is a palindrome
def isPalindrome(number):
```

Use the reverse function to implement isPalindrome. A number is a palindrome if its reversal is the same as itself. Write a test program that prompts the user to enter an integer and reports whether the integer is a palindrome.

Q2: Palindromic primes

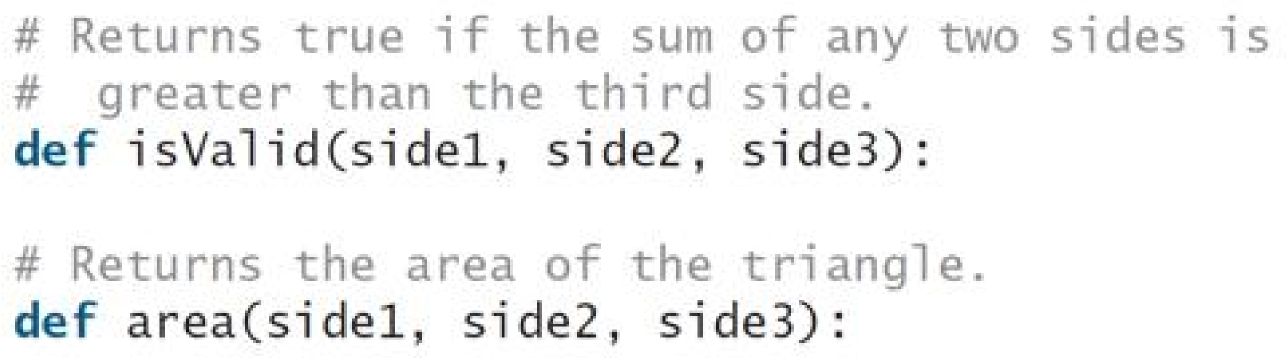
- A palindromic prime is a prime number that is also palindromic. For example, 131 is a prime and also a palindromic prime, as are 313 and 757. Write a program that displays the first 100 palindromic prime numbers. Display 10 numbers per line and align the numbers properly, as follow:



```
     2     3     5     7    11   101   131   151   181   191
   313   353   373   383   727   757   787   797   919   929
```

Q3: Mytriangle module

- Create a module named MyTriangle that contains the following two functions:

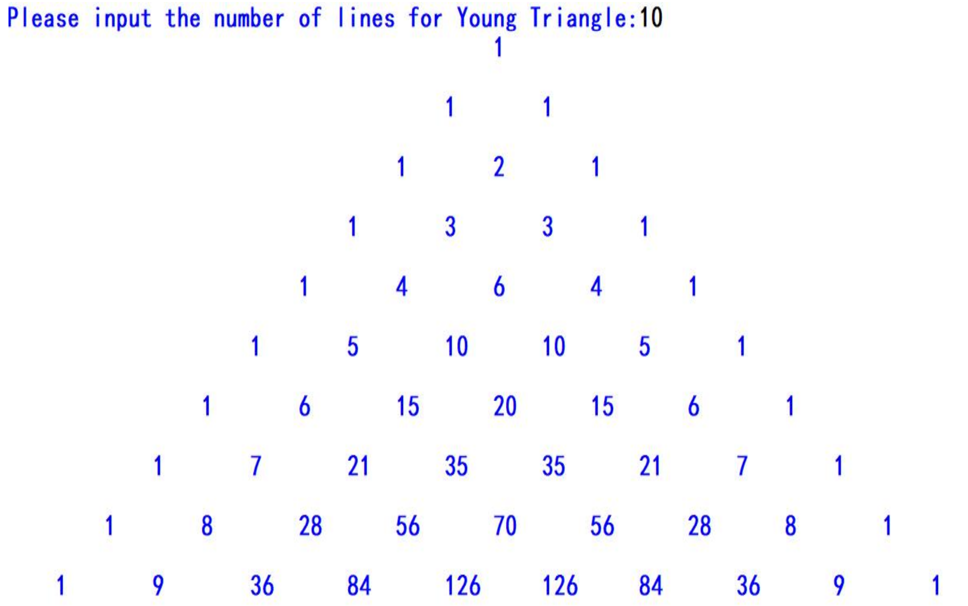


```
# Returns true if the sum of any two sides is
#  greater than the third side.
def isValid(side1, side2, side3):

# Returns the area of the triangle.
def area(side1, side2, side3):
```

Write a test program that reads three sides for a triangle and computes the area if the input is valid. Otherwise, it displays that the input is invalid.

Q4: Yang Hui’ s Triangle

- Yang Hui’ s Triangle(Pascal’ s Triangle) is a triangle with numbers where each of them in one line equals to the sum of the two neighbor numbers in previous line. Equivalently, it is a triangular array of the binomial coefficients. Write a program to print Yang Hui’ s Triangle with given number of lines.



```
Please input the number of lines for Young Triangle:10
                                    1

                                1       1

                            1       2       1

                        1       3       3       1

                    1       4       6       4       1

                1       5       10      10      5       1

            1       6       15      20      15      6       1

        1       7       21      35      35      21      7       1

    1       8       28      56      70      56      28      8       1

1       9       36      84      126     126     84      36      9       1
```

- Hints: ①Define a function factorial(n) to calculate factorial of n: n!=n(n-1)…1. ②Define a function combination(n,k) to calculate binomial coefficients-combination $\binom{n}{k}=\frac{n!}{(n-k)!k!}$.