



มหาวิทยาลัยราชภัฏนครปฐม
Nakhon Pathom Rajabhat University

บทที่ 7 ข้อมูลประเภททูเพิล (Tuple) ของภาษา python

รายวิชา 3602801 การเขียนโปรแกรมคอมพิวเตอร์ทางธุรกิจ

ผศ.ดร.เดช ธรรมศิริ



การประกาศ Tuple ในภาษา Python

- ในภาษา Python การประกาศ Tuple นั้นสามารถทำได้หลายวิธี รูปแบบพื้นฐานของมันคือสมาชิกของ Tuple ทุกตัวจะอยู่ภายในวงเล็บ () และคั่นสมาชิกแต่ละตัวด้วยเครื่องหมายคอมมา (,) ดังตัวอย่าง
- `thistuple = ("apple", "banana", "cherry")`
`print(thistuple)`

```
('apple', 'banana', 'cherry')
```



Tuple อนุญาตให้เก็บข้อมูลซ้ำกันได้

- `thistuple =`
 `("apple", "banana", "cherry", "apple", "cherry")`
 `print(thistuple)`

```
('apple', 'banana', 'cherry', 'apple', 'cherry')
```



การหาจำนวนข้อมูลใน Tuple ด้วย len() function:

- `thistuple = ("apple", "banana", "cherry")`
`print(len(thistuple))`

3



การสร้าง Tuple จาก ข้อมูลเพียง 1 ตัว

- `thistuple = ("apple",)`
`print(type(thistuple))`

#อันนี้ไม่ใช่ `tuple` เป็นแค่ตัวแปร `string` ธรรมดา
`thistuple = ("apple")`
`print(type(thistuple))`

```
<class 'tuple'>  
<class 'str'>
```



ประเภทข้อมูลที่สามารถเก็บใน Tuple ได้

- tuple1 = ("apple", "banana", "cherry")
- tuple2 = (1, 5, 7, 9, 3)
- tuple3 = (True, False, False)
- tuple1 = ("abc", 34, True, 40, "male")



การเข้าถึงข้อมูลใน Tuple ผ่าน index

- `thistuple = ("apple", "banana", "cherry")`
`print(thistuple[1])`

banana



การใช้ค่า index เป็น ลบ

- `thistuple = ("apple", "banana", "cherry")`
`print(thistuple[-1])`

cherry

- #หมายเหตุ -1 หมายถึง การเข้าถึงข้อมูลตัวสุดท้าย -2 เป็นการเข้าถึงข้อมูลก่อนสุดท้าย



การกำหนดช่วงการดึงข้อมูลผ่าน index

- `thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")`
`print(thistuple[2:5])`
`('cherry', 'orange', 'kiwi')`
- `thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")`
`print(thistuple[:4])`
`('apple', 'banana', 'cherry', 'orange')`
- `thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")`
`print(thistuple[2:])`
`('cherry', 'orange', 'kiwi', 'melon', 'mango')`
- `thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")`
`print(thistuple[-4:-1])`
`('orange', 'kiwi', 'melon')`



การเช็คค่าใน Tuple

- ```
thistuple = ("apple", "banana", "cherry")
if "apple" in thistuple:
 print("Yes, 'apple' is in the fruits tuple")
```

```
Yes, 'apple' is in the fruits tuple
```



## การแก้ไขค่า ใน Tuple

```
• x = ("apple", "banana", "cherry")
 y = list(x)
 y[1] = "kiwi"
 x = tuple(y)
```

```
print(x)
```

```
('apple', 'kiwi', 'cherry')
```



## การเพิ่มข้อมูลใน Tuple \*\*\* ทำไม่ได้

- `thistuple = ("apple", "banana", "cherry")`  
`thistuple.append("orange") # This will raise an error`  
`print(thistuple)`

```

AttributeError Traceback (most recent call last)
<ipython-input-11-c200857e23e9> in <module>()
 1 thistuple = ("apple","banana","cherry")
----> 2 thistuple.append("orange")# This will raise an error
 3 print(thistuple)

AttributeError: 'tuple' object has no attribute 'append'
```

SEARCH STACK OVERFLOW



ERROR



## หากจะเพิ่มข้อมูล สามารถทำการแปลงค่า Tuple เป็น List

- ```
thistuple = ("apple", "banana", "cherry")  
y = list(thistuple)  
y.append("orange")  
thistuple = tuple(y)  
print(thistuple)
```

```
('apple', 'banana', 'cherry', 'orange')
```



การ Unpacking a Tuple

- `fruits = ("apple", "banana", "cherry")`

```
(green, yellow, red) = fruits
```

```
print(green)  
print(yellow)  
print(red)
```

```
apple  
banana  
cherry
```



การดึงข้อมูลใน Tuple ผ่าน Loop for

- `thistuple = ("apple", "banana", "cherry")`
`for x in thistuple:`
`print(x)`

```
apple  
banana  
cherry
```

- `thistuple = ("apple", "banana", "cherry")`
`for i in range(len(thistuple)):`
`print(thistuple[i])`

```
apple  
banana  
cherry
```



การเชื่อม Tuple

- `tuple1 = ("a", "b", "c")`
`tuple2 = (1, 2, 3)`

```
tuple3 = tuple1 + tuple2  
print(tuple3)
```

```
('a', 'b', 'c', 1, 2, 3)
```




การเพิ่มTuple ด้วยค่าเดิมที่มี

```
• fruits = ("apple", "banana", "cherry")  
mytuple = fruits * 2
```

```
print(mytuple)
```

```
('apple', 'banana', 'cherry', 'apple', 'banana', 'cherry')
```



ขอบคุณครับ