

Unit 11.

ປະເພດຂໍມູນ Dictionary

Learning objectives

- ✓ ສາມາດສ້າງປະເພດຂໍ້ມູນແບບ dictionary ໃຫ້ກັບ objects ແລະ items
- ✓ ເຮັດໃຈຕຶງຄວາມແຕກຕ່າງລະຫວ່າງ dictionary ແລະ list ພ້ອມທັງສາມາດນຳໃຊ້ປະເພດຂໍ້ມູນທີ່ເໝາະສີມສໍາລັບເງື່ອນໄຂທີ່ແຕກຕ່າງກັນ
- ✓ ສາມາດນໍາວິທີການ dictionary key ເພື່ອ ກຳນົດຄ່າຂໍ້ມູນດ້ວຍການນຳໃຊ້ key

Learning overview

- ✓ ຮຽນຮູ້ກ່ຽວກັບການນຳໃຊ້ປະເພດຂໍ້ມູນ dictionary ເພື່ອກຳນົດຄ່າໂດຍການນຳໃຊ້ key
- ✓ ຮຽນຮູ້ວິທີການ dictionary key ແລະ ກຳນົດຄ່າດ້ວຍ key
- ✓ ຮຽນຮູ້ການສິ່ງຄ່າຂອງ dictionary ດ້ວຍການນຳໃຊ້ pop method
- ✓ ຮຽນຮູ້ປະເພດຂໍ້ມູນແບບ json ແລະ convert ຂໍ້ມູນນັ້ນໃຫ້ຢູ່ໃນຮູບແບບ dictionary

Concepts You Will Need to Know From Previous Units

- ✓ ຕ້ອງຮູ້ ແລະ ເຂົ້າໃຈ ກ່ຽວກັບ list ແລະ tuple
- ✓ ຕ້ອງຮູ້ວິທີການນຳໃຊ້ຕຳແໜ່ງຂໍ້ມູນຂອງ items ສໍາລັບ list ແລະ tuple
- ✓ ຮູ່ນຳໃຊ້ slicing ເພື່ອຕັດເອົາບາງສ່ວນຂອງຂໍ້ມູນ

Keywords

Dictionary

**Key-
Value**

**Key approach and
assignment**

json

Mission

1. Real world problem

1.1. Kiosk order បៀវេងមិនអាចទិន្នន័យបាន



- ▶ ເນື່ອງຈາກການແຜ່ລະບາດຂອງພະຍາດທົ່ວໄລກ, ທຸກຢ່າງຕ້ອງມີການການຕິດຕໍ່ສານ.
 - ▶ ການແຜ່ລະບາດຂອງພະຍາດ COVID-19 ສີ່ເປັນໃຫ້ລັດຖະບານເກົ່າຫຼືສະໜັບສະໜູນຕິດຕັ້ງຕັ້ງ kiosks ເຊິ່ງເປັນຕັ້ງທີ່ບໍ່ມີຄົນເຜົາ ໃນຮ້ານກາເຟ ແລະ ຮ້ານອາຫານ.
 - ▶ ຜ່ານການສໍາຫຼວດ ປະຊາຊົນຈໍານວນໜີ່ມີຄວາມເພິ່ງພິຈສູງໃນການນຳໃຊ້ຕັ້ງກ່າວ່າວ.
 - ▶ ຢ່າງໃດກໍຕາມ, ຍັງມີປະຊາຊົນຈໍານວນໜ້າຍລວມທັງຜູ້ສູງອາຍ ແລະ ຜູ້ອ່ນໆທີ່ບໍ່ຄຸນເຄີຍກັບການນຳໃຊ້ເຕັກໂນໂລຢີດິຈິຕອນ ເຫັນວ່າການນຳໃຊ້ການບໍລິການຕັ້ງກ່າວຍັງມີຄວາມຫຍຸ້ງຍາກ.
 - ▶ ສະນັ້ນ ຈຶ່ງຈໍາເປັນຕ້ອງໄດ້ຮູນຮູ້ເຕັກໂນໂລຢີຂັ້ນພື້ນຖານ ເພື່ອເຮັດໃຫ້ການບໍລິການລົກຄ້າໄດ້ດີທີ່ສົດ ທັນນີ້ກໍເພື່ອແນະນຳລາຍການຕ່າງໆໃຫ້ກັບລົກຄ້າໄດ້.

1. Real world problem

1.2. จะЛОМສອງປະເພດຂໍ້ມູນເຊົ້າກັນແນວໃດ



- ▶ ຮັນກາເຟສ່ວນຫຼາຍຈະສະແດງລາຍການເຄືອງດື່ມ ແລະ ລາຄາໄວ້ນໍາກັນ
- ▶ ສະນັ້ນ, ຖ້າມັດລວມ "beverage:price" ເຊົ້າກັນ ຈະຮັດໃຫ້ງ່າຍຕໍ່ ການຂຽນ order ສໍາລັບໂປຣແກຣມ café ທີ່ບໍ່ມີຄົນຄວບຄຸມ.
- ▶ ເຖິງຢ່າງໃດກໍຕາມ, ເນື້ອມີການນຳໃຊ້ list ເພື່ອມັດລວມກັນລະຫວ່າງ ລາຍການເຄືອງດື່ມ ແລະ ລາຄາ ມັນຈະຍາກໃນການຮັດວຽກ ເນື້ອງຈາກ ປະເພດຂໍ້ມູນມັນມີຄວາມແຕກຕ່າງ.
- ▶ ບັນຫາກຳລົດ ການຊອກຫາວິທີລວມເມນູ ແລະ ລາຄາ ໃຫ້ເປັນໜຶ່ງດຽວ ໂດຍມີປະເພດຂໍ້ມູນທີ່ແຕກຕ່າງກັນ.

1. Real world problem

1.3. ການສ້າງໂປຣແກຣມ café menu



- ▶ ໃນນີ້ໄຄງສ້າງຂໍ້ມູນແບບ dictionary ຈະຖືກນຳໃຊ້ ເພື່ອພິມເມນູ ແລະ ລາຄາ ກາເຟ.
- ▶ ໃນຮ້ານກາເຟ ມີເມນູກາເຟຢູ່ 5 ຢ່າງ ແຊ້ງປະກອບມີ: Americano, Iced Americano, Cappuccino, Caffe Latte ແລະ Espresso, ໃນນີ້ມີລາຄາ 3,000 KRW, 3,500 KRW, 4,000 KRW, 4,500 KRW, ແລະ 3,600 KRW ຕາມລຳດັບ.
- ▶ ຈາກນັ້ນ, ໃຫ້ກວດສອບວ່າມີ item ຢູ່ໃນເມນູ ຫຼື ບໍ່ ເພື່ອພິມ item ທີ່ຜູ້ໃຊ້ເລືອກອກກາມາຫາງໜ້າຈຳ.

2. Mission

2.1. café menu ເຮັດວຽກໄດ້ແນວໃດ #1

```
1 menu = {'Americano': 3000, 'Ice Americano': 3500, 'Cappuccino' : 4000, 'Cafe Latte':4500, 'Espresso':3600 }  
2 for key in menu:  
3     print("{} price : {:.}KRW".format(key, menu[key]))
```

Americano	Price : 3,000 KRW
Ice Americano	Price : 3,500 KRW
Cappuccino	Price : 4,000 KRW
Cafe Latte	Price : 4,500 KRW
Espresso	Price : 3,600 KRW

2. Mission

2.1. café menu ເຮັດວຽກໄດ້ແນວໃດ #2

```
1 print('Americano      Price : 3,000 KRW')
2 print('Ice Americano  Price : 3,500 KRW')
3 print('Cappuccino     Price : 4,000 KRW')
4 print('Cafe Latte      Price : 4,500 KRW')
5 print('Espresso        Price : 3,600 KRW')
6 choice = input('Select one menu from the list: ')
7 if choice in menu.keys():
8     print('{} is {:.}KRW. Please make a payment.' .format(choice, menu[choice]))
9 else:
10    print('Sorry. {} is not in the menu.' .format(choice))
```

Americano Price : 3,000 KRW
Ice Americano Price : 3,500 KRW
Cappuccino Price : 4,000 KRW
Cafe Latte Price : 4,500 KRW
Espresso Price : 3,600 KRW
Select one menu from the list: Café Latte
Café Latte is 4,500KRW. Please make a payment.

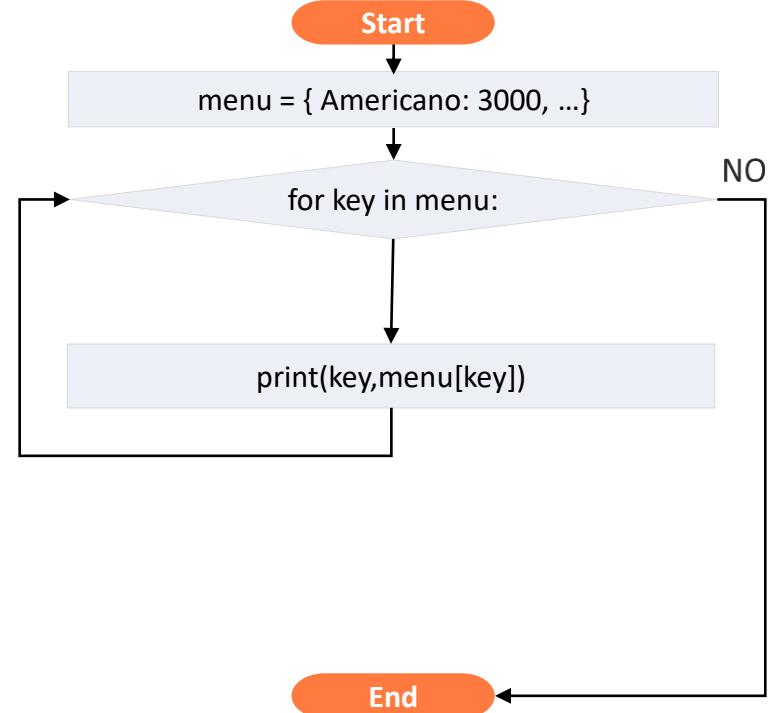
2. Mission

2.2. ချော်ထုတေသန #1

Pseudocode

```
[1] Start  
[2] Enter the menu and price  
[3] for key in the dictionary, as the number of key for printing  
do {  
    [4] print the menu. }  
[5] End
```

Flowchart



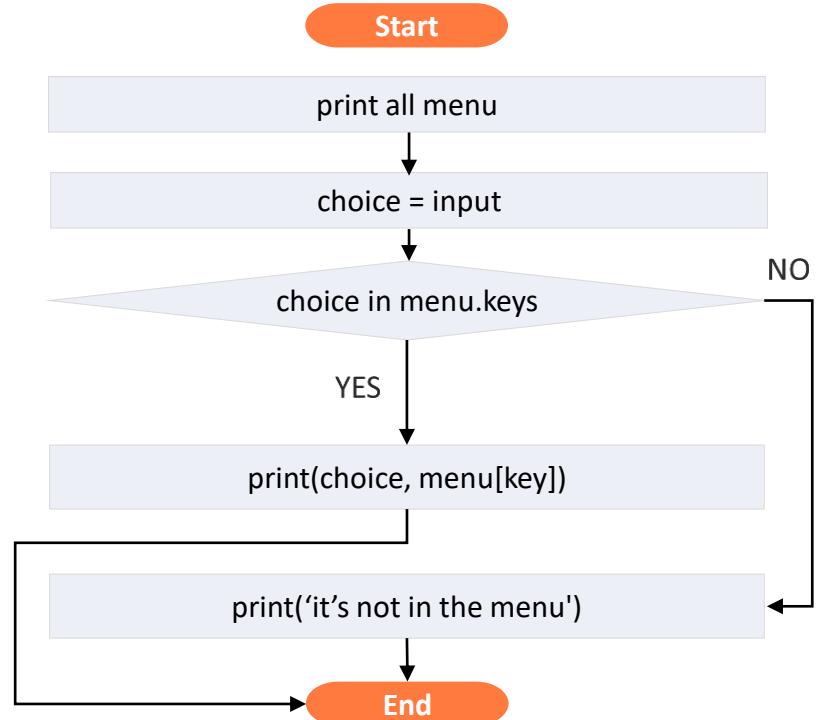
2. Mission

2.2. ချိုင်းယောက်ပြန်အောက် #2

Pseudocode

```
[1] Start  
[2] Print the menu of choice  
[3] Select one menu to eat  
[4] if the drink I chose is in the menu, then{  
[5]   print the menu }  
[6] else{ it's not in the menu. print }  
[7] End
```

Flowchart



2. Mission

2.3. final code 29) café menu #1

```
1 menu = {'Americano': 3000, 'Ice Americano': 3500, 'Cappuccino' : 4000, 'Cafe Latte':4500, 'Espresso':3600 }
2 for key in menu:
3     print("{:16s} price : {:}KRW" .format(key, menu[key]))
```

2. Mission

2.3. final code 29] café menu #2

```
1 print('Americano      Price : 3,000 KRW')
2 print('Ice Americano  Price : 3,500 KRW')
3 print('Cappuccino      Price : 4,000 KRW')
4 print('Cafe Latte       Price : 4,500 KRW')
5 print('Espresso         Price : 3,600 KRW')
6 choice = input('Select one menu from the list: ')
7 if choice in menu.keys():
8     print('{} is {:.}KRW. Please make a payment.' .format(choice, menu[choice]))
9 else:
10    print('Sorry. {} is not in the menu.' .format(choice))
```

| Key concept

1. Dictionary Declaration

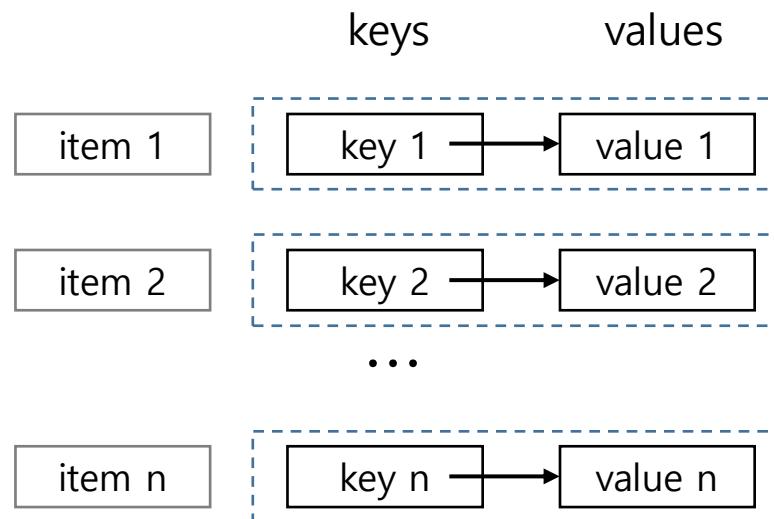
1.1. និយាយខ្លឹមខំរួល Dictionary

- | ឱນ unit នេះ, จะសិនថា វាកំរើវកាបខ្លឹមខំរួល ដូចជាគារបង្ហាញខ្លឹមខំរួល ដែលត្រូវបានរាយការណ៍ នៅក្នុងការបង្ហាញខ្លឹមខំរួល នៃភាសា python.
- | Dictionary ធម៌បញ្ជីរបាយការណ៍ ដែលមានការបង្ហាញខ្លឹមខំរួល ដែលត្រូវបានរាយការណ៍ នៅក្នុងការបង្ហាញខ្លឹមខំរួល នៃភាសា python.
- | មានរាយការបង្ហាញខ្លឹមខំរួល ដែលត្រូវបានរាយការណ៍ នៅក្នុងការបង្ហាញខ្លឹមខំរួល នៃភាសា python.
- | Dictionary ត្រូវបានរាយការណ៍ នៅក្នុងការបង្ហាញខ្លឹមខំរួល ដែលត្រូវបានរាយការណ៍ នៅក្នុងការបង្ហាញខ្លឹមខំរួល នៃភាសា python.
- | សម្រាប់ គ្រប់គ្រង និយាយខ្លឹមខំរួល នៅក្នុងការបង្ហាញខ្លឹមខំរួល នៃភាសា python.

1. Dictionary Declaration

1.1. និយាយខ្លឹមខទេរយ៍ Dictionary

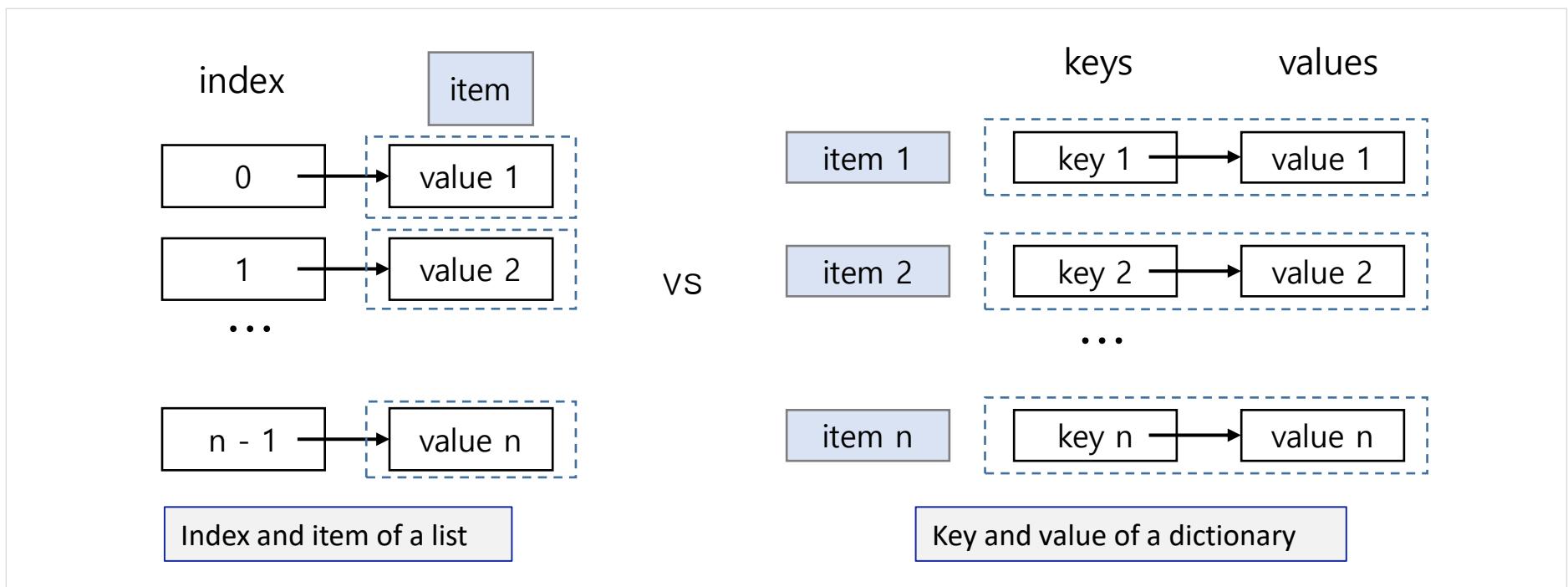
- | ឯកសារនៃ Dictionary គឺជាដែលត្រូវបានដឹងដោយការនិភ័យ [key] : [value].
- | នៃ Dictionary មានការបញ្ជាក់ថា key នឹងចូលរួមនៅក្នុង Dictionary.
- | សារណ៍នៃ Dictionary គឺជាបញ្ជីដែលបានរាយការណ៍ដោយការបញ្ជាក់ថា key នឹងចូលរួមនៅក្នុង Dictionary.



1. Dictionary Declaration

1.2. ការសិរុបលេខវាំង list និង dictionary

- ໃນ list, តាំងអំពីទីមុនសាមាតដីឡើងដែលត្រូវតាមលេខរៀងចំ: [value 1], [value 2], ... [value n] ផ្លូវលក្ខណៈតាមលេខនៃវត្ថុរៀងចំ។
- តាំងអំពីទីមុនខ្លួន list មែនត្រូវតាមលេខ 0 ទៅ n-1.
- ໃនខាងមុខនៃ dictionary, ការចុះតុះខ្លួនទីមុន key-value មែនត្រូវបានបញ្ជាក់។ តួនាទីនឹងនឹងត្រូវបានបញ្ជាក់។



1. Dictionary Declaration

1.3. Dictionary syntax

| ນີ້ແມ່ນ syntax ຂອງ list.

```
List name = [[value 1], ...]
```

| ຕົວຢ່າງລຸ່ມນີ້ ແມ່ນການສ້າງ list ທີ່ຊື່ fruits.

```
fruits = ['banana', 'apple', 'orange', 'kiwi'] # List with string
```

| ນີ້ແມ່ນ syntax ຂອງ dictionary. ການຈັບຄຸ້ງ key-value ແມ່ນໃຊ້ສອງຈໍາ, ດັ່ງສະແດງລຸ່ມນີ້.

```
Dictionary name = {[height 1] : [value 1], ...}
```

| ຕົວຢ່າງລຸ່ມນີ້ ແມ່ນການສ້າງ dictionary ທີ່ຊື່ person.

```
person = {'Name' : 'David Doe', 'Age' : 26, 'Weight' : 82 } # Dictionary with name, age, weight
```

2. ການຈັບຄຸ້ງ Key-Value ຂອງ Dictionary

2.1. ການພິມຄ່າດ້ວຍການນຳໃຊ້ key

| ສ້າງ dictionary ດັ່ງສະແດງໃນຕາຕະລາງລຸ່ມນີ້. Code ຄໍາສັ່ງຂ້າງລຸ່ມເປັນການຄົ້ນຫາຄ່າ 'David Doe' ຈາກ dictionary ທີ່ຊື່ person ດ້ວຍ key 'Name,' ແລະ ເຊັ່ນດຽວກັນໃຫ້ຊອກຫາຄ່າຂຶ້ນນຸ່ມໂດຍນຳໃຊ້ key 'Age' ແລະ 'Weight.'

Item	key	value
Item 1	'Name'	'David Doe'
Item 2	'Age'	26
Item 3	'Weight'	82

```
1 person = {'Name': 'David Doe', 'Age': 26, 'Weight': 82}
```

```
1 person['Name']
```

'David Doe'

```
1 person['Age']
```

26

```
1 person['Weight']
```

82

2. ການຈັບຄຸ້ງ Key-Value ຂອງ Dictionary

2.2. ຕົວຢ່າງ Code ຄໍາສັ່ງການຈັບຄຸ້ງໃນ dictionary

| ສ້າງ dictionary ໃໝ່ຂຶ້ນມາ ໂດຍມີຊື່ວ່າ “students” ທີ່ປະກອບດ້ວຍການຈັບຄຸ້ງ [key] : [value] ສໍາລັບນັກສຶກສາ 3 ຄົນ.

Item	key	value
Item 1	2019001	'John Smith'
Item 2	2019002	'Jane Carter'
Item 3	2019003	'Peter Kelly'

```
1 students = { 2019001:'John Smith', 2019002:'Jane Carter', 2019003:'Peter Kelly'}
```

```
2 students[2019001]
```

```
'John Smith'
```

```
1 students[2019002]
```

```
'Jane Carter'
```

```
1 students[2019003]
```

```
'Peter Kelly'
```

3. Key และ Value ຂອງ Dictionary

3.1. ການແຊກອີງປະກອບໃໝ່ເຂົ້າໃນ dictionary

- | ນໍາໃຊ້ method ລຸ່ມນີ້ ເພື່ອແຊກອີງປະກອບໃໝ່ເຂົ້າໄປ dictionary.

Dictionary name [key] = value

- | ສໍາລັບການລຶບ, ໃຫ້ປ້ອນ key ຂອງອີງປະກອບທີ່ຕ້ອງການລຶບ ໂດຍໃຊ້ຄໍາສັ່ງ del .
- | KeyError ເກີດຂຶ້ນ ເມື່ອ ມີການປ້ອນ key ທີ່ຕ້ອງການລຶບບໍ່ຖືກຕ້ອງ.

3. Key และ Value ຂອງ Dictionary

3.2. ການແຊກ ແລະ ການປັບປຸງຂອງປະກອບໃນ dictionary

| ຕົວຢ່າງລຸ່ມນີ້ແມ່ນການເພີ່ມຂຶ້ນປະກອບເຂົ້າໄປໃນ dictionary: ໂດຍໃຫ້ key: 'Job' ແລະ value : 'Data Scientist'

```
1 person = {'Name' : 'David Doe', 'Age' : 26, 'Weight' : 82 }
```

```
1 person['Job'] = 'Data Scientist' # New key:insert the item of the value
```

```
1 person
```

```
{'Name': 'David Doe', 'Age': 26, 'Weight': 82, 'Job': 'Data Scientist'}
```

| ການປັບປຸງຂອງປະກອບໃນ dictionary: ປັບປຸງຄ່າ Age ຈາກ 26 ເປັນ 27 ດ້ວຍ key : 'Age' ແລະ value : 27. ດັ່ງສະແດງລຸ່ມນີ້

```
1 person = {'Name' : 'David Doe', 'Age' : 26, 'Weight' : 82 }
```

```
1 person['Age'] = 27
```

```
1 person
```

```
{'Name': 'David Doe', 'Age': 27, 'Weight': 82}
```

3. Key และ Value ຂອງ Dictionary

3.3. ການລຶບອົງປະກອບໃນ dictionary

| ການລຶບ ອົງປະກອບໃນ dictionary : ລຶບ 'Age': 26 ໂດຍໃຊ້ຄຳສັ່ງ del

```
1 person = {'Name' : 'David Doe', 'Age' : 26, 'Weight' : 82 }
```

```
1 del person['Age']
```

```
1 person
```

```
{'Name': 'David Doe', 'Weight': 82}
```

3. Key และ Value ຂອງ Dictionary

3.3. ການລຶບອີງປະກອບໃນ dictionary

!

ຄໍາເຕືອນ: ເວລາລຶບ ອີງປະກອບໃນ dictionary, KeyError ຈະເກີດຂຶ້ນ ຫຼາຍ້ອນ key ບໍ່ຖືກຕ້ອງ

```
1 person = {'Name' : 'David Doe', 'Age' : 26, 'Weight' : 82 }  
1 del person['Hometown']
```

```
KeyError Traceback (most recent call last)  
<ipython-input-24-2edca7f1fd7c> in <module>  
----> 1 del person['Hometown']
```

KeyError: 'Hometown'

- ຂ້າງເທິງນີ້, KeyError ເກີດຂຶ້ນ ເມື່ອມີການຢ້ອນ key ບໍ່ຖືກຕ້ອງ ແຊ້ນ: ‘Hometown.’

3. Key และ Value ແລະ Dictionary

3.4. ພັງຊັນ ແລະ ຕົວດຳເນີນການສໍາລັບ dictionary

- | ພັງຊັນ ແລະ ຕົວດຳເນີນການໃນ dictionary ແມ່ນຄ້າຍຄືກັບ list.
 - | ພັງຊັນ len
 - ▶ ໃຊ້ສໍາລັບນັບຈຳນວນອີງປະກອບ ຫຼື ຄວາມຍາວຂອງ dictionary
 - | ຕົວດຳເນີນການ in
 - ▶ ສິ່ງຄ່າ True ຖ້າ key ມີໃນ dictionary
 - | ຕົວດຳເນີນການ not in
 - ▶ ສິ່ງຄ່າ False ບ້າ key ບໍ່ມີໃນ dictionary
 - | ຕົວດຳເນີນການ ==, !=
 - ▶ ໃຊ້ກວດສອບອີງປະກອບໃນ 2 dictionary ວ່າມີອີງປະກອບຕີກັນຫຼືບໍ່

3. Key และ Value ຂອງ Dictionary

3.4. ផ្សេង និង ពិវត្តន៍យុទ្ធសាស្ត្រលីប dictionary

- | ផ្តល់ព័ត៌មានអំពីការបង្កើតកូដនៃការសរសៃរបស់ខ្លួន នៅក្នុង dictionary ដោយការប្រើប្រាស់ពិតិយការ “in” និង ពិតិយការ “not in”
- | key ‘Name’ គឺជាកូដដែលបានបង្កើតឡើងនៅក្នុង dictionary ដើម្បីត្រូវបានបញ្ជូនទៅការបង្កើតនៃការបង្កើតនៃកូដ។ នៅពេលបង្កើតនៅក្នុង dictionary កូដ ‘Name’ នឹងត្រូវបានបង្កើតឡើងដោយត្រូវបានបញ្ជូនទៅការបង្កើតនៃកូដ។ នៅពេលបង្កើតនៅក្នុង dictionary កូដ ‘Job’ នឹងត្រូវបានបង្កើតឡើងដោយត្រូវបានបញ្ជូនទៅការបង្កើតនៃកូដ។ នៅពេលបង្កើតនៅក្នុង dictionary កូដ ‘Age’ នឹងត្រូវបានបង្កើតឡើងដោយត្រូវបានបញ្ជូនទៅការបង្កើតនៃកូដ។ នៅពេលបង្កើតនៅក្នុង dictionary កូដ ‘Weight’ នឹងត្រូវបានបង្កើតឡើងដោយត្រូវបានបញ្ជូនទៅការបង្កើតនៃកូដ។

```
1 person = {'Name' : 'David Doe', 'Age' : 26, 'Weight' : 82 }
```

```
1 len(person) # Returns the number of items in dictionary
```

3

```
1 'Name' in person # 'Name' is found in the key
```

True

```
1 'Job' in person # 'Job' is not found in the key
```

False

```
1 'David Doe' in person # 'David Doe' is found in the value but not in the key (caution)
```

False

```
1 'David Doe' not in person # Returns True because 'David Doe' is not found in the key
```

True

3. Key และ Value ຂອງ Dictionary

3.4. ផ្សេង និង ពិវត្តន៍យុទ្ធសាស្ត្រ dictionary

- | ចុងក្រោមវិវាទៗ value, ប៉ុណ្ណោះ key, សារធានាបែងចែកពីរដែលមានការ “in”
- | “David Doe” រួមជារៀង់រាល់ នៅក្នុង value ប៉ុណ្ណោះ ប៉ុណ្ណោះ key ត្រូវបានស្វែងរកដោយការ “in” ចុងក្រោមវិវាទៗ False ទេ។

```
1 'David Doe' in person # 'David Doe' is found in the value but not in the key (caution)
```

False

```
1 'David Doe' not in person # Returns True because 'David Doe' is not found in the key
```

True

3. Key และ Value ຂອງ Dictionary

3.4. ផ្ទាត់ខ្លួន និង ពិនិត្យការសំល។ប dictionary

- | ពិនិត្យការសំល។ប == ត្រូវត្រូវការសំល។បដើម្បីរាយការណ៍ និងការសំល។បទិន្នន័យ។
- | ពិនិត្យការសំល។ប != ត្រូវត្រូវការសំល។បទិន្នន័យ។

```
1 d1 = {'Name' : 'David Doe', 'Age' : 26 }
```

```
1 d2 = {'Age' : 26, 'Name' : 'David Doe' }
```

```
1 d1 == d2
```

True

```
1 d1 != d2
```

False

3. Key และ Value ຂອງ Dictionary

3.5. ពិវត្តន៍យោងការងារប៉ះសាមាតធ្វើនៅ dictionary



ការសិរុប្បីនៃ dictionary – TypeError

```
1 d1 > d2
```

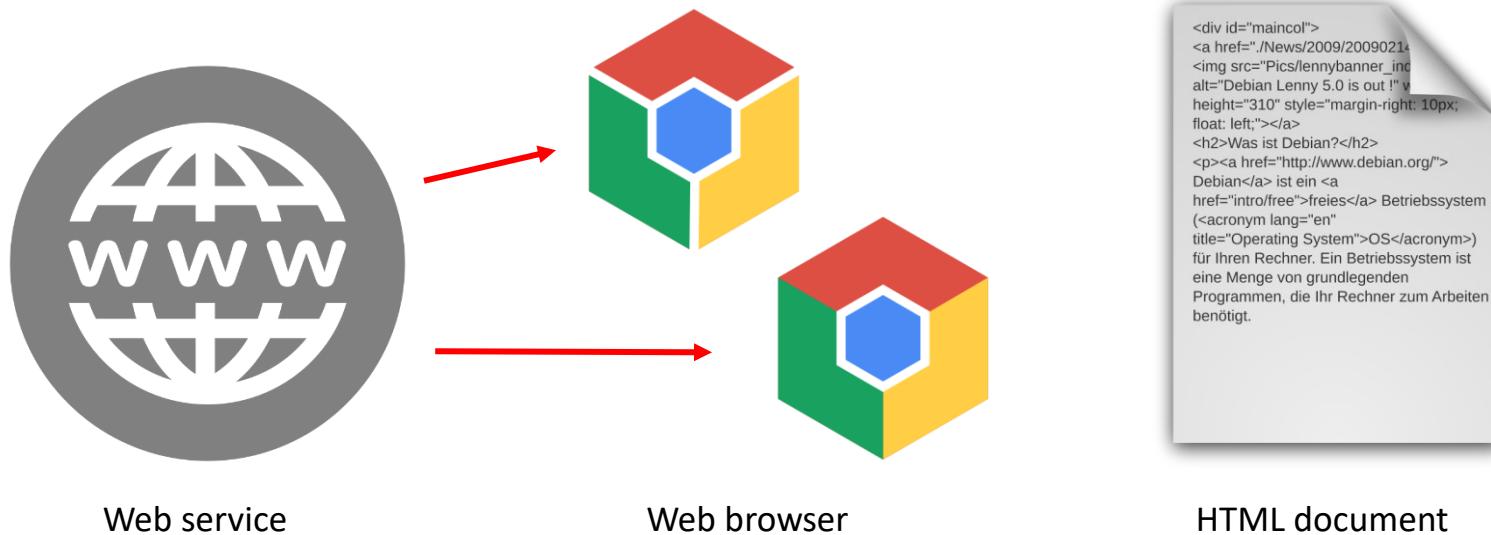
```
TypeError                                 Traceback (most recent call last)
<ipython-input-26-93fca8e036ac> in <module>
      1 d1 > d2
TypeError: '>' not supported between instances of 'dict' and 'dict'
```

- នៃ Dictionary ប៉ះសាមាតណា ឲ្យពិវត្តន៍យោងការងារ `>`, `>=`, `<`, `<=`. ត្រូវឲ្យដាក់ពីការងារនេះ ដូចជាអាសយដ្ឋាន.

4. json Data Dictionary / List Conversion

4.1. ມາດຕະຖານຂອງເອກະສານໃນໜ້າ Web service

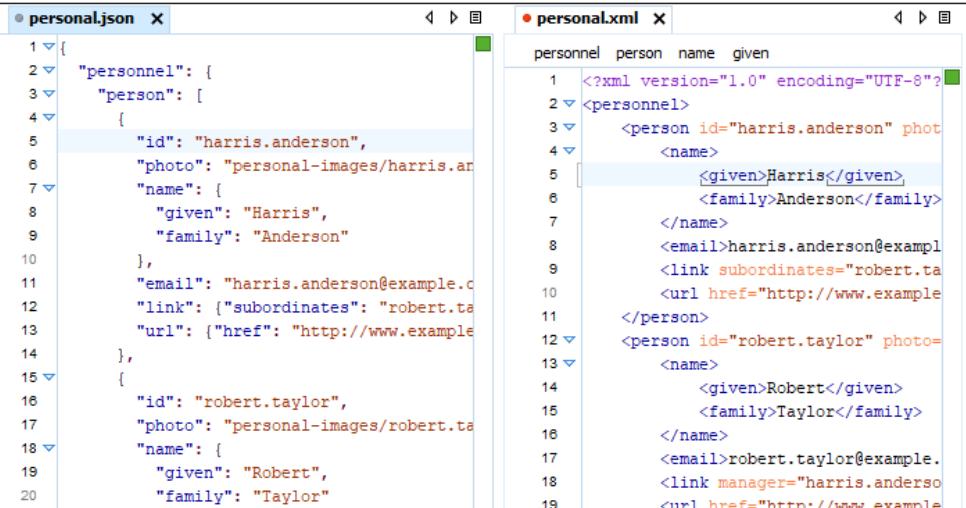
- | ໂດຍທີ່ໄປການໃຊ້ມາດຕະຖານເອກະສານໃນໜ້າ web services ແມ່ນ HTML (HyperText Markup Language) document.
- | HTML document ແມ່ນກຸ່ມຍ່ອຍຂອງ XML document.
- | XML (eXtensible Markup Language) ແມ່ນປະເພດໜຶ່ງຂອງ file ທີ່ສະແດງໂຄງສ້າງຂອງຂໍ້ມູນຜ່ານ text document.



4. json Data Dictionary / List Conversion

4.2. json type

- | พิจารณา วิธีการสื่อสารด้วยภาษาสำหรับ text document.
- | วิธีการใดที่จะสื่อสาร เช่น “Name”：“David Doe”
- | ตัวแบบ text document นี้ เป็น JSON.
 - ▶ JSON เป็น text format ที่สามารถ อ่าน และ เขียน ข้อมูล ได้มากกว่า XML.
 - ▶ ใน Unit นี้, ขอเรียกชื่อ ภาษาสำหรับ JSON แบบง่ายๆ.



```
personal.json
1 {
2   "personnel": [
3     "person": [
4       {
5         "id": "harris.anderson",
6         "photo": "personal-images/harris.an",
7         "name": {
8           "given": "Harris",
9           "family": "Anderson"
10          },
11         "email": "harris.anderson@example.c",
12         "link": {"subordinates": "robert.ta",
13         "url": {"href": "http://www.example
14        },
15        {
16          "id": "robert.taylor",
17          "photo": "personal-images/robert.ta
18          "name": {
19            "given": "Robert",
20            "family": "Taylor"
21          }
22        }
23      ]
24    }
25  ]
26}
```

```
personal.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <personnel>
3   <person id="harris.anderson" phot
4     <name>
5       <given>Harris</given>
6       <family>Anderson</family>
7     </name>
8     <email>harris.anderson@example.
9     <link subordinates="robert.ta
10    <url href="http://www.example
11  </person>
12  <person id="robert.taylor" photo=
13    <name>
14      <given>Robert</given>
15      <family>Taylor</family>
16    </name>
17    <email>robert.taylor@example.
18    <link manager="harris.anderso
19  <url href="http://www.example
20
```

JSON

XML(eXtensible Markup Language)



4. json Data Dictionary / List Conversion

4.3. ການນິຍາມ json

- | JSON (JavaScript Object Notation) ແມ່ນພັດທະນາຂຶ້ນເພື່ອສື່ງຂໍ້ມູນ object ທີ່ປະກອບດ້ວຍ “key-value pair.”
- | ຂໍ້ມູນນີ້ແມ່ນ format ທີ່ເປັນມາດຕະຖານ ທີ່ສາມາດອ່ານໄດ້. ມັນເປັນຮູບແບບຂໍ້ມູນສໍາລັບການສື່ສານລະຫວ່າງ browser ແລະ server.
- | ໂດຍສະເພາະ ມັນຖືກໃຊ້ຢ່າງກວ້າງຂວາງ ເພື່ອແລກປ່ຽນຂໍ້ມູນເທິງ Internet.
- | ບໍ່ມີຂໍ້ຈໍາກັດໃດໆໃນປະເພດຂໍ້ມູນ, ມັນມີຄວາມເໝາະສົມທີ່ຈະນຳມາໃຊ້ເປັນຕົວປ່ຽນໃນໂປຣແກຣມຄອມພິວເຕີ.
- | ໃນເບື້ອງຕົ້ນ, ມັນມາຈາກພາສາ JavaScript , ສະນັ້ນ ມັນຈຶ່ງມີ syntax ໄປຕາມ JavaScript, ແຕ່ມັນເປັນພາສາທີ່ມີຮູບແບບຂໍ້ມູນເປັນເອກະລາດ
- | ສະນັ້ນ, ນັບແຕ່ມັນມີຄວາມເປັນເອກະລາດທາງດ້ານ programming language ຫຼື platform, ໂດດຄໍາສັ່ງສໍາລັບ syntax analysis ແລະ JSON data ຈຶ່ງ ຖືກນຳໃຊ້ໄດ້ຢ່າງໆຍາດາຍໃນພາສາໂປຣແກຣມຕ່າງໆ ເຊັ່ນ: C, C++, C#, Java, JavaScript, Perl, Python, etc.

4. json Data Dictionary / List Conversion

4.4. ຕົວຢ່າງ json data

| ລຸ່ມນີ້ສະແດງໃຫ້ເຫັນຕົວຢ່າງຂອງ JSON file. ຈາກ JSON ປະກອບມີ ຂໍ້ມູນສ່ວນບຸກຄົນ ເຊັ່ນ: name, age, etc.

```
{  
    "Name": "David Doe",  
    "Age": 25,  
    "Hobby": ["basketball"]  
    "Family": {"father": "John Doe", "mother": "Marry Doe"},  
    "Married": true  
}
```

→ ໃນນີ້ ສ່ວນທຳອິດແມ່ນ key, ສ່ວນທີ່ສອງແມ່ນ value
→ ສ່ວນທີ່ເປັນຕົວເລກແມ່ນ value
→ value ສາມາດເປັນ List ໄດ້
→ value ສາມາດເປັນ dictionary
→ value ສາມາດເປັນ Boolean

4. json Data Dictionary / List Conversion

4.5. ពិរួយៗ Code ការសែងរកនៃការប្រើប្រាស់ការបង្កើត json ជា dictionary

- | ឲ្យសែនខោយ json មិនមែនចាប់ផ្តើនឱ្យមែនជាផាយ json ទីនេះ strings បែងdictionary និង បញ្ជូនជាពាយ strings.
- | ដើម្បីអាចបង្កើត json ឱ្យមែនជាផាយដែលមានការប្រើប្រាស់ការបង្កើត loads. និងការបង្កើតការបង្កើត dictionary នៃការបង្កើត json ដោយប្រើប្រាស់ការបង្កើត loads ដែលបង្កើតឡើងជាបញ្ហាប្រើប្រាស់ការបង្កើត dictionary ដោយការបង្កើត loads នៃការបង្កើត json module.

```
1 import json
2
3 data = '{"Name": "David Doe", "Age": 25, "Hobby": ["basketball"], \
4 "Family": {"father": "John Doe", "mother": "Marry Doe"}, "Married": true}'
5
6 json_data = json.loads(data)
7
8 print(type(json_data))
9 print(json_data['Name'])
10 print(json_data['Family'])
11 print(json_data['Married'])
```

```
<class 'dict'>
David Doe
{'father': 'John Doe', 'mother': 'Marry Doe'}
True
```

4. json Data Dictionary / List Conversion

4.5. ពិរួយា Code ការសែងរកចំណាំក្នុង json ជំនួយ dictionary

- | json module ខាង Python ត្រូវបានដោឡូលើក្នុងតាមរយៈដើរតាមខ្លួន ដោយ json ត្រូវបានបង្កើតជាបញ្ជី strings ជំនួយ dictionary និង បញ្ជី strings ជំនួយ list . ម៉ោងនឹងធ្វើការក្នុង json module នៃកើតឡើងនៅក្នុងតាមរយៈដើរតាមខ្លួន .
- | ឧបតាថ្មីទាំងនេះមានការបង្កើតជាបញ្ជី strings ជំនួយ dictionary និង បញ្ជី strings ជំនួយ list .

```

1 import json
2
3 data = '{"title": "The Old man and The Sea", "ISBN": "12345", "Author" :"Ernest Hemingway"}'
4 json_data = json.loads(data)
5
6 print(type(json_data))
7 print(json_data['title'])
8 print(json_data['ISBN'])
9 print(json_data['Author'])

```

```

<class 'dict'>
The Old man and The Sea
12345
Ernest Hemingway

```

 Line 4

- បង្កើតជាបញ្ជី string ជំនួយ json ត្រូវបានដោឡូលើក្នុង loads .
- សម្រាប់ json_data object ត្រូវបានបង្កើតជាបញ្ជី dictionary.

4. json Data Dictionary / List Conversion

4.6. ពិវិះការប្រើប្រាស់មុនបច្ចេកទៀត dictionary បែង json

| បានឱ្យដាក់ជាបញ្ជី "dump" នៃ json module ដើម្បីប្រើប្រាស់មុនបច្ចេកទៀត dictionary បែងជាផាយ json.

```
1 import json
2
3 data = '{"title": "The Old man and The Sea", "ISBN": "12345", "Author": "Er
4 json_data = json.loads(data)
5
6 # Code to create json_data as book.json file
7 with open('book.json', 'w') as f:
8     json.dump(json_data, f, indent='\t')
```

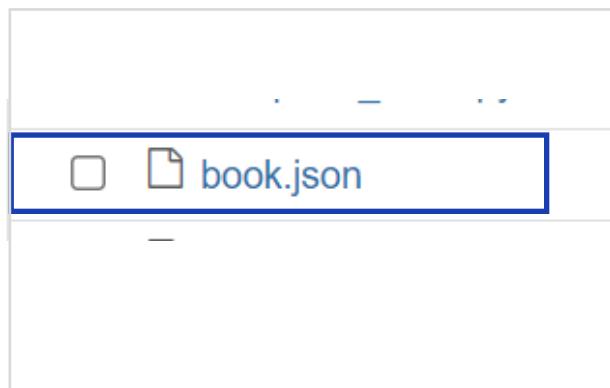
Line 7,8

- បានឱ្យដាក់ជាបញ្ជី with open() ដើម្បីសំរាប់ book.json ដែលមានការបែង mode ('w').
- បានឱ្យដាក់ជាបញ្ជី json.dump ដើម្បីបង្ហាញ json_data ដោយបង្ហាញជាផីរីកិច្ច "f" នៃ json type.
- ការបង្ហាញមានការណា-indent = '\t'

4. json Data Dictionary / List Conversion

4.7. បំពុកខ្លឹមុន json ដោយ file

- | ឯកសារ book.json តិចសារឱ្យខ្លួនជាភាសាភិភាក្សានៃកូដរបស់ក្នុង Jupyter Notebook.
- | បើកឯកសារ book.json ដោយប្រើប្រាស់កូដខាងលើមនី.
- | សំណើនាំ tab ខ្លឹមុនពីរូបគោលនៃខ្លួនដែលត្រូវបានអនុញ្ញាតឡើង។



ឯកសារ book.json ដោយប្រើប្រាស់កូដ json.dump

A screenshot of a Jupyter Notebook cell. The title bar says 'jupyter book.json 17분 전'. The cell contains the following JSON data:

```
{  
  "title": "The Old man and The Sea",  
  "ISBN": "12345",  
  "Author": "Ernest Hemingway"  
}
```

ឯកសារ book.json ដោយប្រើប្រាស់កូដ json.dump ត្រូវបានអនុញ្ញាតឡើង។

4. json Data Dictionary / List Conversion

4.8. with ~ as syntax

- | with ~ as syntax จะอะທິບາຍລາຍລະອຽດໃນ Unit 13. ແຕ່ໃນ Unit ນີ້ຈະอะທິບາຍໂດຍຫຍໍ້.
- | ເມື່ອພາຍຂໍ້ມູນມີການປະມວນຜົນໃນ Python, ເຊັ່ນວ່າ ການເປີດພາຍອາດຈະບໍ່ປະສົບຜົນສໍາເລັດ.
- | ຫຼັງຈາກການເປີດພາຍຂຶ້ນມາຮັດວຽກສໍາເລັດແລ້ວ, ຕ້ອງໄດ້ປິດພາຍດັ່ງກ່າວທຸກຄັ້ງ.
- | ການປະມວນຜົນຂອງພາຍສາມາດຮັດໄດ້ດ້ວຍ “with” syntax. ເນື່ອຈາກ “with” syntax ແມ່ນຈະຮັດໃຫ້ການຮັດວຽກຂອງ method ຈຳປັດຂຶ້ນ, ໃນເວລາດຽວກັນມັນກໍຈະລ້າງ syntax.

ອ່ານພາຍ book.json ດ້ວຍຝັງຊັນ open ດ້ວຍການຂຽນໃນຮູບແບບ (w).

ຫຼັງຈາກອ່ານພາຍສໍາເລັດແລ້ວ,
ອ້າງອີງພາຍດັ່ງກ່າວດ້ວຍ file type “f.”

```
1 # Code to create json_data as book.json file
2 with open('book.json', 'w') as f:
3     json.dump(json_data, f, indent='\t')
```

ບັນທຶກ json_data ລົງໃນພາຍ “f.”
ເພື່ອໃຫ້ຈ່າຍໃນການອ່ານ, ຈຶ່ງຕ້ອງນຳໃຊ້ character (\t) ເພື່ອຫຍ້ໜ້າ.

Paper coding

- ຕ້ອງເຂົ້າໃຈແນວຄິດພື້ນຖານຂອງຫຼັກສູດນີ້ໃຫ້ຄົບຖ້ວນກ່ອນຈະໄປສູ່ຂັ້ນຕອນຕໍ່ໄປ.
- ການທີ່ບໍ່ເຂົ້າໃຈແນວຄິດພື້ນຖານ ຈະເພີ່ມພາລະໃນການຮຽນຮູ້ຂອງຫຼັກສູດນີ້ ແລະ ຈະເຮັດໃຫ້ບໍ່ປະສົບຜົນສໍາເລັດ.
- ມັນອາດຈະເປັນເລື່ອງທີ່ຍາກຕອນນີ້, ແຕ່ຖ້າຢາກປະສົບຜົນສໍາເລັດໄດ້ນັ້ນ ພວກເຮົາຂໍແນະນຳໃຫ້ເຂົ້າໃຈແນວຄິດພື້ນຖານ ນີ້ຢ່າງລຶກເຊິ່ງ ແລະ ກ້າວໄປສູ່ຂັ້ນຕອນຕໍ່ໄປ.

Q1. ចូលរួមទៅការក្នុងការសរសៃរបស់ខ្លួន។ តើតុលាការនេះត្រូវបានដោះស្រាយដោយប្រើប្រាស់បណ្តុះបណ្តាលការណ៍ដែលបានផ្តល់ជាបន្ទីរ។ តើតុលាការនេះត្រូវបានដោយប្រើប្រាស់បណ្តុះបណ្តាលការណ៍ដែលបានផ្តល់ជាបន្ទីរ។

Program Execution Variable Declaration	key : Korea value : Seoul key : China value : Beijing key : USA value : Washington DC
Time	5 Minutes

Output example

Seoul

 Write the entire code and the expected output results in the note.

Q2.

จิํงส້າງ dictionary ທີ່ຊື່ fruits_dic ທີ່ມີອີງປະກອບ ຄື key-value. ຈາກນັ້ນໃຫ້ພິມເປັນໄດ້ຮັບອອກທາງໜ້າຈຳ ດັ່ງສະແດງລຸ່ມນີ້.

Conditions for Execution

The price of an apple is 5000 KRW.

The price of a banana is 4000 KRW.

The price of a grape is 5300 KRW.

The price of a melon is 6500 KRW.

Time

5 Minutes

fruits_dic dictionary

key: apple

value: 5000

key: banana

value: 4000

key: grape

value: 5300

key: melon

value: 6500



Write the entire code and the expected output results in the note.

Unit 11. Dictionary Data Types

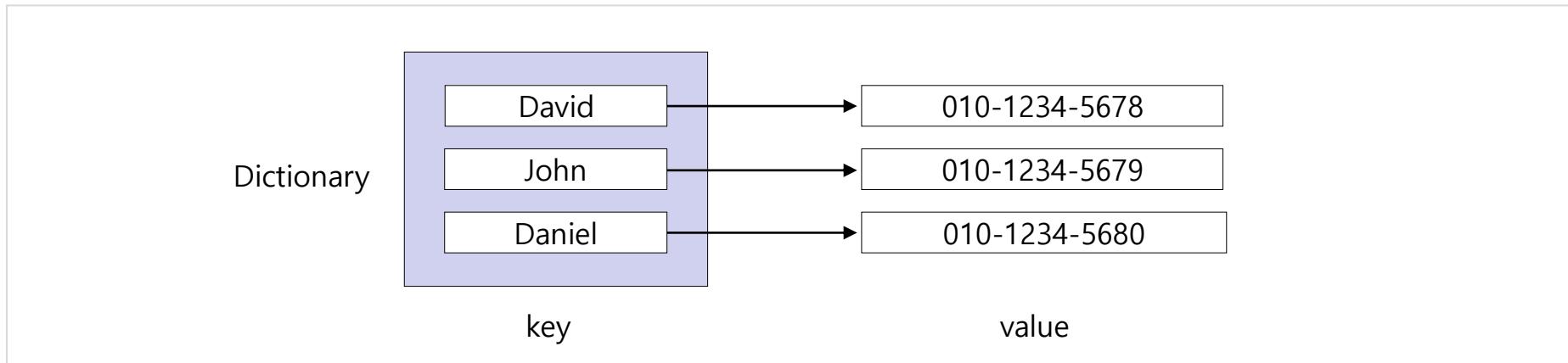
| Let's code

1. Dictionary

1.1. ໂຄງສໍາງຂອງ dictionary

ບັນທຶກຂໍ້ມູນເປັນ dictionary ດ້ວຍ key ແລະ value.

- ຄ້າຢືນກັບ list, dictionary ມີໂຄງສໍາງຂໍ້ມູນ ດັ່ງສະແດງໃນຮູບລຸ່ມນີ້.
- ແຕ່ມັນກຳນົດວ່າ ການເກັບຂໍ້ມູນໃນ dictionary ມີ key ອ້າງອີງໄປຫາ value.



1. Dictionary

1.2. ការស្វែងខ្លឹមប្រប Dictionary

| មិនាយើវិທី ថា តើមានការណែនាំដើម្បីធ្វើឡើងមាយ { } ដើម្បីស្វែង dictionary ទៅវាទៀត់ ទាន់ ឬការណែនាំ ឱ្យបោនខ្លឹមប្របក្នុង key និង value ឯងឲ្យបាន ត្រូវបានរាយការណ៍។

```
1 phone_book = {}      # Create an empty dictionary
```

| នៃលទ្ធផល ឱ្យបានបង្ហាញលើកឈ្មោះ David និងលេខទូរសព្ទ 010-1234-5678 ។

```
1 phone_book["David"] = "010-1234-5678"
```

1. Dictionary

1.3. ການປະຕິບັດການຂອງ dictionary

| ພິມຂໍ້ມູນອອກທາງໜ້າຈຳ.

```
1 print(phone_book)  
{'David': '010-1234-5678'}
```

| Dictionary ສາມາດສ້າງ ແລະ ໄສ່ຄ່າເລີ່ມຕົ້ນໄດ້.

```
1 phone_book = {"David": "010-1234-5678"}
```

| ເພີ່ມເບີໂທລະສັບຄືນອື່ນງົງ ແລະ ພິມຜົນໄດ້ຮັບອອກທາງໜ້າຈຳ ດັ່ງສະແດງລຸ່ມນີ້.

```
1 phone_book["John"] = "010-1234-5679"  
2 phone_book["Daniel"] = "010-1234-5680"  
3 print(phone_book)
```

```
{'David': '010-1234-5678', 'John': '010-1234-5679', 'Daniel': '010-1234-5680'}
```

| ເວລາພິມຜົນໄດ້ຮັບອອກທາງໜ້າຈຳ, ບັນດາອີງປະກອບຈະຖືກແຍກອອກຈາກກັນດ້ວຍເຄື່ອງໝາຍຈຸດ.

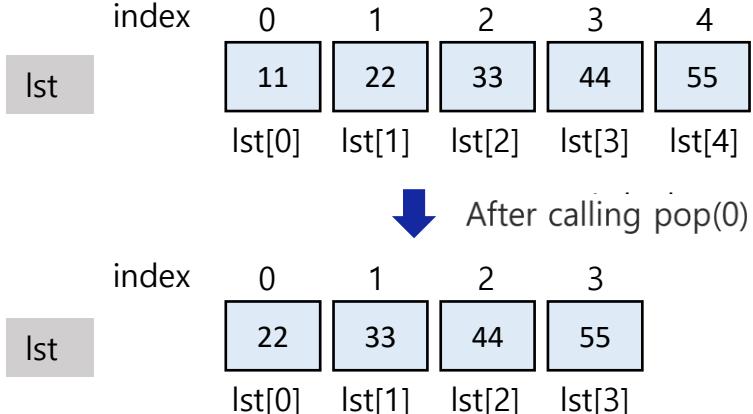
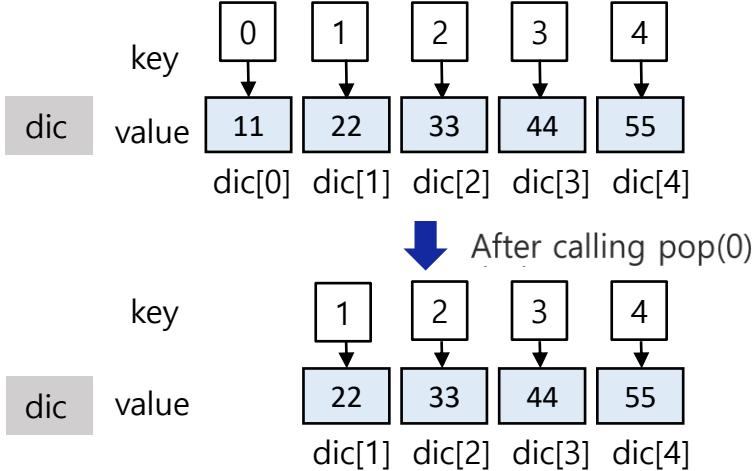
2. ການສືມທຽບລະຫວ່າງ list ແລະ dictionary

2.1. ແຜນສະແດງສືມທຽບລະຫວ່າງ List ແລະ dictionary

List	Dictionary																																				
ຕຳແໜ່ງຂຶ້ນທີ່ມີນຳອິດຈະແມ່ນ 0, ເຊິ່ງມັນກຳນົດຂຶ້ນແບບອັດຕະໂນມັດ.	ກຳນົດ key ແລະ value ເພື່ອຈັບຄູ່.																																				
ex) lst = [11, 22, 33, 44, 55]	ex) dic = {0:11, 1:22, 2:33, 3:44, 4:55}																																				
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: right; padding-right: 10px;">index</td> <td style="text-align: center; padding: 5px;">0</td> <td style="text-align: center; padding: 5px;">1</td> <td style="text-align: center; padding: 5px;">2</td> <td style="text-align: center; padding: 5px;">3</td> <td style="text-align: center; padding: 5px;">4</td> </tr> <tr> <td style="text-align: right; padding-right: 10px;">lst</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">11</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">22</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">33</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">44</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">55</td> </tr> <tr> <td></td> <td style="text-align: center; padding: 5px;">lst[0]</td> <td style="text-align: center; padding: 5px;">lst[1]</td> <td style="text-align: center; padding: 5px;">lst[2]</td> <td style="text-align: center; padding: 5px;">lst[3]</td> <td style="text-align: center; padding: 5px;">lst[4]</td> </tr> </table>	index	0	1	2	3	4	lst	11	22	33	44	55		lst[0]	lst[1]	lst[2]	lst[3]	lst[4]	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: right; padding-right: 10px;">key</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">0</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">1</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">2</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">3</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">4</td> </tr> <tr> <td style="text-align: right; padding-right: 10px;">dic</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">11</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">22</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">33</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">44</td> <td style="text-align: center; border: 1px solid black; padding: 5px;">55</td> </tr> <tr> <td></td> <td style="text-align: center; padding: 5px;">dic[0]</td> <td style="text-align: center; padding: 5px;">dic[1]</td> <td style="text-align: center; padding: 5px;">dic[2]</td> <td style="text-align: center; padding: 5px;">dic[3]</td> <td style="text-align: center; padding: 5px;">dic[4]</td> </tr> </table>	key	0	1	2	3	4	dic	11	22	33	44	55		dic[0]	dic[1]	dic[2]	dic[3]	dic[4]
index	0	1	2	3	4																																
lst	11	22	33	44	55																																
	lst[0]	lst[1]	lst[2]	lst[3]	lst[4]																																
key	0	1	2	3	4																																
dic	11	22	33	44	55																																
	dic[0]	dic[1]	dic[2]	dic[3]	dic[4]																																

2. ການສົມທຽບລະຫວ່າງ list ແລະ dictionary

2.1. ແຜນສະແດງສົມທຽບລະຫວ່າງ List ແລະ dictionary

ການລຶບຂໍ້ມູນໃນ list ໂດຍນໍາໃຊ້ “pop”	ການລຶບຂໍ້ມູນໃນ dictionary ໂດຍນໍາໃຊ້ “pop”																														
<p>ຫຼັງຈາກລຶບ <code>lst.pop(0)</code> <code>lst = [22, 33, 44, 55]</code></p>  <p style="text-align: center;">After calling <code>pop(0)</code></p>	<p>ຫຼັງຈາກລຶບ <code>dic.pop(0)</code> <code>dic = {0:11, 1:22, 2:33, 3:44, 4:55}</code></p>  <p style="text-align: center;">After calling <code>pop(0)</code></p>																														
<p>index</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td> </tr> <tr> <td>11</td><td>22</td><td>33</td><td>44</td><td>55</td> </tr> <tr> <td><code>lst[0]</code></td><td><code>lst[1]</code></td><td><code>lst[2]</code></td><td><code>lst[3]</code></td><td><code>lst[4]</code></td> </tr> </table> <p>lst</p>	0	1	2	3	4	11	22	33	44	55	<code>lst[0]</code>	<code>lst[1]</code>	<code>lst[2]</code>	<code>lst[3]</code>	<code>lst[4]</code>	<p>key</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td> </tr> <tr> <td>11</td><td>22</td><td>33</td><td>44</td><td>55</td> </tr> <tr> <td><code>dic[0]</code></td><td><code>dic[1]</code></td><td><code>dic[2]</code></td><td><code>dic[3]</code></td><td><code>dic[4]</code></td> </tr> </table> <p>value</p> <p>dic</p>	0	1	2	3	4	11	22	33	44	55	<code>dic[0]</code>	<code>dic[1]</code>	<code>dic[2]</code>	<code>dic[3]</code>	<code>dic[4]</code>
0	1	2	3	4																											
11	22	33	44	55																											
<code>lst[0]</code>	<code>lst[1]</code>	<code>lst[2]</code>	<code>lst[3]</code>	<code>lst[4]</code>																											
0	1	2	3	4																											
11	22	33	44	55																											
<code>dic[0]</code>	<code>dic[1]</code>	<code>dic[2]</code>	<code>dic[3]</code>	<code>dic[4]</code>																											

2. ການສົມທຽບລະຫວ່າງ list ແລະ dictionary

2.2. ການລຶບອີງປະກອບໃນ list

| ເນັ້ງການປ່ຽນແປງ `lst[1]` ກ່ອນ ແລະ ຫຼັງລຶບອີງປະກອບໃນ list ດ້ວຍ `pop` method.

```
1 lst = [11, 22, 33, 44, 55]
2 print('before pop(0) :', lst)
3 print('before pop(0) lst[1] = ', lst[1])
4 lst.pop(0)      # Use index 0 to delete the first item of the list
5 print('after pop(0) :', lst)
6 print('after pop(0) lst[1] = ', lst[1])           # The value referred by lst[1] changes
```

before pop(0) : [11, 22, 33, 44, 55] |

before pop(0) list[1] = 22

after pop(0) : [22, 33, 44, 55]

after pop(0) list[1] = 33

| ຖ້າວ່າອີງປະກອບໄດ້ໜຶ່ງຖືກລຶບ, ຈະເຮັດໃຫ້ອີງປະກອບຖຸດໄປປ່ຽນແປງຕຳແໜ່ງຂໍ້ມູນທັນທິ.

2. ការសិរុបលទ្ធផលវាំង list និង dictionary

2.2. ការលើបអីរបៀបរួចរាល់នៅលើ list

- | សិរុបការលើបអីរបៀបរួចរាល់នៅលើ list និង dictionary.
- | ស៉ាងកៅតការលើបអីរបៀបរួចរាល់នៅលើ dictionary ផ្ទៀង dic[1] មិនពិនិត្យបានបញ្ជូនលើកដែលត្រូវបានលើបអីនៅក្នុង code តាមតម្លៃការណ៍នេះ.
- | និងពីរបៀបគ្រប់គ្រងការលើបអីនៅលើ dictionary គឺជាដំឡើងលើកដែលត្រូវបានលើបអីនៅក្នុង code តាមតម្លៃការណ៍នេះ.

```
1 dic = {0:11, 1:22, 2:33, 3:44, 4:55}  
2 # Print the items() function that returns the tuple pair (key-value) of dictionary  
3 print('before pop(0) :', dic.items())  
4 print('before pop(0) dic[1] =', dic[1])  
5 dic.pop(0)      # Items (0, 11) are deleted with the key 0  
6 print('after pop(0) :', dic.items())  
7 print('after pop(0) dic[1] =', dic[1])          The value referred by dic[1] does not change
```

before pop(0) : dict_items([(0, 11), (1, 22), (2, 33), (3, 44), (4, 55)])

before pop(0) dic[1] = 22

after pop(0) : dict_items([(1, 22), (2, 33), (3, 44), (4, 55)])

after pop(0) dic[1] = 22

- | ក្នុងការលើបអីនៅលើ list ឬ dictionary តើវាបានលើបអីនៅលើកដែលត្រូវបានលើបអីនៅក្នុង code តាមតម្លៃការណ៍នេះ.

2. ການສືມທຽບລະຫວ່າງ list ແລະ dictionary

2.3. ສ້າງ list ແລະ dictionary ພ້ອມທັງກວດສອບຄ່າຂອງອີງປະກອບ

Classification	List (uses indices)	Dictionary (uses keys)
ກວດສອບຄ່າຂອງອີງປະກອບ	lst[0]	dic['one']

Classification	List	Dictionary
ສ້າງ object ຫວ່າງ	lst = []	dic = {}
ສ້າງ object ທີ່ມີອີງປະກອບ	lst = [1, 2, 3]	dic = {'one':1, 'two':2}

2. ການສືມທຽບລະຫວ່າງ list ແລະ dictionary

2.4. ການນັບ ແລະ ການລຶບອີງປະກອບໃນ list ແລະ dictionary

| ການກວດສອບຈໍານວນ items ໃນ list ແລະ dictionary ພ້ອມທັງລຶບຄ່າຂຶ້ມູນ

Classification	List	Dictionary
ນັບຈໍານວນອີງປະກອບ	<code>len(lst)</code>	<code>len(dic)</code>
ລຶບອີງປະກອບ	<code>del(lst[0]) or del lst[0]</code>	<code>del(dic['one']) or del dic['one']</code>
ລຶບທັງໝົດອີງປະກອບ	<code>lst.clear</code>	<code>dic.clear</code>

| ການກວດສອບສະເພາະຄ່າໄດ້ໜຶ່ງໃນ list ແລະ dictionary

Classification	List	Dictionary
ກວດສອບຄ່າ	<code>2 in lst</code>	<code>'one' in dic.keys 'one' in dic</code>

3. งานจัดรูบแบบสำลับภาษาพิมอອກຫຼາຈຳ

3.1. ວິທີການຈັດຮູບແບບເພື່ອພິມອອກທາງຫຼາຈຳ

- | ການພິມຂໍ້ຄວາມອອກທາງຫຼາຈຳມີຄວາມຈຳເປັນທີ່ຈະຕ້ອງໄດ້ເຊົ້າໃຈຢ່າງລະອຽດ.
- | ການພິມຂໍ້ຄວາມອອກມາທາງຫຼາຈຳ ຕ້ອງໄດ້ໃຊ້ “format” method ເພື່ອພິມຂໍ້ຄວາມ ຫຼື ຕົວເລກອອກມາ.
- | ເຄື່ອງໝາຍ {} ລຸ່ມນີ້ ແມ່ນຕໍ່າຫຼັງທີ່ຈະໃຫ້ຂໍ້ຄວາມ ຫຼື ຕົວເລກປະກິດອອກມາ, ດັ່ງສະແດງລຸ່ມນີ້.

```
1  '{} Python!'.format('Hello')
```

```
'Hello Python!'
```

```
1  'I like {} and {}'.format('Python', 'Java')
```

```
'I like Python and Java'
```

3. ការចែករូបរោបសំឡែងការពិនិត្យការអនុវត្តន៍

3.2. ការពិនិត្យការអនុវត្យន៍ដោយប្រើប្រាស់ការបង្ហាញខ្លួន

- | ប្រាការបង្ហាញខ្លួនដែលបានរាយការណ៍ដោយ 0, 1, 2, .. ដើម្បីការពិនិត្យការអនុវត្យន៍.
- | {0} ធម៌នាំបង្ហាញខ្លួនដែលបានរាយការណ៍ដោយ 1.
- | {1} ធម៌នាំបង្ហាញខ្លួនដែលបានរាយការណ៍ដោយ 2. តែងតាំងល្អជាដីល្អជាមីន្ទី

```
1 'I like {0} and {1}'.format('Python', 'Java')
```

```
'I like Python and Java'
```

```
1 'I like {1} and {0}'.format('Python', 'Java')
```

```
'I like Java and Python'
```

```
1 'I like {0}, {0}, {0}'.format('Python', 'Java')
```

```
'I like Python, Python, Python'
```

- | {0:4d} ធម៌នាំបង្ហាញខ្លួន 4 digits, {0:5d} ធម៌នាំបង្ហាញខ្លួន 5 digits និង {0:6d} ធម៌នាំបង្ហាញខ្លួន 6 digits

```
1 print('12345678901234567890')
2 print('{0:4d},{0:5d},{0:6d}'.format(123))
```

```
12345678901234567890
```

```
123, 123, 123
```

3. ការត្រួតពេលវេលាភាសាទិន្នន័យ

3.3. ការកំណត់រាយការណ៍ field សំឡើរការពិនិត្យការងារ

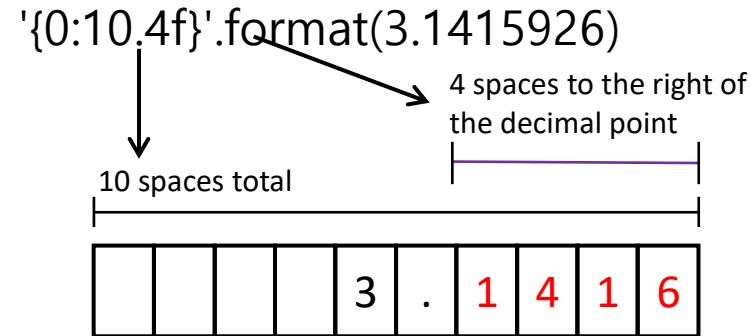
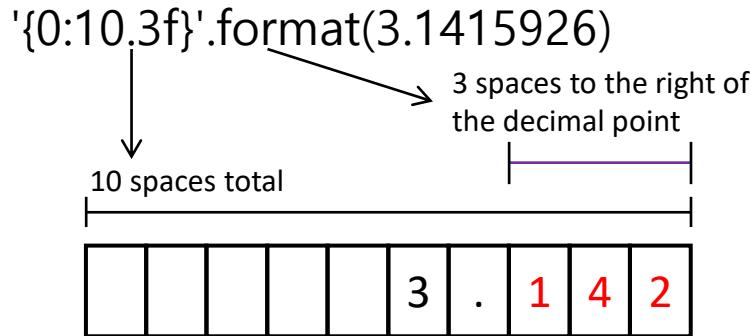
- | នឹងសាមុទ្ធកំណត់លក្ខណៈជូនដោយ ផ្តល់ចំណាំ: `{0:.2f}`, `{0:.3f}` ឬមែនការពិនិត្យលក្ខណៈតុលាសងញាំង និង សាមត្រាំងញាំង។
- | “f” ឬមែនអាជីវិតិវត្ថុខ្លួនដែលត្រូវបាន `float` . ដំឡើង ការងារកំណត់លក្ខណៈជូននៃការណ៍ `field` សាមុទ្ធកំណត់លក្ខណៈជូនដោយ ផ្តល់ចំណាំ: `{0:5.2f}` ឬ `{0:6.2f}`.

1 'សាមុទ្ធកំណត់លក្ខណៈជូនទាំង 3 និង សាមពួលដោយ: `{0:.2f},{0:3f}`'.format(1/3)

'ិនໄតឱរបភ័ណ្ឌ: 0.33 និង 0.333'

1 'ចំណាត់ថ្នាក់សំឡើរ 10 spaces : `{0:10.3f},{0:10.4f}`'.format(3.1415926)

'ិនໄតឱរបភ័ណ្ឌ : 3.142, 3.1416'



3. ការត្រួតពេលវេលាប្រភេទការពិន័យខ្លួន

3.3. ការកំណត់រាយការពិន័យខ្លួន field សំគាល់ការពិន័យខ្លួន

- | សំគាល់ការពិន័យ integer, ត្រូវកំណត់រាយការពិន័យខ្លួនពីលេខសំគាល់ការពិន័យ ដោយនឹង {1:3d}, {1:4d}, {1:5d}.
- | តាមរបៀបដែលបានបង្ហាញឡើង ត្រូវបានកំណត់រាយការពិន័យខ្លួនដោយការកំណត់រាយការពិន័យខ្លួន field លើ 3, 4 និង 5 តាមលំដាប់.
- | ឯងត្រូវបានបង្ហាញឡើងដោយលើមីនី.

```
1 for i in range(2, 11, 2):
2     print('{0:3d} {1:4d} {2:5d}'.format(i, i*i, i*i*i))
3
```

```
2      4      8
4     16     64
6     36    216
8     64    512
10   100   1000
```

| Pair programming



Pair Programming Practice

| ແນວທາງ, ກົມໄກ ແລະ ແຜນສຸກເສີນ

ການຈັບຄຸ້ຂຽນໂປຣແກຣມ ເປັນການຈັບຄຸ້ຂອງນັກຮຽນເພື່ອຮັດວຽກມອບໝາຍ, ນັກຮຽນຄວນມີແຜນ ແລະ ສາມາດປ່ຽນແທນກັນໄດ້ ໃນ ກໍາລະນີມີຜູ້ໃຫ້ນີ້ບໍ່ສາມາດເຂົ້າຮ່ວມຮັດວຽກມອບໝາຍໄດ້ບໍ່ວ່າໃນກໍາລະນີໃດກຳຕາມ ເຊິ່ງບັນຫາເລື່ອນັ້ນຕ້ອງຮັດໃຫ້ຈະເຈັ້ງ ແລະ ກຳບໍ່ແມ່ນ ຄວາມຜິດຂອງນັກຮຽນທີ່ຈັບຄຸ້ບໍ່ດີ.

| ຈັບຄຸ້ທີ່ຄ້າຢັກນັ້ນ, ບໍ່ຈໍາເປັນເຫົ້າທຽມກັນ, ຄວາມສາມາດເປັນຄຸ້ຮ່ວມງານ

ການຈັບຄຸ້ຂຽນໂປຣແກຣມ ຈະໄດ້ຮັບຜົນທີ່ກຳຕໍ່ເນື້ອນັກຮຽນມີຄວາມສາມາດຄ້າຢັກນັ້ນ ຫາຍວ່າ ບໍ່ຈໍາເປັນຕ້ອງມີຄວາມສາມາດຄ້າຢັກນັ້ນກໍໄດ້, ແຕ່ວ່າ ການຈັບຄຸ້ ນັກຮຽນທີ່ມີຄວາມສາມາດແຕກຕ່າງກັນຫຼາຍ ກໍຈະຮັດໃຫ້ບໍ່ສົມດຸນກັນ. ຄຸສອນຮູ້ດີວ່າ ການຈັບຄຸ້ກັນບໍ່ແມ່ນ ຍຸດທະສາດ “ແບ່ງເພື່ອເອົາຊະນະ” ແຕ່ເປັນຄວາມ ພະຍາຍາມຮັດວຽກຮ່ວມກັນຂອງນັກຮຽນໃຫ້ປະສົບຜົນສໍາເລັດ. ຄຄວນທີ່ກາເວັ້ນການຈັບຄຸ້ກັນລະຫວ່າງນັກຮຽນອ່ອນ ແລະ ນັກຮຽນເກົ່າງ.

| ກະຕຸ້ນນັກຮຽນໂດຍການໃຫ້ສິ່ງຈຸງໃຈພິເສດ

ຂໍສະເໜີແຮງຈຸງໃຈທີ່ຮັດໃຫ້ນັກຮຽນຈັບຄຸ້, ໂດຍສະເພາະນັກຮຽນທີ່ມີຄວາມສາມາດສຸງ. ບາງຄຸສອນໄດ້ພິບວ່າ ການຈັບຄຸ້ຮັດວຽກມອບໝາຍ ແມ່ນມີ ປະໂຫຍດ ສໍາລັບໜຶ່ງ ຫຼື ສອງວຽກມອບເທົ່ານັ້ນ



Pair Programming Practice

| ចំណាំការបំពើនិងរុញខែងមករុញ

ສິ່ງທ້າທາຍສໍາລັບຄູແມ່ນເພື່ອຊອກຫາວິທີທີ່ຈະປະເມີນຜົນການຮຽນຂອງນັກຮຽນ, ຄູຮູ້ບໍ່ວ່າ ນັກຮຽນ ໄດ້ຕັ້ງໃຈຮຽນ ຫຼື ບໍ່ຕັ້ງໃຈຮຽນ. ຜູ້ສ່ວຽດຊານໄດ້ແນະນຳໃຫ້ທົບທວນການອອກແບບຫຼັກສຸດການຮຽນ ແລະ ຮູບແບບການປະເມີນ ພ້ອມທັງປີກສາຫາລືຢ່າງຈົງຈັງກັບນັກຮຽນ ກ່ຽວກັບພິດຕິກຳທີ່ຈະບໍ່ຕັ້ງໃຈຮຽນ ນອກຈາກນີ້ຢັ້ງໄດ້ແນະນຳມອບວຽກມອບໝາຍໃຫ້ນັກຮຽນ ພ້ອມທັງອະທິບາຍໃຫ້ເຂົ້າເຈົ້າຢ່າງຈະເຈັ້ງ

| ສະພາບແວດລ້ອມຂອງການຮຽນຮັກ

ສະພາບແວດລ້ອມການຮຽນຮູ້ຮ່ວມກັນເກີດຂຶ້ນໄດ້ທຸກເວລາທີ່ຜູ້ສອນຮຽກຮ້ອງໃຫ້ນັກຮຽນເຮັດວຽກຮ່ວມກັນໃນກິດຈະກຳການຮຽນຮູ້ ເຊິ່ງອາດຈະເປັນກິດຈະກຳທີ່ເປັນທາງການ ແລະ ບໍ່ເປັນທາງການ ແລະ ອາດຈະບໍ່ລວມເຖິງການປະເມີນຜົນການຮຽນໂດຍກົງ. ເຊັ່ນຕົວຢ່າງ ໃຫ້ນັກຮຽນຈັບຄຸ້ງກັນເພື່ອເຮັດວຽກມອບໝາຍ ໂດຍນັກຮຽນຈະຕ້ອງທີບທວນກ່ຽວກັບການສອນຂອງອາຈານທີ່ຜ່ານມາ ແລະ ລະດົມແນວຄິດພາຍໃນກຸ່ມ ພ້ອມທັງມືການແບ່ງວຽກໃຫ້ແຕ່ລະຄົນຮັບຜິດຊອບ ຈາກນັ້ນກຳໃຫ້ມີການແລກປ່ຽນຄວາມຄິດເຫັນເຊິ່ງກັນ ແລະ ກັນ ເພື່ອເຮັດວຽກມອບໝາຍໃຫ້ສໍາເລັດຕາມເປົ້າໝາຍທີ່ວ່າງໄວ້.

Q1. ស້າງ dictionary ທີ່ຊື່ fruits_dic ທີ່ປະກອບດ້ວຍ key-value ດັ່ງນີ້ ('apple', 6000), ('melon', 3000), ('banana', 5000), ('orange', 4000). ຈາກນີ້ນ ໃຫ້ພິມຜົນຮັບດັ່ງສະແດງລຸ່ມນີ້.

Output example

```
dict_keys(['apple', 'melon', 'banana', 'orange'])
apple is in fruits_dic.
mango is not in fruits_dic.
```