

Unit 14.

Dictionary Method 1

Learning objectives

- ✓ ສາມາດພິ້ມ ຫຼື ແກ້ໄຂຄ່າໃໝ່ ໂດຍນຳໃຊ້ dictionary key.
- ✓ ສາມາດຄື້ນຫາ ແລະ ລຶບບັນດາ item ພາຍໃນ dictionary.
- ✓ ສາມາດລຶບທັງໝົດ item ພາຍໃນ dictionary ໂດຍນຳໃຊ້ clear method.
- ✓ ສາມາດແຍກ key-value ໂດຍນຳເວີທິການອ້າງອີງ key ໃນ dictionary.

Learning overview

- ✓ เพิ่ม และ แก้ไขข้อมูล dictionary โดยนำໃຊ້ key-value.
- ✓ ການລຶບຄ່າ โดยນຳໃຊ້ວິທີການລຶບສຸມອົງປະກອບໃນ dictionary ດ້ວຍ key-value.
- ✓ ລຶບທັງໝົດ โดยນຳໃຊ້ method ການລຶບອົງປະກອບທັງໝົດໃນ dictionary ດ້ວຍ key-value.
- ✓ ບັນທຶກ dictionary key-value ໄປຕົວປ່ຽນອື່ນໂດຍໃຊ້ວິທີການ key-value ຂອງ dictionary

Concepts You Will Need to Know From Previous Units

- ✓ ເຂົ້າໃຈການເຂົ້າເຖິງຂໍ້ມູນແບບ dictionary ໄດ້ໂດຍໃຊ້ keys ດີກວ່າໃຊ້ ຕໍາແໜ່ງຂໍ້ມູນ (index).
- ✓ ສາມາດເຂົ້າເຖິງ ແລະ ກຳນົດຄ່າໃຫ້ກັບ dictionary key ແລະ ດຳເນີນການ dictionary
- ✓ ເຂົ້າໃຈຄວາມແຕກຕ່າງລະຫວ່າງ list ແລະ dictionary ພ້ອມທັງສາມາດປະກາດ ແລະ ນຳໃຊ້ປະເພດຂໍ້ມູນ (ເຊັ່ນ: list, dictionary) ຕາມສະຖານະການໄດ້.

Keywords

Dictionary

Key-Value

Add, Edit

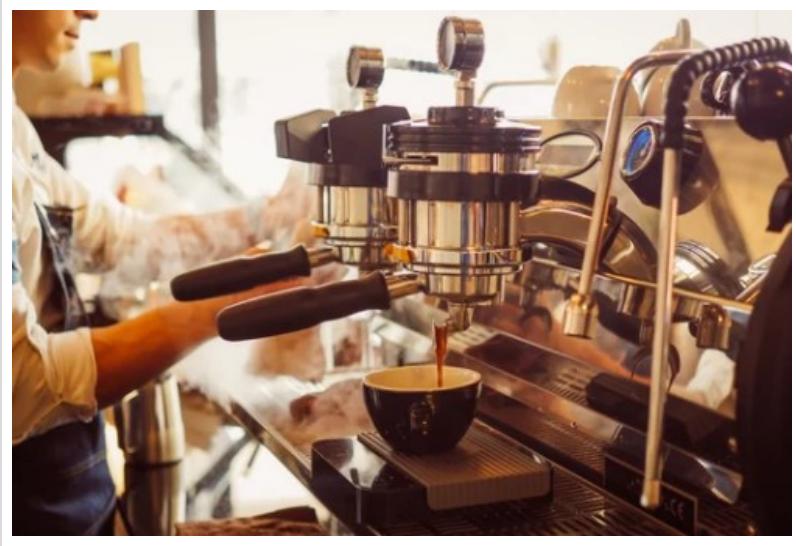
Delete

Save

Mission

1. Real world problem

1.1. ບັນຫາຂອງຜູ້ປະກອບການໃນຮ້າກາເຟ ແລະ ຮ້ານອາຫານ ເນື່ອງຈາກລາຄາວັດຖຸດີບ ແລະ ຄ່າແຮງງານສູງຂຶ້ນ



- ▶ ເຊດການຄ້າທ້ອງຖິ່ນທີ່ມີຜູ້ປະກອບອາຊີບອິດສະຫຼະ ເຊັ່ນ: ຮ້ານກາເຟ ແລະ ຮ້ານອາຫານ ເປັນກິດຈະການທາງດ້ານເສດຖະກິດຂອງຜູ້ອາໄສໃນທ້ອງຖິ່ນດັ່ງກ່າວໜີຍົມເຮັດ.
 - ▶ ເນື່ອງຈາກລາຄາວັດຖຸດີບ ແລະ ຄ່າແຮງງານສູງຂຶ້ນຢ່າງຕໍ່ເນື່ອງຈາດຈະເຮັດໃຫ້ການບໍລິຫານຈັດການຂອງຜູ້ປະກອບການມີຄວາມລໍາບາກ.
 - ▶ ຕ້ອງໃຊ້ຄວາມພະຍາຍາມຫຼາຍໃນການຫຼຸດຕົ້ນທຶນການດຳເນີນງານ ແລະ ຫັນມາພັດທະນາໂປຣແກຣມ ເພື່ອຊ່ວຍໃນການດຳເນີນງານເລີ່ມ້ນ.
 - ▶ ຈາກ mission ກ່ອນໜັນນີ້ ເຊິ່ງໄດ້ຮູບການພິມເມນູກາເຟ. ສໍາລັບ mission ປັບປຸງຝັງຊັ້ນກ່ອນໜັນນີ້ເພື່ອສ້າງຝັງຊັ້ນເພີ່ມເມນູໃໝ່ ແລະ ລາຄາ ຂອງມັນ.

1. Real world problem

1.2. ໂປຣແກຣມຈະບອກລາຄາ ເມື່ອເລືອກເມນຸຂັ້ນມາ



- ▶ ການຈ້າງພະນັກງານໃໝ່ໃນຮ້ານກາເຟ ຕ້ອງໃຊ້ເວລາ ແລະ ຕຶງເປັນຈຳນວນຫຼາຍໃນການຝຶກອົບຮົມໃຫ້ພວກເຂົາໃຫ້ເຂົ້າໃຈເພື່ອໃຊ້ໂປຣແກຣມດັ່ງກ່າວ.
- ▶ ເມນຸ ແລະ ລາຄາ ຈະປ່ຽນແປງໄປຕາມສະຖານະການ ແຊັນ ເມນຸອາດຈະມີການເພີ່ມ ຫຼື ລຶບອອກຕາມລະດຸການ ແລະ ໄຕມາດ.
- ▶ ດ້ວຍເຫດນີ້ ຈຶ່ງຈໍາເປັນຕ້ອງສ້າງໂປຣແກຣມສໍາລັບຮ້ານກາເຟ ໂດຍໂປຣແກຣມດັ່ງກ່າວ ຕ້ອງແຈ້ງລາຄາໃຫ້ກັບລູກຄ້າຮູ້ທັນທີເມື່ອເອັນເມນຸຂັ້ນມາ.
- ▶ ເຮົາຕ້ອງຜູກຕິດລາຄາໃສ່ກັບເມນຸ ເພື່ອໃຫ້ສະແດງລາຄາ ເວລາເອັນໃຊ້ເມນຸ.
- ▶ ມາສ້າງໂປຣແກຣມຮ້ານກາເຟໃຫ້ສາມາດຮັດວຽກໄດ້ຕາມທີ່ເວົ້າມາຂ້າງເທິງ.

1. Real world problem

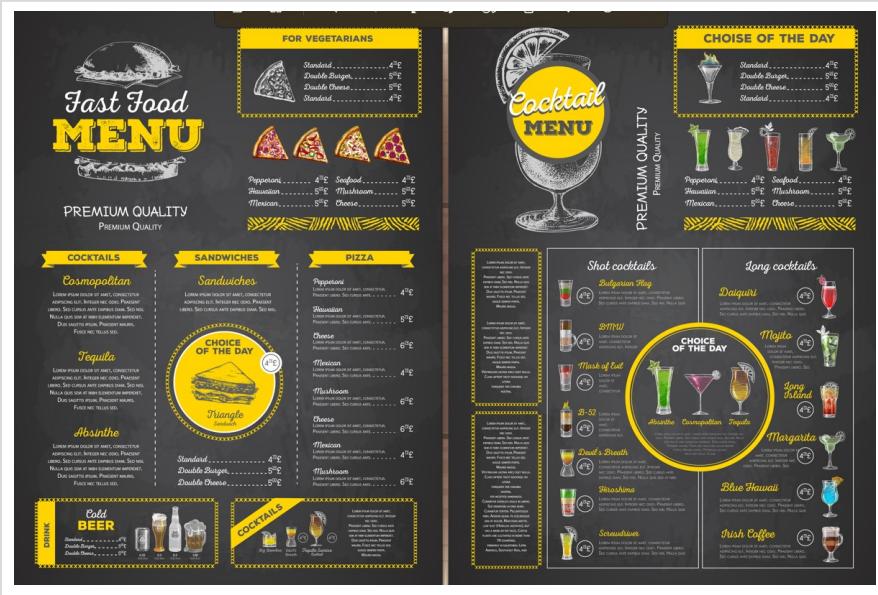
1.3. ນີ້ເວດວິທະຍາຂອງຮ້ານກາເຟ



- ▶ ສະຖານທີ່ ແລະ ທີ່ຕັ້ງຂອງການດື່ມກາເຟ, ຊາ ແລະ ອື່ນໆ. ໃນປະເທດຝຣັ້ງ, ກາເຟຖືກເອັ້ນວ່າ cafe, **ຊື່ມີ hardened into a "house selling coffee."**
- ▶ ປະຈຸບັນ ຮ້ານກາເຟໄດ້ຮັບຄວາມນີ້ຍົມຫຼາຍ ຄືນສ່ວນຫຼາຍໄປຮ້ານກາເຟເພື່ອອ່ານຫັງສີ, ໃຊ້ອິນເຕີເນັດ.
- ▶ ຜູ້ປະກອບການເຄືອຂ່າຍຮ້ານກາເຟລະດັບໂລກ ເຊັ່ນ Starbucks ກໍາລັງໄດ້ຮັບຄວາມນີ້ຍົມໃນປະເທດຕ່າງໆ ໂດຍສະພາະປະເທດ ເກົ້າຫຼື ແລະ ຢື່ບຸນ ດ້ວຍກົນລະບຸດຂັ້ນສູງ ແລະ ປະສິບການ.
- ▶ ປະຈຸບັນ Starbucks ໄດ້ໃຫ້ບໍລິການສໍາລັບ mobile era ເຊັ່ນວ່າ SIREN Order ແລະ ເປັນຜູ້ນໍານະວັດຕະກຳໃນການສັ່ງຊື້.

2. Solution

2.1. Café menu program



- ▶ ใน mission นี้, ป้อนเมนูในโปรแกรม cafe และ พิมลาຄาอອກมา. ນອກຈາກນີ້ จะສ້າງຝ່ານບ້ອນເມນູໃໝ່ ພ້ອມລາຄາ.
- ▶ ຄໍາສັ່ງເລົ່ານີ້ຈະພິມເຂົ້າໄປໃນ Command Prompt.
- ▶ ใน command prompt '\$' ຈະປະກິດໜ້າຕ່າງການດໍາເນີນການ ເຊັ່ນວ່າ ຄໍາສັ່ງບ້ອນຂໍ້ມູນຈະສະແດງເຖິງໝາຍ '<' ສ່ວນການຄົ້ນຫາຈະສະແດງເຖິງໝາຍ >!
- ▶ Input ກຳຄົງ menu: price form. ບ້ອນ menu ເພື່ອຄົ້ນຫາພວກມັນ.
- ▶ ຖ້າຕ້ອງການອອກຈາກ cafe menu program ໃຫ້ບ້ອນ q.

3. Mission

3.1. ລຸ່ມນີ້ແມ່ນການເຮັດວຽກຂອງ café menu program

```
1 print("Start café menu program... Press q to exit")
2 cafe_menu = {'Ice coffee' : 3000}
3
4 while True:
5     str = input('$ ')
6     if str.startswith('q'):
7         break
8     command = str[0]
9
10    if command == '<':
11        str = str.replace('<', "")
12        inputStr = str.split(':')
13        if len(inputStr) < 2 :
14            print('input error')
15            continue
16        else:
17            cafe_menu[inputStr[0].strip()] = inputStr[1].strip()
18
19    elif command == '>':
20        str = str.replace('>', "")
21        inputStr = str.strip()
22        if inputStr in cafe_menu:
23            print(cafe_menu[inputStr])
24        else :
25            print("{} not in the menu.".format(inputStr))
26    elif command == 'q':
27        break
28    else :
29        print('input error.')
30
print("exiting café menu program.")
```

3. Mission

3.1. ລຸ່ມນີ້ແມ່ນການເຮັດວຽກຂອງ café menu program

```
Start café program... Enter q to exit
```

```
$ > Ice coffee
```

```
3000
```

```
$ > green tea
```

```
Green tea is not on the menu.
```

```
$ < espresso : 5000
```

```
$ espresso
```

```
Input error
```

```
$ > espresso
```

```
5000
```

```
$ q
```

```
Exit the café program.
```

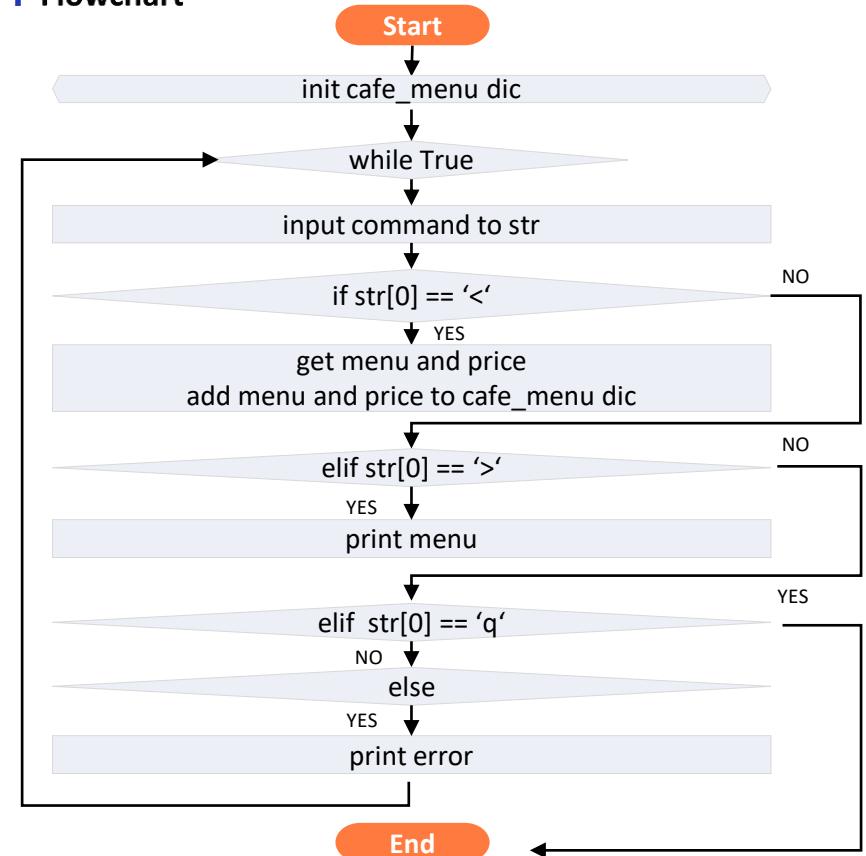
3. Mission

3.2. Programming plan

Pseudocode

```
[1] Start  
[2] Reset café menu dictionary.  
[3] while True  
    [4] Input command and café menu.  
        [5] if the first command is '<'(Input), then  
            [6] Get menu and price from the string input  
                Add menu and price to café dictionary.  
            [7] else if first command'>'(output), then  
                [8] Print the menu.  
  
        [9] else if the first command is 'q',  
            [10] exit the program [13] End  
  
        [11] else print error  
  
    [12] Exit program  
  
[13] End
```

Flowchart



3. Mission

3.3. final code 29] Café menu program

```
1 print("Start café menu program... Press q to exit")
2 cafe_menu = {'Ice coffee' : 3000}
3
4 while True:
5     str = input('$ ')
6     if str.startswith('q'):
7         break;
8     command = str[0]
9
10    if command == '<':
11        str = str.replace('<', "")
12        inputStr = str.split(':')
13        if len(inputStr) < 2 :
14            print('input error.')
15            continue
16        else:
17            cafe_menu[inputStr[0].strip()] = inputStr[1].strip()
18
19    elif command == '>':
20        str = str.replace('>', "")
21        inputStr = str.strip()
22        if inputStr in cafe_menu:
23            print(cafe_menu[inputStr])
24        else :
25            print("{} not in the menu.".format(inputStr))
26    elif command == 'q':
27        break
28    else :
29        print('input error.')
30 print("exiting café menu program.")
```

| Key concept

1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.1. ເພີ່ມ ແລະ ແກ້ໄຂ ຂໍ້ມູນປະເພດ dictionary

- | ສາມາດເພີ່ມ ແລະ ລຶບຂໍ້ມູນໃນ dictionary ໄດ້ຕາມຕ້ອງການ.
- | ການເພີ່ມ key-value ໃນ dictionary ເປັນສິ່ງສໍາຄັນ
- | ວິທີການເພີ່ມ key-value ມີຮູບແບບດັ່ງນີ້ dic[key] = value.
- | ຢ່າງໃດກໍຕາມ, ບາງກໍລະນີຈໍາເປັນຕ້ອງເພີ່ມ ຫຼື ປັບປຸງຄ່າເລີ່ມຕົ້ນ ຖ້າຈໍາເປັນ.
- | ມາເປົ່ງຂັ້ນຕອນຂອງ dictionary ໃນການກຳນົດຄ່າເລີ່ມຕົ້ນ (setdefault) ແລະ ປັບປຸງມັນ(update).
 - ▶ setdefault : ເພີ່ມ default key-value
 - ▶ update : ແກ້ໄຂຄ່າຂອງ key ແລະ ເພີ່ມ key-value ທັງບໍ່ມີ key.

1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.2. Dictionary code ປັດສະຈາກ setdefault

- | code ດຳລັງລຸ່ມນີ້ເປັນ code ມັບຈຳນວນຕົວອັກສອນໃນ list ທີ່ຊື່ str1.
- | ໃນນີ້ a ມີສອງຄ່າ ສ່ວນ b ມີສາມຄ່າ ແລະ c ມີ ສອງຄ່າ. ໄສຂໍ້ມູນໃນ dictionary ແລະ ພິມມັນອອກມາ.

```
1 str1 = ['a', 'b', 'c', 'b', 'a', 'b', 'c']
2 dic = {}
3 for ch in str1:
4     if ch not in dic.keys():
5         dic[ch] = 0
6         dic[ch] += 1
7
8 print('alphabet counting :', dic)
```

```
alphabet counting : {'a': 2, 'b': 3, 'c': 2}
```

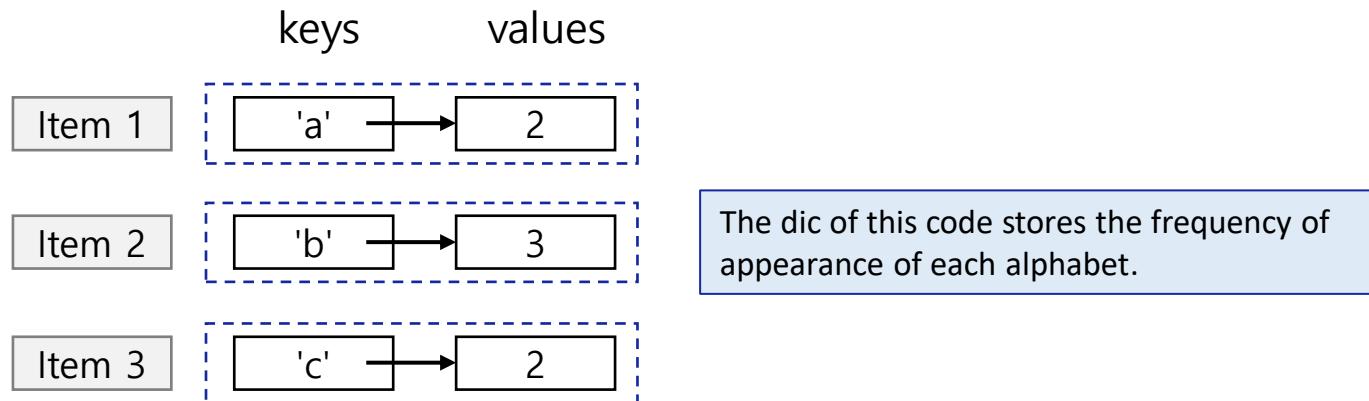
Line 4, 5, 6

- ກວດສອບວ່າ ch ຢູ່ໃນ key ຂອງ dic ຫຼື ບໍ່.
- ຖ້າບໍ່, ໃຫ້ພື້ນ key ແລະ ກຳນົດຄ່າເລີ່ມຕົ້ນເປັນສູນ.
- ເພີ່ມ 1 ໃຫ້ກັບ key ຂອງ dic.

1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.2. Dictionary code ປັດສະຈາກ setdefault

- | ໃນໂຄງສ້າງຂອງ dictionary, ຈະມີການກຳນົດຄ່າເລີ່ມຕົ້ນໃຫ້ກັບ key.
- | ຢ່າງໄດ້ກຳຕາມ dictionary ທີ່ໃຊ້ໃນນີ້ ແມ່ນ dictionary ຫວ່າງເປົ່າ ສະນັ້ນຈຶ່ງບໍ່ມີຄ່າເລີ່ມຕົ້ນ.
- | ໃນການນຳໃຊ້ dictionary, ຈໍາເປັນຕົງກວດສອບວ່າມີຕົວອັກສອນ ch ຢູ່ໃນ dictionary key ຫຼື ບໍ່, ຖ້າຢູ່ໃຫ້ກຳນົດຄ່າເລີ່ມຕົ້ນດ້ວຍ `dic[i] = 0`
- | ມາເບິ່ງ `setdefault` ທີ່ຮັດໃຫ້ການເຮັດວຽກຂອງ dictionary ມີຄວາມສະດວກຂຶ້ນ.



1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.3. Dictionary code ກັບ setdefault

| ໂຄງສາງຂຶ້ນແບບ dictionary ມີ method ຫຼື້ນ setdefault

| ຖ້າວ່າ key value ແມ່ນ factor ທຳອິດ ແລະ default value ແມ່ນ factor ທີ່ສອງ ດັ່ງສະແດງລຸ່ມນີ້, default value ສໍາລັບ key ແມ່ນກຳນົດເປັນ 0.

```
1 str1 = ['a', 'b', 'c', 'b', 'a', 'b', 'c']
2 dic = {}
3 for ch in str1:
4     dic.setdefault(ch, 0)
5     dic[ch] += 1
6     print(list(dic.items()))
7
8 print('alphabet counting :', dic)
```

```
[('a', 1)]
[('a', 1), ('b', 1)]
[('a', 1), ('b', 1), ('c', 1)]
[('a', 1), ('b', 2), ('c', 1)]
[('a', 2), ('b', 2), ('c', 1)]
[('a', 2), ('b', 3), ('c', 1)]
[('a', 2), ('b', 3), ('c', 2)]
alphabet counting : {'a': 2, 'b': 3, 'c': 2}
```



Line 4, 5, 6

- setdefault ຈະສື່ງ key ຊັ້ນມີ key ແລະ ຈະສື່ງ factor ທີ່ສອງທ້າບໍ່ມີ key .

1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.3. Dictionary codeກັບ setdefault

| ໂຄງສໍາຂໍ້ມູນແບບ dictionary ມີ method ທີ່ຊື່ setdefault

| ຖ້າວ່າ key value ແມ່ນ factor ທໍາອິດ ແລະ default value ແມ່ນ factor ຫີສອງ ດັ່ງສະແດງລຸ່ມນີ້, default value ສໍາລັບ key ແມ່ນກຳນົດເປັນ 0.

```

1 str1 = ['a', 'b', 'c', 'b', 'a', 'b', 'c']
2 dic = {}
3 for ch in str1:
4     dic.setdefault(ch, 0)
5     dic[ch] += 1
6     print(list(dic.items()))
7
8 print('alphabet counting :', dic)
```

```

[('a', 1)]
[('a', 1), ('b', 1)]
[('a', 1), ('b', 1), ('c', 1)]
[('a', 1), ('b', 2), ('c', 1)]
[('a', 2), ('b', 2), ('c', 1)]
[('a', 2), ('b', 3), ('c', 1)]
[('a', 2), ('b', 3), ('c', 2)]
alphabet counting : {'a': 2, 'b': 3, 'c': 2}
```



Line 4, 5, 6

- ດັ່ງນັ້ນ ຈຶ່ງເປັນໄປໄດ້ທີ່ຈະເຫັນຜົນຮັບຂອງຕົວອັກສອນໃຫ້ການຝຶກກຳນົດເປັນສູນໄດ້ຜ່ານ dic.setdefault(ch, 0).

1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.3. Dictionary codeກັບ setdefault

| ໂຄງສາງຂໍ້ມູນແບບ dictionary ມີ method ທີ່ຊື່ setdefault

| ຖ້າວ່າ key value ແມ່ນ factor ທໍາອິດ ແລະ default value ແມ່ນ factor ຫີສອງ ດັ່ງສະແດງລຸ່ມນີ້, default value ສໍາລັບ key ແມ່ນກຳນົດເປັນ 0

```

1 str1 = ['a', 'b', 'c', 'b', 'a', 'b', 'c']
2 dic = {}
3 for ch in str1:
4     dic.setdefault(ch, 0)
5     dic[ch] += 1
6     print(list(dic.items()))
7
8 print('alphabet counting :', dic)
```

```

[('a', 1)]
[('a', 1), ('b', 1)]
[('a', 1), ('b', 1), ('c', 1)]
[('a', 1), ('b', 2), ('c', 1)]
[('a', 2), ('b', 2), ('c', 1)]
[('a', 2), ('b', 3), ('c', 1)]
[('a', 2), ('b', 3), ('c', 2)]
alphabet counting : {'a': 2, 'b': 3, 'c': 2}
```



Line 4, 5, 6

- Key value ຈະປ່ຽນແປງທຸກຄັ້ງເມື່ອມີການປ້ອນໂຄຍັກສອນຜ່ານ print statement.

1. เพิ่ม และ แก้ไข Key-Value

1.3. Dictionary code กับ setdefault

- | ถ้า key และ ค่าเลี่มตื้น ทີກກຳນົດ, ເຊັ່ນ setdefault(key, value), ค่าເລື່ມຕົ້ນຈະຖືກແກ້ບໄວ້ໃນຄ່າ ແລະ ຄ່ານີ້ ຈະຖືກສິ່ງກັບ.
- | ຈາກຕົວຢ່າງລຸ່ມນີ້, ເປັນການເພີ່ມ key ທີ່ຊື່ 'f' ເຂົ້າໄປ dictionary ພ້ອມເອັນ dic, ທີ່ແກ້ບຄ່າ 50 ເປັນຄ່າເລື່ມຕົ້ນ ແລະ ພິມມັນອອກນາ.

```

1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 dic.setdefault('f', 50)
3 print(dic)

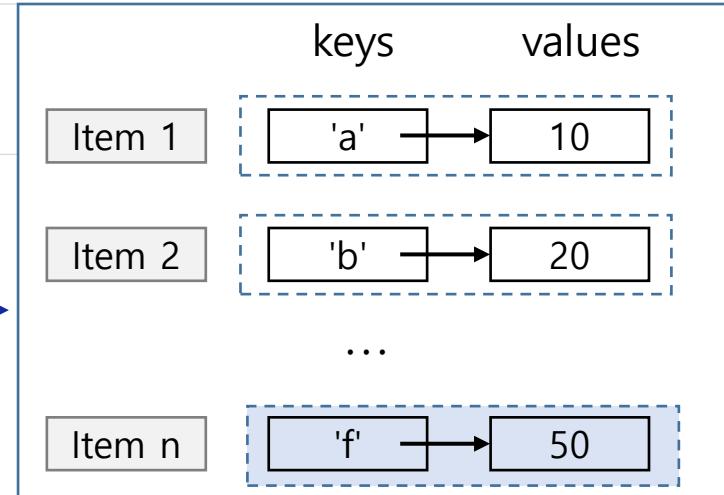
```

{'a': 10, 'b': 20, 'c': 30, 'd': 40, 'f': 50}

Line 2

- Since there is no key 'f', add it and put the value as 50.

The default value for the
'f' key of this dic is 50.



1. ផែន និង របៀប Key-Value

1.4. ការប្រើប្រាស់ value នៅក្នុង key.

- | វិធីការ update(key=value) មែនការប្រើប្រាស់តាម key នៃ dictionary.
- | តើ dictionary ទាំងនេះ dic = {'a' : 10} (មិន key បាន string), ត្រូវដោយតួចតាមលក្ខណៈ. នៅ dic.update (a = 90), សៀវភៅ key និង values ត្រូវបានផ្តល់នៅក្នុងមាយវូរិយ័យ.
- | នៃកំលែននេះ, value នៅក្នុង key 'a' នឹងត្រូវរៀបចំជាបែង 90 ត្រូវដោយតួចតាមលក្ខណៈ.

```

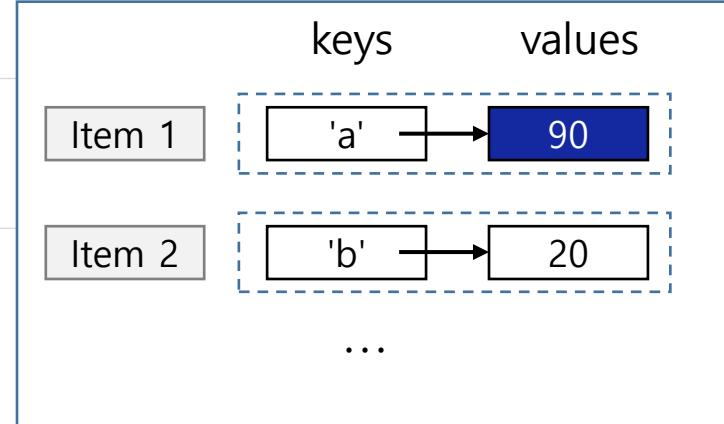
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 dic.update(a = 90)
3 print(dic)

```

{'a': 90, 'b': 20, 'c': 30, 'd': 40}

Line 2

- ប្រើប្រាស់ 90.



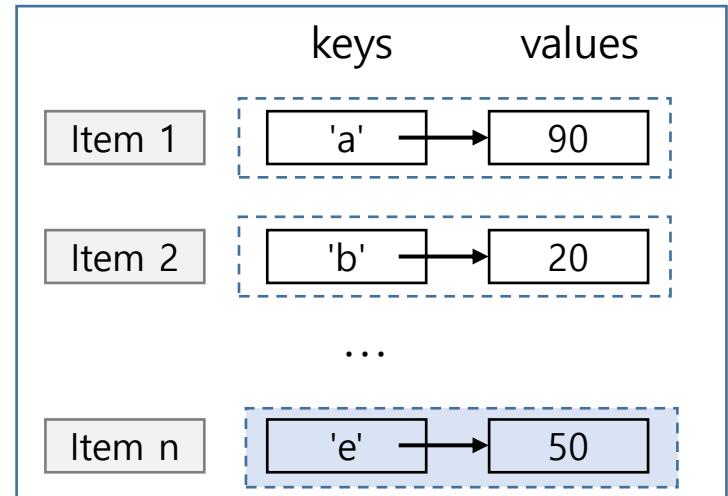
1. เพิ่ม และ แก้ไข Key-Value

1.5. งาน update และ งานเพิ่ม key เข้าไปใหม่

- | ถ้าบ้ม key ใน dictionary, update method ยังสามารถเพิ่ม key-value เข้าไปใหม่ได้.
- | ถ้าจะเพิ่ม key 'e' ปุ่ม key 'e' ยังไม่ใน dic, สมมั่น ท้าต้องการเพิ่ม ต้องประทับดั้งนี้ dic.update (e=50) .

```
1 dic.update(e = 50)  
2 print(dic)
```

```
{'a': 90, 'b': 20, 'c': 30, 'd': 40, 'e': 50}
```



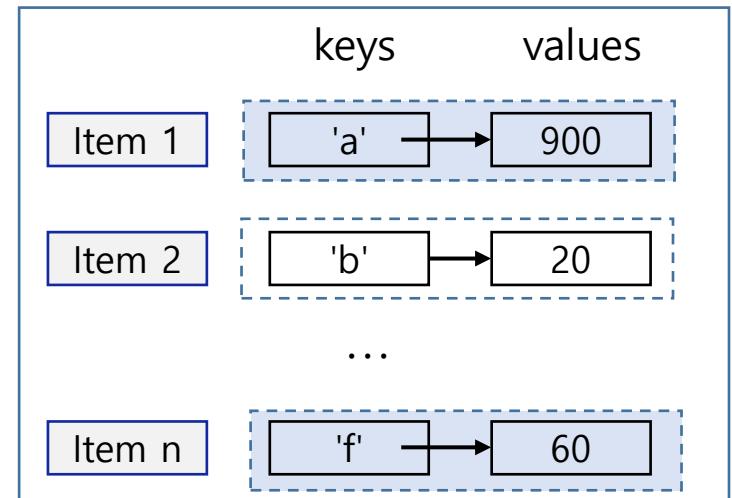
1. เพิ่ม และ แก้ไข Key-Value

1.6. การแก้ไขข้อมูล items ด้วย update method

- | สามารถแก้ไขข้อมูลได้ผ่านการใช้จุดขั้นตอนที่ว่า key-value เลื่อนั้น.
- | ในนี้, ถ้ามี key ใน dictionary, การแก้ไขจะใส่ value ใหม่ให้กับ key ดังก่อ, แต่ถ้าไม่มีเพิ่มทั้ง key และ value เข้าไปได้เลย.
- | ลุமน์เป็นการเพิ่ม value=900 ให้กับ key 'a' และ เพิ่ม 'f': 60.

```
1 dic.update(a = 900, f = 60)
2 print(dic)
```

```
{'a': 900, 'b': 20, 'c': 30, 'd': 40, 'e': 50, 'f': 60}
```



1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.7. ຄວນລະວັງການນຳໃຊ້ update method

 Focus update (key=value) ໃຊ້ໄດ້ສະເພາະ key ເປັນ string.

| ທີ່ key ເປັນ number, value ອາດຈະຖືກປັບປຸງຕ້ອງການແຊກ dictionary ດັ່ງນີ້ update(dictionary).

```
1 dic2 = {1: 'one', 2: 'two'}  
2 dic2.update({1: 'ONE', 3: 'THREE'})  
3 print(dic2)
```

```
{1: 'ONE', 2: 'two', 3: 'THREE'}
```

1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.8. ການປ່ຽນແປງໂດຍໃຊ້ update method ເມື່ອ key ມັນບໍ່ເປັນ string

- | ແນວທາງໃນການປ່ຽນແປງ key-value ແມ່ນໃຊ້ໃນ list ແລະ tuple.
- | Update (list) ແລະ update (tuple) ຈະປ່ຽນແປງຄ່າໃນ list ແລະ tuple.
- | ໃນນີ້, list ຈະສ້າງ list ທີ່ປະກອບດ້ວຍ keys ແລະ values ດັ່ງນີ້ [Key1, Value1] ແລະ [Key2, Value2]]
- | ສໍາລັບ Tuple ກໍາເຊັ້ນດຽວກັບ List.

List

```
1 dic2.update([[2, 'TWO'], [4, 'FOUR']])
2 print(dic2)
```

```
{1: 'ONE', 2: 'TWO', 3: 'THREE', 4: 'FOUR'}
```

Tuple

```
1 dic2.update((2, 'TWO'), (4, 'FOUR'))
2 print(dic2)
```

```
{1: 'ONE', 2: 'TWO', 3: 'THREE', 4: 'FOUR'}
```

1. ເພີ່ມ ແລະ ແກ້ໄຂ Key-Value

1.9. ຄວາມແຕກຕ່າງລະຫວ່າງ setdefault ແລະ update

- | Setdefault ສາມາດເພີ່ມໄດ້ແຕ່ key-value, ບໍ່ສາມາດເພີ່ມ ຫຼື ແກ້ໄຂໂຕໃດໂຕໜຶ່ງໄດ້.
- | ເຖິງຢ່າງໃດກໍຕາມ, update ສາມາດເພີ່ມ ແລະ ແກ້ໄຂໄດ້ທັງ key-value.
- | ໃນ key ‘a’ ຖ້າແກ້ໄຂ value ໃຫ້ເປັນ 90 ໂດຍໃຊ້ setdefault, ຈະບໍ່ສາມາດຮັດໄດ້ ດັ່ງສະແດງລຸ່ມນີ້

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 print(dic)
3 dic.setdefault('a', 90)
4 print(dic)
```

```
{'a': 10, 'b': 20, 'c': 30, 'd': 40}
{'a': 10, 'b': 20, 'c': 30, 'd': 40}
```

2. ການລຶບ Key-Value

2.1. Pop method

| pop method ຈະລຶບສະເພາະ key-value ທີ່ກຳນົດ ແລະ ຈະສິ່ງມັນກັບ.

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 print(dic.pop('a'))
3 print(dic)
```

```
10
{'b': 20, 'c': 30, 'd': 40}
```

- ▶ Pop(key) ຈະສິ່ງຄໍາທີ່ໄດ້ລຶບ ຫຼັງຈາກລຶບ key-value ທີ່ກຳນົດ.
- ▶ ໃນນີ້ ລຶບ key ‘a’ ຈາກ dictionary ທີ່ຊື່ dic ແລະ return 10.

2. ການລືບ Key-Value

2.2. Pop application

- | ຖ້າກໍານົດຄ່າ default ດັ່ງນີ້ pop(key, default value), ມັນຈະລືບ key-value ແລະ ສິ່ງຄ່າທີ່ລືບກຳຕ່ຳເມື່ອມີ key ໃນ dictionary, ແຕ່ຖ້າບໍ່ມີ key ມັນຈະສິ່ງພຽງແຕ່ຄ່າ default ເຫັນນັ້ນ.
- | ເນື່ອງຈາກບໍ່ມີ key 'f' ໃນ dictionary ທີ່ຊື້ dic, 8 ເປັນຄ່າ default ຈະຖືກສິ່ງກັບ.

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 print(dic.pop('a'))
3 print(dic)
```

```
10
{'b': 20, 'c': 30, 'd': 40}
```

```
1 print(dic.pop('f', 8))
```

```
8
```

Line 1

- ເນື່ອງຈາກບໍ່ມີ key 'f' ໃນ dic ຈຶ່ງສິ່ງຄ່າ 8 ມາ.

2. ການລຶບ Key-Value

2.3. Popitem method

| Popitem ລຶບ key-value ແລະ ສິ່ງກັບທັງ key ແລະ value (pair).

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 print(dic.popitem())
```

```
('d', 40)
```

```
1 print(dic)
```

```
{'a': 10, 'b': 20, 'c': 30}
```

- ▶ Popitem ລຶບ key-value ຈາກ dictionary ແລະ ສື່ງ key-value ທີ່ຖືກລຶບເປັນ tuple.
- ▶ ລັກສະນະການເຮັດວຽກຂອງ method ມີ ຂຶ້ນກັບ version ຂອງ Python, ຖ້າເປັນ Python 3.6 ຂຶ້ນໄປ ຈະລຶບ key-value ພ້າຍສຸດ.
- ▶ ເນື້ອໃນ course ມີແມ່ນໃຊ້ Python version 3.6 ຫຼື ສູງກວ່ານີ້ ເຊິ່ງຈະສື່ງອົງປະກອບທີ່ຖືກລຶບມາ.

3. ການລຶບທັງໝົດ Key-Value : clear

3.1. clear method

| ການລຶບທັງໝົດ key-value ໃນ dictionary.

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 print(dic)
3 dic.clear()
4 print(dic)
```

```
{'a': 10, 'b': 20, 'c': 30, 'd': 40}
{}
```

▶ ຫຼັງຈາກລຶບ dictionary ຂີ່ຂື້ dic ຈະຫວ່າງເປົາ ເຊັ່ນ: dictionary {}.

4. เอื้องสะພາຄ່າຂອງ key ແລະ save ມັນ : get

4.1. get method

| Get(key) ແມ່ນເອີ້ນຄ່າສະພາຂອງ key ຈາກ dictionary.

Ex ເອີ້ນສະພາຄ່າຂອງ key 'a' ຈາກ dictionary ທີ່ຊື່ dic

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}
2 a = dic.get('a')
3 print(a)
```

10

- ▶ ຖ້າກໍານົດຄ່າເລີ່ມຕົ້ນ ເຊັ່ນ get(key default value), ມັນຈະສົ່ງຄ່າ value ຂອງ key ອອກມາ ຖ້າມີໃນ dictionary, ແຕ່ຖ້າບໍ່ມີ ມັນຈະສົ່ງຄ່າ default ອອກມາ.
- ▶ ເນື່ອງຈາກບໍ່ມີ key 'z' ໃນ dictionary ທີ່ຊື່ dic, ມັນຈະສົ່ງຄ່າ default ອອກມາ ຄື 0 ເຊິ່ງເປັນ default value.

```
1 b = dic.get('z',0)
2 print(b)
```

0

5. Items, Keys, Values

5.1. Print key

| สิ่งที่สำคัญของงานดำเนินงานใน dictionary คือการหาค่า value ด้วย key ตั้งลุ่มนี้.

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}  
2 dic['a']
```

10

| ลุ่มนี้ แม่นางานสะແດງ key หັງໝົດໃນ dictionary โดยໃຊ້ method keys .

```
1 dic = {'a': 10, 'b': 20, 'c': 30, 'd': 40}  
2 dic.keys()  
  
dict_keys(['a', 'b', 'c', 'd'])
```

5. Items, Keys, Values

5.2. Print value

- | ຄ່າທັງໝົດຈະຖືກສື່ງອອກມາໂດຍໃຊ້ພັງຊັນ values.
 - ▶ ລຸ່ມນີ້ແມ່ນການສື່ງຄ່າທັງໝົດທີ່ມີໃນ dictionary.

```
1 dic.values()
```

```
dict_values([10, 20, 30, 40])
```

5. Items, Keys, Values

5.3. พิมพ์ key และ value

| พิมพ์ key และ value.

- ▶ เป็นการสืบค่าทั้งหมดใน dictionary.
- ▶ ฟังก์ชัน dic.items() ถูกอ่านใน for loop, จะส่ง (key,value) เป็น tuple ของมา, ดังจะแสดงลุ่มมี.

```
1 dic.items()
```

```
dict_items([('a', 10), ('b', 20), ('c', 30), ('d', 40)])
```

```
1 for str1, num in dic.items():
2     print(str1, ':', num)
```

```
a : 10
b : 20
c : 30
d : 40
```

 Line 1

- Values corresponding to keys and values of the dictionary are assigned to str1 and num, respectively.

5. Items, Keys, Values

5.4. method ຕ່າງໆຂອງ dictionary ແລະ functions

Method	function
keys()	Returns ຫັງໝົດ keys ໃນ dictionary.
values()	Returns ຫັງໝົດ values ໃນ dictionary.
items()	Returns ຫັງໝົດ items ໃນ dictionary in [key]:[value] pairs.
get(key)	Returns ຄ່າ ຂອງ key. ຖ້າບໍ່ມີ key, return None.
pop(key)	Returns ຄ່າຂອງ key ແລະ ລືບ item. ຖ້າບໍ່ມີ key, ມັນເຮັດໃຫ້ເກີດຂໍ້ຍົກເວັ້ນ KeyError.
popitem()	ສິ່ງຄືນ item ທີ່ເລືອກແບບສຸມ ແລະ ລືບ item ດັ່ງກ່າວ.
clear()	ລືບຫັງໝົດ ໃນ dictionary.

Paper coding

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

Q1. ໃຫ້ສ້າງ dictionary ທີ່ຂຶ້ນ person_dic ໂດຍມີຂໍ້ມູນ ດັ່ງສະແດງລຸ່ມນີ້.
ຈາກນັ້ນໃຫ້ພິມຂໍ້ມູນດັ່ງກ່າວອອກມາ ໂດຍໃຊ້ for loop ເພື່ອໃຫ້ໄດ້ຜົນຮັບດັ່ງກ່າວ

Conditions for Execution

```
'Last Name': 'Doe', 'First Name': 'David', 'Company': 'Samsung'
```

Time

5min

Last Name

First Name

Company

Last Name : Doe
First Name : David
Company : Samsung



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

Q2.

ชูงนโปรแกรมเพื่อจัดกานสินค้าในร้าน ที่ร้านสะดวกซื้อ, สินค้าในร้านจะถูกจัดเก็บไว้ใน dictionary ดังแสดงลุ่มนี้ . จึงชูน
โปรแกรมเพื่อรับຊีของสินค้าจากผู้ใช้ เพื่อให้สั่งจำนำวนสินค้ามีน้อยลงมา.

**Example program
execution results.**

Enter name of the item: Milk
1

Time

5min

Items

```
1 items = {"Coffee": 7, "Pen":3, "Paper cup": 2, "Milk": 1, "Coke": 4, "Book":5}
```



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

| Let's code

1. หัวข้อความเข้าใจ code ในงานประมวลผลของ Text Data

1.1. Text data

- | Text คือคำที่เป็นเวลาเดินทาง ซึ่งมันเป็นสิ่งจำเป็น และ สำคัญยิ่งสำหรับมนุษย์ที่ใช้ในสื่อสาร.
- | Text data อาจจะแบ่งออก成ที่มีโครงสร้าง (HTML, XML, CSV, JSON files) และ เอกสารที่บ่อมีโครงสร้าง (text in natural language).
- | โดยทั่วไป, เมื่อจากข้อมูลต้นฉบับเป็นข้อมูลที่บ่อบรูปแบบที่สามารถนำไปประมวลผลได้ จึงจำเป็นที่จะต้องแก้ไขให้สามารถนำไปประมวลผลได้.



1. ทำความเข้าใจ code ในงานประเมินผู้เรียน Text Data

1.2. Text data ແລະ word cloud expression



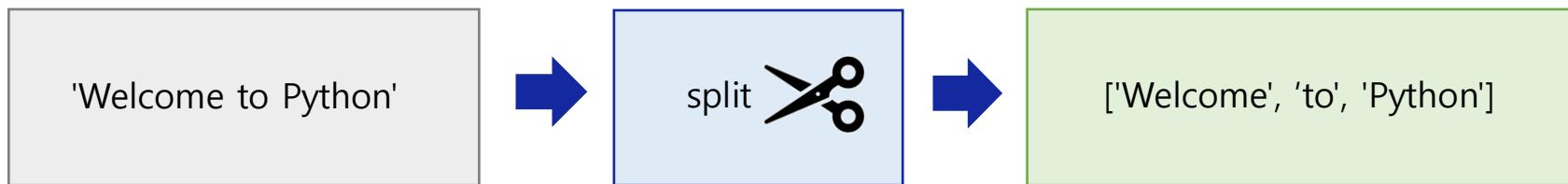
- | ຮູບດ້ານເທິງເປັນຕົວຢ່າງເຕັກໂນໄລຊີການສະແດງຮູບພາບທີ່ເອັນວ່າ word cloud ສ້າງໂດຍ wordart.com.
 - | Text data ສາມາດປະມວນຜົນສະເພາະ string function ທີ່ໄດ້ຮັບຈາກ Python's primary library, ແຕ່ modules ພາຍນອກ library ນີ້ທີ່ດີທີ່ສຸດ ເຊັ່ນ: BeautifulSoup, csv, json ແລະ nltk ເຮັດໃຫ້ການປະມວນຜົນ ແລະ ການວິຄາະຂໍ້ມູນໄດ້ງ່າຍເຂັ້ນ.(ການນຳໃຊ້ module ພາຍນອກ ຈະໄດ້ສະເໜີໃຫ້ຮູ້ໃນພາຍຫຼັງ)
 - | ຖ້າຕ້ອງການເຮັດວຽກທີ່ຊັບຂອ້ອນນີ້ ຈໍາເປັນຕົວງ່າວີທີ່ຈັດການກັບ string.

1. หัวข้อความเข้าใจ code ในงานประมวลผลข้อมูล Text Data

1.3. Split method

I Split method เป็น method ตัด strings.

- ▶ สิ่งมุ่งค่าสับใน string ที่กแบ่งกันด้วยจุด และ ช่องหว่าง.
- ▶ ตัวอย่าง, ถ้าใช้ method split ใน string ดังลุ่มนี้, จะรีดให้成 string สามารถแบ่งออกเป็นสาม string.



```
1 s = 'Welcome to Python'  
2 s.split()  
['Welcome', 'to', 'Python']
```

1. ហំភាពមេដីៗ នៃ code នៃការបែងចែកពិនិត្យ Text Data

1.4. ពិវិះពាក្យការណ៍ដីៗ split

| ក្នុងបញ្ជីកម្មវិធី, តើអ្វីដោយពីរក្នុងការបែងចែកពិនិត្យការណ៍ដីៗ split មួយនៅក្នុងការណ៍ដីៗ តើជាប្រព័ន្ធដឹងដូចម្នៀៗ?

```
1 s = '2021.8.15'  
2 s.split('.')  
  
['2021', '8', '15']
```

| រួចរាល់ការណ៍ដីៗ 'Hello, World!' ជាអ្នកប្រើប្រាស់ក្នុងការណ៍ដីៗ តើជាប្រព័ន្ធដឹងដូចម្នៀៗ?

```
1 s = 'Hello, World!'  
2 s.split(',')  
  
['Hello', ' World!']
```

| សំណើនៅក្នុងការណ៍ដីៗ 'Hello' និង 'World!'. តើបានប្រើប្រាស់ក្នុងការណ៍ដីៗ តើជាប្រព័ន្ធដឹងដូចម្នៀៗ?

| ខ្លួនឯកសារណ៍ដីៗ តើបានប្រើប្រាស់ក្នុងការណ៍ដីៗ 'Hello' និង 'World!'. តើបានប្រើប្រាស់ក្នុងការណ៍ដីៗ តើជាប្រព័ន្ធដឹងដូចម្នៀៗ?

1. ទាំងវាមើនៅក្នុង code និងការបែងចាយជាបន្ទាន់ Text Data

1.5. Strip method

- I Strip ត្រូវបានដោះស្រាយពីការលើកខ្លួន និងខ្លួន string.
 - ▶ ពីរគឺជាការបែងចាយ ដែលបានដោះស្រាយពីការលើកខ្លួន និងខ្លួន string.
 - ▶ code លើមនឹម បែងចាយជាបន្ទាន់ Text Data និងខ្លួន string ដោយបានដោះស្រាយពីការលើកខ្លួន និងខ្លួន string.

```
1 s='Hello, world!'
2 s.split(' ',')
```

```
['Hello, world']
```

1. ທຳຄວາມເຂົ້າໃຈ code ໃນການປະມວນຜົນຂອງText Data

1.5. Strip method

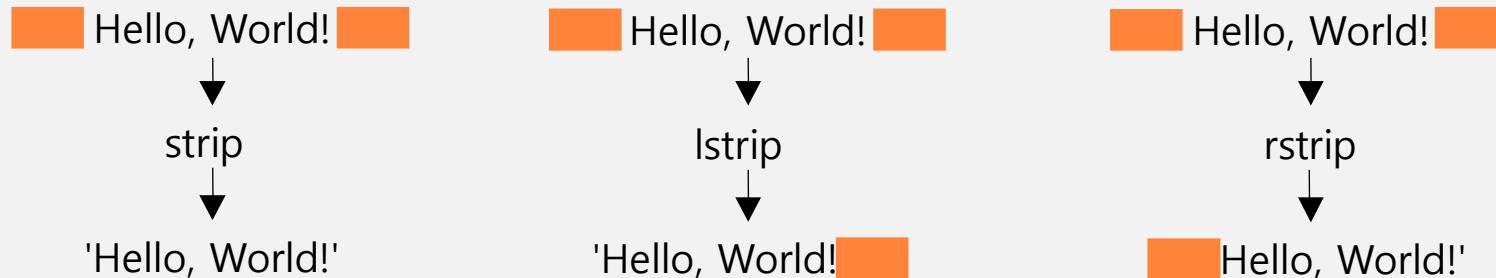
- | Strip ລຶບຊ່ອງຫວ່າງ ກ່ອນ ແລະ ຫຼັງ string.
- | ▷ ທ້າມີຊ່ອງຫວ່າງຫຼັງເຄື່ອງໝາຍຈຸດ ດັ່ງລຸ່ມນີ້, ມັນຈະລຶບຊ່ອງຫວ່າງດັ່ງກ່າວຫຼັງຈາກແບ່ງ string.
 - ▶ ລຸ່ມນີ້ເປັນການອະທິບາຍການລຶບຊ່ອງຫວ່າງ ກ່ອນ ແລະ ຫຼັງ string ໂດຍນຳໃຊ້ strip method.
 - ▶ Code ຄໍາສັ່ງລຸ່ມນີ້ ແມ່ນ code ແບບຫຍໍ້ ຂອງ list ແລະ ຈະນຳສະເໜີລາຍລະອຽດໃນUnit 19.
 - ▶ expression method ນີ້ ສະແດງໃນ Python coding.

```
1 s = 'Welcome, to, Python, and , bla, bla '
2 s.split(',') # separate s using comma as separator
['Welcome', ' to', ' Python', ' and ', ' bla', ' bla ']
```

```
1 s = 'Welcome, to, Python, and , bla, bla '
2 [x.strip() for x in s.split(',')] # delete space before and after separator character.
['Welcome', 'to', 'Python', 'and', 'bla', 'bla']
```

| ដើម្បីក្នុង strip จะលើបន្លំខ្លួចខាងក្រោម និង ខ្លួចខាងក្រោម string.

- ▶ เมែនឹងការប្រមិជ្ជនា, strip method ត្រូវបានបន្លំខ្លួចខាងក្រោម និង ខ្លួចខាងក្រោម string.
- ▶ lstrip ត្រូវបានបន្លំខ្លួចខាងក្រោមតាមខ្លួចខាងក្រោម string.
- ▶ rstrip ត្រូវបានបន្លំខ្លួចខាងក្រោមខ្លួចខាងក្រោម string.



```
1 s = "Hello, World! "
2 s.strip()
'Hello, World!'
```

```
1 s.lstrip()
'Hello, World! '
```

```
1 s.rstrip()
'Hello, World! '
```

1. หัวใจความเข้าใจ code ในงานปั๊มวนผินของ Text Data

1.6. Strip method และ การลีบสะเนะ character

- | ถ้าต้องการลีบสะเนะ character ท่อน และ ท้าย string ใช้คำสั่งนี้.
- | code ลุ่มนี้ແມ່ນการลีบ '#' character.

```
1 s = "#####this is an example#####"
2 s.strip('#')
```

'this is an example'

- | rstrip และ lstrip ยังสามารถใช้ในการลีบสะเนะ characters ได้.

```
1 s = "#####this is an example#####"
2 s.lstrip('#')
```

'this is an example#####'

```
1 s.rstrip('#')
```

'#####this is an example'

1. ทำความเข้าใจ code ในงานปั๊มวนผินຂອງ Text Data

1.7. งานดี้อ้ม string โดยใช้ Join method

| split เป็นฟังก์ชันแยก string เป็น string ย่อๆ, ส่วน join เป็นฟังก์ชันรวม string ย่อๆเป็น string.



```
1 ', '.join(['apple', 'grape', 'banana']) ← Connects three words using comma  
'apple,grape,banana'
```

```
1 '-'.join('010.1234.5678'.split('.')) # change phone number with . to -  
'010-1234-5678'
```

| From the example above, we can see that join puts adhesive characters only between strings and not before or after strings.

1. หัวข้อความเข้าใจ code ในงานประมวลผลข้อมูล Text Data

1.7. งานดึงข้อมูล string โดยใช้ Join method

- | ถ้าบ่งบอกว่ามี string เมื่อใช้ join, มันจะนำสิ่ง string ปั๊ดระหว่างทว่าๆ กัน
- ▶ ถ้าบ่งบอกว่ามี string มันจะนำสิ่ง string ดังลุ่มนี้.

```
1  '.join(['apple', 'grape', 'banana'])  
'applegrapebanana'
```

1. ทำความเข้าใจ code ในงานปั๊มวนผินของ Text Data

1.7. งานดึงดูม string โดยใช้ Join method

| Join บางครั้งบังใช้เพื่อรวม characters.

- ▶ join ยังสามารถใช้เพื่อรวม characters ที่ได้จากการแยก string โดยใช้ฟังก์ชัน list ดังจะแสดงลุ่มๆ, ดึงมันรวมเป็นข้อความ characters ให้เป็น string เดี๋ม.

Ex Let's connect clist, "hello world" that is changed from string to list, using join.

```
1 s = 'hello world'  
2 clist = list(s)  
3 clist  
['h', 'e', 'l', 'l', 'o', ' ', 'w', 'o', 'r', 'l', 'd']
```

```
1 ''.join(clist)  
'hello world'
```

1. หัวข้อความเข้าใจ code ในงานประมวลผลข้อมูล Text Data

1.8. งานใช้ split และ join ร่วมกัน

| ด้วยงานใช้ split และ join ร่วมกัน, สามารถลบ ช่องทว่างที่บีบจำเป็นออกจาก string.

```
1 a_string = 'Actions \n\t speak louder than words'  
2 a_string
```

```
'Actions \n\t speak louder than words'
```

```
1 print(a_string)
```

```
Actions  
    speak louder than words
```

```
1 word_list = a_string.split()  
2 word_list
```

```
['Actions', 'speak', 'louder', 'than', 'words']
```

```
1 refined_string = " ".join(word_list) # line change and tab disappears from a_string  
2 print(refined_string)
```

```
Actions speak louder than words
```

| After removing the space using split, the string is attached using join.

1. ทำความเข้าใจ code ในงานปั๊มวนผินของ Text Data

1.9. งานแบบตัวอักษรสอง

| ใน string, มีฟังก์ชันแบบตัวอักษรสองให้เป็นตัวน้อยมิດ ซึ่ง ให้เป็นตัวใหญ่มิດ ซึ่ง ขึ้นต้นด้วยตัวใหญ่, เนื่องจากฟังก์ชันถูกกำหนดไว้ใน code ลุบมี.

```
1 s = 'Hello, World!'
2 s.lower()
```

```
'hello, world!'
```

```
1 s.upper()
```

```
'HELLO, WORLD!'
```

| Capitalize ฟังก์ชัน จะเรัดให้ คำสับขึ้นต้นด้วยโตใหญ่.

```
1 s = 'hello'
2 s.capitalize()
```

```
'Hello'
```

1. หำຄວາມເຂົ້າໃຈ code ໃນການປະມວນຜົນຂອງText Data

1.10. Startswith method

- | Startswith ເປັນການກວດສອບຕົວອັກສອນຕົວທຳອິດຂອງ string.
- | ຈາກ code ດ້ວຍລຸ່ມນີ້, ຖ້າຕົວອັກສອນຕົວທຳອິດຂອງ string ແມ່ນ "A", ມັນຈະສົ່ງຄ່າ True ອອກມາ.

```
1 s = 'ABCDEFGHIJKLMNPQRSTUVWXYZ'  
2 s.startswith('A')
```

True

- | ລຸ່ມນີ້ເປັນການກວດສອບຕົວອັກສອນທີ່ມີການກຳນົດຕໍ່ແຫ່ງ.

```
1 s.startswith('B',1)
```

True

```
1 s.startswith('B',5)
```

False

- | ເນື່ອງຈາກ 'B' ແມ່ນຢູ່ຕໍ່ແຫ່ງທີ່ສອງຂອງ string, startswith('B',1) ຈະສົ່ງຄ່າ True, ຖ້າ startswith('B',5) ຈະສົ່ງຄ່າ False.

1. หัวข้อความเข้าใจ code ในงานปั๊มวนผินของ Text Data

1.11. endswith method

- | endswith เป็นฟังก์ชันที่มี功能ตรวจสอบว่าตัวอักษรที่กำหนดมีอยู่ในตัวอักษรท้ายของ string หรือไม่
- ▶ จาก code ลุ่มนี้, เมื่อจาก 'Z' เป็นตัวอักษรที่ลิ่งท้ายของ string, สะนั่น endswith('Z') จึงสิ่ง True ออกมานะ

```
1 s = 'ABCDEFGHIJKLMNPQRSTUVWXYZ'  
2 s.endswith('Z')
```

True

1. หำຄວາມເຂົ້າໃຈ code ໃນການປະມວນຜິນຂອງText Data

1.11. endswith method

- | endswith ເປັນພື້ນຂັ້ນທີ່ໃຊ້ກວດສອບຕົວອັກສອນທີ່ລົງທ້າຍຂອງ string.
- | ໃນ endswith method, ສາມາດກຳນົດຕຳແໜ່ງເລີ່ມຕົ້ນ ແລະ ສຸດທ້າຍຂອງ string ດັ່ງລຸ່ມນີ້.

► **endswith(Search string, start of string, end of string)**

```
1 s.endswith('C', 1, 3)
```

True

```
1 s.endswith('C', 1, 10)
```

False



Line 1

- $s = 'ABCDEFGHIJKLMNPQRSTUVWXYZ'$, ຖ້າໃຫ້ 1 ເປັນຕຳແໜ່ງເລີ່ມຕົ້ນ ແລະ 3 ເປັນຕຳແໜ່ງສຸດທ້າຍຂອງ s, ສະນັ້ນ s ຈະກາຍເປັນ 'ABC' .
- ແລະ $\text{endswith('C', 1, 3)}$ ແມ່ນ True.

1. หัวข้อความเข้าใจ code ในงานประมวลผลข้อมูล Text Data

1.12. Endswith method

| Replace ฟังก์ชัน สามารถบูรณาการตัวอักษรสองตัวเพื่อหาว่ามีตัวอักษรที่ต้องการอยู่ใน string.

- ▶ 功用คำสั่ง replace method จะดำเนินการดังนี้
 - ▶ replace(old, new, [count]) -> replace("Value to find", "Value to change", [number of change])
 - ▶ code คำสั่งลุ่มนี้ แบ่ง code ที่ลิบ และ บูรณาการจุด (,) ใน stringe.

```
1 text = '123,456,789,999'
2
3 replaceAll= text.replace(",","")
4 replace_t1 = text.replace(","," ",1)
5 replace_t2 = text.replace(","," ",2)
6 replace_t3 = text.replace(","," ",3)
7 print("Result:")
8 print(replaceAll)
9 print(replace_t1)
10 print(replace_t2)
11 print(replace_t3)
```

Result :

```
123456789999
123456,789,999
123456789,999
123456789999
```

| 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 ກ່າວຈະຮັດໃຫ້ບໍ່ສົມດຸນກັນ. ຄຸສອນຮູ້ດີວ່າ ການຈັບຄຸ້ກັນບໍ່ແມ່ນ ຍຸດທະສາດ “ແບ່ງເພື່ອເອົາຊະນະ” ແຕ່ເປັນຄວາມ ພະຍາຍາມຮັດວຽກຮ່ວມກັນຂອງນັກຮຽນໃຫ້ປະສົບຜົນສໍາເລັດ. ຄຄວນທີ່ກາເວັ້ນການຈັບຄຸ້ກັນລະຫວ່າງນັກຮຽນອ່ອນ ແລະ ນັກຮຽນເກົ່າງ.

| ກະຕຸ້ນນັກຮຽນໂດຍການໃຫ້ສິ່ງຈຸງໃຈພື້ນເສດ

ຂໍສະເໜີແຮງຈຸງໃຈທີ່ຮັດໃຫ້ນັກຮຽນຈັບຄຸ້, ໂດຍສະເພາະນັກຮຽນທີ່ມີຄວາມສາມາດສຸງ. ບາງຄຸສອນໄດ້ພື້ນວ່າ ການຈັບຄຸ້ຮັດວຽກມອບໝາຍ ແມ່ນມີ ປະໂຫຍດ ສໍາລັບໜຶ່ງ ຫຼື ສອງວຽກມອບເທົ່ານັ້ນ



Pair Programming Practice

| ចំណាំការបំពើនិងរុញខែងមករុញ

ສິ່ງທ້າທາຍສໍາລັບຄູແມ່ນເພື່ອຊອກຫາວິທີທີ່ຈະປະເມີນຜົນການຮຽນຂອງນັກຮຽນ, ຄູຮູ້ບໍ່ວ່າ ນັກຮຽນ ໄດ້ຕັ້ງໃຈຮຽນ ຫຼື ບໍ່ຕັ້ງໃຈຮຽນ. ຜູ້ສ່ວວຊານໄດ້ແນະນຳໃຫ້ທີບທວນການອອກແບບຫຼັກສຸດການຮຽນ ແລະ ຮູບແບບການປະເມີນ ພ້ອມທັງປີກສາຫາລືຢ່າງຈິງຈຳກັບນັກຮຽນ ກ່ຽວກັບພິດຕິກຳທີ່ຈະບໍ່ຕັ້ງໃຈຮຽນ ນອກຈາກນີ້ຢັ້ງໄດ້ແນະນຳມອບວຽກມອບໝາຍໃຫ້ນັກຮຽນ ພ້ອມທັງອະທິບາຍໃຫ້ເຂົ້າເຈົ້າຢ່າງຈະເຈັ້ງ

| ສະພາບແວດລ້ອມຂອງການຮຽນຮັກ

ສະພາບແວດລ້ອມການຮຽນຮູ້ຮ່ວມກັນເກີດຂຶ້ນໄດ້ທຸກເວລາທີ່ຜູ້ສອນຮຽກຮ້ອງໃຫ້ນັກຮຽນເຮັດວຽກຮ່ວມກັນໃນກິດຈະກຳການຮຽນຮູ້ ເຊິ່ງອາດຈະເປັນກິດຈະກຳທີ່ເປັນທາງການ ແລະ ບໍ່ເປັນທາງການ ແລະ ອາດຈະບໍ່ລວມເຖິງການປະເມີນຜົນການຮຽນໂດຍກິງ. ເຊັ່ນຕົວຢ່າງ ໃຫ້ນັກຮຽນຈັບຄຸ້ງກັນເພື່ອເຮັດວຽກອບໝາຍ ໂດຍນັກຮຽນຈະຕ້ອງທີບທວນກ່ຽວກັບການສອນຂອງອາຈານທີ່ຜ່ານມາ ແລະ ລະດົມແນວຄິດພາຍໃນກຸ່ມ ພ້ອມທັງມືການແບ່ງວຽກໃຫ້ແຕ່ລະຄົນຮັບຜິດຊອບ ຈາກນັ້ນກໍໃຫ້ມີການແລກປ່ຽນຄວາມຄິດເຫັນເຊິ່ງກັນ ແລະ ກັນ ເພື່ອເຮັດວຽກອບໝາຍໃຫ້ສໍາເລັດຕາມເປົ້າໝາຍທີ່ວ່າງໄວ້.

Q1.

ມາ ແກ້ໄຂໂປຣແກຣມ ເພື່ອຈັດການສິນຄ້າໃນຄັງ ຂອງຮ້ານສະດວກຊື້ ທີ່ເຮົາໄດ້ຮັດໃນ paper coding ໂດຍການເພີ່ມ code ເພື່ອເພີ່ມ ຫຼື ຫຼຸດ ສິນຄ້າໃນຄັງ, ນອກຈາກນີ້ຍັງຈັດທໍາມານຸ່ງໆ ແລ້ວ ຂໍ້ຕົກລົງ: ການສອບຖາມສິນຄ້າໃນຄັງ, ຄັງສິນຄ້າ ແລະ ການຈັດສິ່ງ.

```
1 items = {"Coffee": 7, "Pen":3, "Paper cup": 2, "Milk": 1, "Coke": 4, "Book":5}
```

**Output example**

```
Select menu 1)check stock 2)warehousing 3) release 4) exit :1  
[check stock] Enter item: milk  
Stock: 1  
Select menu 1)check stock 2)warehousing 3) release 4) exit :3  
[Release] Enter item and quantity: coke 1  
Select menu 1)check stock 2)warehousing 3) release 4) exit :1  
[check stock] Enter item: coke  
Stock: 3  
Select menu 1)check stock 2)warehousing 3) release 4) exit :4  
Program exited.
```