**[อาร์เรย์ (array) เบื้องต้นใน Python](https://python3.wannaphong.com/2014/12/array-python.html)**

  อาร์เรย์ (array) เป็นข้อมูลชนิดหนึ่ง เป็นโครงสร้างข้อมูล แต่ว่า Python ไม่สนับสนุนอาร์เรย์ชนิดนี้โดยตรง จึงต้องใช้[ชนิดข้อมูลที่เรียกว่ารายการ](https://python3.wannaphong.com/2014/07/basic-data-types.html) ซึ่งเกือบเหมือนกับอาร์เรย์ แต่สามารถเข้าถึงได้เพียงหนึ่งดัชนี

อาร์เรย์ 1 มิติ

ใน Python จะมีรูปแบบดังนี้ [1,2,3,4,5,6]

a = [1,2,3,4,5,6]

print(a[2])

จะได้ผลลัพธ์ คือ 3 จะเห็นได้ว่าใน python จะนับอาร์เรย์ตัวที่ 1  เป็น 0 ไล่ไปเรื่อย ๆ  
หากต้องการจำนวน 3 ถึง 5 ใช้คำสั่ง

print(a[2:5])

ผลลัพธ์จะได้ [3, 4, 5]  
ถ้าต้องการค่าข้อมูลบนอาร์เรย์ ตามตำแหน่งคอลัมน์

print(a[5:])

จะได้ผลลัพธ์ [6]  
ต่อมาเราจะเปลี่ยนค่าข้อมูลในอาร์เรย์กันครับ  
เช่น a[5] จากเดิมเป็น  [6] หากใช้คำสั่ง

>>a[5] = 11

>>print(a[5:])

[11]

จะเห็นได้ว่าข้อมูลใหม่จะทับข้อมูลเก่า ข้อมูลเก่าจะหายไปเป็นไปตามหลักการเก็บข้อมูลของคอมพิวเตอร์  
หากใช้คำสั่ง print(a[:]) คือแสดงอาร์เรย์ทั้งหมดออกมา

>>> print(a[:])

[1, 2, 3, 4, 5, 11]

อาร์เรย์ 2 มิติ

อาร์เรย์ 2 มิติมีการจัดเรียงข้อมูลในแถว  / คอลัมน์ สามารถเปรียบเทียบได้กับเมทริกซ์ในคณิตศาสตร์  
ใน Python อาร์เรย์ 2 มิติ จะมีรูปแบบเป็น [[1, 2], [3, 4]]

b = [[1, 2], [3, 4]]

ถ้าเป็นรูปแบบเมทริกซ์ในคณิตศาสตร์จะเป็น  
b =  [1, 2]  
       [3, 4]  
หากต้องการดึงข้อมูลมาใช้จากอาร์เรย์ 2 มิติ ใช้คำสั่งดังนี้ เช่น

>>> b = [[1, 2], [3, 4]]

>>> print(b[0][0])

1

แถวที่ 1 หลักที่ 1 คือ 1

>>> print(b[0][1])

2

แถวที่ 1 หลักที่ 2 คือ 2

>>> print(b[1][0])

3

แถวที่ 2 หลักที่ 1 คือ 3 

>>> print(b[1][1])

4

แถวที่ 2 หลักที่ 2 คือ 4  
หากต้องการแสดงทั้งแถวและคอลัมน์ใช้คำสั่ง

>>> print(b[0:][0:])

[[1, 2], [3, 4]]

หากต้องการเฉพาะแถวที่ 1 ใช้คำสั่ง

>>> print(b[0][0:])

[1, 2]

อาร์เรย์ 3 มิติ

มีลักษณะการใช้งานเหมือนกับที่ผ่านมา แต่มีอาร์เรย์เพิ่มอีกมิติ ตัวอย่างเช่น  
[[[1, 2, 3], [4, 5, 6], [7, 8, 9]],  
 [[10, 11, 12], [13, 14, 15], [16, 17, 18]],  
 [[19, 20, 21], [22, 23, 24], [25, 26, 27]]]  
การนำไปใช้งาน

>>> d = [[[1, 2, 3], [4, 5, 6], [7, 8, 9]],

 [[10, 11, 12], [13, 14, 15], [16, 17, 18]],

 [[19, 20, 21], [22, 23, 24], [25, 26, 27]]]

>>> d[0][1]

[4, 5, 6]

>>> d[0][1][2]

6