

Приклади задач з використанням словників

Задача. Побудувати словник даних прогнозу погоди. Кожен рядок вхідних даних друкують на клавіатурі у форматі:

Місто, денна_t°, нічна_t°, хмарність, вітер, тиск

Наприклад:

Львів, +14, +8, невелика хмарність, північ-захід, 745

(На практиці такі дані отримують з файлів). Елементи даних розділені комами.

Надрукувати отриманий словник у формі таблиці.

Розв'язання. Спершу потрібно вирішити, що вважати ключем. Аналіз даних показує, що доцільно прийняти за ключ назву міста. По-друге, вирішуємо, як зображати значення до кожного ключа: як список окремих елементів, чи єдиний текстовий рядок. Єдиний текстовий рядок спрощує зображення, але ускладнює аналіз окремих показників. Список елементів ускладнює зображення, але полегшує аналіз окремих показників.

Подібні міркування є характерними для багатьох задач побудови словників. Зважаючи на перспективу можливих застосувань, оберемо зображення даних як список окремих елементів. Всі елементи будуть окремими текстовими рядками:

'Львів': ['+14', '+8', 'невелика хмарність', 'північ-захід', '745']

Задачу будемо розв'язувати за таким загальним алгоритмом. 1) Прочитаємо весь список рядків і збережемо в деякій величині програми, щоб за потреби мати можливість повторно переглядати. Розмір списку наперед не відомий, тому будемо вважати, що список закінчується порожнім рядком. 2) Будуємо словник шляхом поділу тексту кожного рядка. Перший елемент рядка буде ключем, решту елементів рядка організуємо як значення у формі внутрішнього списку словника. 3) Виконуємо операції з словником за умовою задачі і будуємо зображення текстових рядків для наступного друкування. Тобто, маємо в пам'яті прообрази друкованих результатів у формі окремого списку рядків виду:

| | | | | | | | | | | |
|-------|--|-----|--|----|--|--------------------|--|--------------|--|-----|
| Львів | | +14 | | +8 | | невелика хмарність | | північ-захід | | 745 |
|-------|--|-----|--|----|--|--------------------|--|--------------|--|-----|

4) Друкуємо на екрані отримані результати.

Зауважимо, що така методика кроків розв'язання є характерною для реальних задач.

```
# крок 1: прочитати список рядків
listdata = [] # порожній список
while True :
    nextline = input()
    if nextline != '' : listdata.append(nextline)
    else : break
```

```
[ 'Київ,+12,+6,сонячно,захід,739',
  'Львів,+14,+8,невелика хмарність,північ-захід,745',
  'Одеса,+17,+12,сонячно,південь-захід,726',
  'Запоріжжя,+11,+3,місцями дощ,захід,731',
  'Харків,+9,+1,дощ,північ-захід,723',
  'Житомир,+10,+2,хмарно,північ,722' ]
```

```
# крок 2: побудувати словник
weather = { } # початково словник пустий
for oneline in listdata :
    values = oneline.split(',') # поділити текст кожного рядка
    weather[values[0]] = values[1:] # додати елемент до словника
```

```
{ 'Київ': ['+12', '+6', 'сонячно', 'захід', '739'],
  'Запоріжжя': ['+11', '+3', 'місцями дощ', 'захід', '731'],
  'Львів': ['+14', '+8', 'невелика хмарність', 'північ-захід', '745'],
  'Житомир': ['+10', '+2', 'хмарно', 'північ', '722'],
  'Харків': ['+9', '+1', 'дощ', 'північ-захід', '723'],
  'Одеса': ['+17', '+12', 'сонячно', 'південь-захід', '726'] }
```

побудувати можна зразу цілий генератором словника

```
#weather= { values.split(',')[0] : values.split(',')[1:] for values in listdata}
```

крок 3: операції з словником, зображення результатів

```
itmls = list(weather.keys()) # складемо список ключів
print(itmls)
['Київ', 'Запоріжжя', 'Львів', 'Житомир', 'Харків', 'Одеса']
listout = [ ] # список зображень текстових рядків результатів
for k in range(len(itmls)) :
    res = '{0:<18}|{1:^6}|{2:^6}|{3:<25}|{4:<20}|{5:^8}'.format(
        itmls[k], weather[itmls[k]][0], weather[itmls[k]][1],
        weather[itmls[k]][2], weather[itmls[k]][3],
        weather[itmls[k]][4])
    listout.append(res)
# крок 4: надрукувати результати
for i in range(len(listout)) : print(listout[i])
pass
```

Приклад отриманих результатів:

| | | | | | |
|-----------|-----|-----|--------------------|---------------|-----|
| Київ | +12 | +6 | сонячно | захід | 739 |
| Запоріжжя | +11 | +3 | місцями дощ | захід | 731 |
| Львів | +14 | +8 | невелика хмарність | північ-захід | 745 |
| Житомир | +10 | +2 | хмарно | північ | 722 |
| Харків | +9 | +1 | дощ | північ-захід | 723 |
| Одеса | +17 | +12 | сонячно | південь-захід | 726 |

Задача. За умовою попередньої задачі сортувати показники денної температури і надрукувати назви міст. Окремо друкувати список міст з нічною температурою менше +4°.

Для реалізації цієї задачі потрібно замінити у розв'язку попередньої задачі фактично лише третій крок.

```
# сортувати показники денної температури
# відібрати в список назви міст і денну температуру
daytm = [ (city, weather[city][0]) for city in itmls ]
[('Запоріжжя', '+11'), ('Львів', '+14'), ('Київ', '+12'), ('Одеса', '+17'),
 ('Житомир', '+10'), ('Харків', '+9')]
lml = lambda x : int(x[1]) # ключ сортування - температура
daytm.sort(key=lml, reverse=True)
[('Одеса', '+17'), ('Львів', '+14'), ('Київ', '+12'), ('Запоріжжя', '+11'),
 ('Житомир', '+10'), ('Харків', '+9')]
for s in daytm :
    print( '{0:<18}|{1:^6}'.format(s[0],s[1]) )
```

| | |
|-----------|-----|
| Одеса | +17 |
| Львів | +14 |
| Київ | +12 |
| Запоріжжя | +11 |
| Житомир | +10 |
| Харків | +9 |

```
# список міст з нічною температурою менше +4°
# звичайний перегляд цілого словника - якщо не треба сортувати
print('зю-зю-зю:')
for s in itmls : # перегляд за списком ключів
    if int(weather[s][1]) < 4 : print(s,weather[s][1])
print('-'*30, 'це вже все !')
pass
```

```
зю-зю-зю:
Харків +1
Запоріжжя +3
Житомир +2
----- це вже все !
```

Задача. Інформація про трамвайні маршрути Львова подана як сукупність пар текстових рядків формату:

номер_маршруту

перелік_зупинок_через_дефіс

Наприклад:

9
вокзал (через Вітовського)-площа Кропивницького-Бандери-Вітовського-Франка-площа Ринок-центр-головна пошта-політехніка-вокзал

Сукупність пар закінчується порожнім рядком.

1) Надрукувати початкові і кінцеві пункти відправлень.

2) Надрукувати список маршрутів, за якими можна доїхати до зупинки «політехніка».

Задачу будемо розв'язувати за таким загальним алгоритмом. 1) Прочитаємо весь список рядків і збережемо в деякій величині програми, щоб за потреби мати можливість повторно переглядати. 2) Будуємо словник, кожен елемент якого визначає пару сусідніх рядків вхідного тексту. Перший рядок пари – це є ключ, а другий – значення. Ключі перетворимо до числового формату. Значення збережемо як єдиний рядок. 3) Шляхом перегляду словника записуємо необхідні оператори для розв'язання задач і друкуємо результати.

Зауваження щодо елементів розв'язування. Початкові і кінцеві пункти відправлень – це перший і останній елемент списку зупинок кожного маршруту. Перевірка зупинки за маршрутом – це перевірка входження елемента в список зупинок.

```
# трамвайні маршрути
# крок 1: прочитати вхідні рядки даних і побудувати список
textindata = [] # порожній список
while True :
    nextline1 = input() # номер маршруту
    if nextline1 == '' : break # вихід з циклу за порожнім рядком
    nextline2 = input() # список зупинок
    textindata.append( (nextline1,nextline2) ) # додаємо пару - кортеж
```

Список textindata буде виглядати так:

```
[ ('9', 'вокзал (через Вітовського)-площа Кропивницького-Бандери-Вітовського-Франка-площа Ринок-центр-головна пошта-політехніка-вокзал'),
  ('1', 'вокзал (через центр)-політехніка-головна пошта-центр-площа Ринок-Франка-Вітовського-Бандери-площа Кропивницького-вокзал'),
  ('3', 'гідропарк-Володимира Великого-Княгині Ольги-Сахарова-Вітовського-Франка-центр'),
  ('5', 'гідропарк-Володимира Великого-Княгині Ольги-Сахарова-Вітовського-Франка-Підвальна-Замарстинівська-Липинського'),
  ('2', 'Коновальця-Чупринки-Бандери-головна пошта-центр-площа Ринок-Личаківська-Мечникова-Пасічна'),
  ('7', 'Погулянка-Мечникова-Личаківська-Підвальна-центр-Городоцька-Шевченка-Татарбунарська'),
  ('6', 'вокзал-площа Кропивницького-Городоцька-Замарстинівська-Хмельницького-Промислова-Липинського'),
  ('10', 'вокзал-площа Кропивницького-політехніка-головна пошта-центр-площа Ринок-Личаківська-Мечникова-Пасічна'),
  ('11', 'вокзал-площа Кропивницького-Бандери-Вітовського-Франка-Стуса') ]
```

```
# крок 2: побудувати словник
tramway = { } # початково словник пустий
for (route,station) in textindata : # зразу для пари - кортежа
    tramway[int(route)] = station # додати елемент до словника
```

Елементи словника будуть виглядати так:

```
{ 1: 'вокзал (через центр)-політехніка-головна пошта-центр-площа Ринок-Франка-Вітовського-Бандери-площа Кропивницького-вокзал',
  2: 'Коновальця-Чупринки-Бандери-головна пошта-центр-площа Ринок-Личаківська-Мечникова-Пасічна',
  3: 'гідропарк-Володимира Великого-Княгині Ольги-Сахарова-Вітовського-Франка-центр',
```

```
5: 'гідропарк-Володимира Великого-Княгині Ольги-Сахарова-Вітовського-Франка-  
Підвальна-Замарстинівська-Липинського',  
6: 'вокзал-площа Кропивницького-Городоцька-Замарстинівська-Хмельницького-  
Промислова-Липинського',  
7: 'Погулянка-Мечникова-Личаківська-Підвальна-центр-Городоцька-Шевченка-  
Татарбунарська',  
9: 'вокзал (через Вітовського)-площа Кропивницького-Бандери-Вітовського-Франка-  
площа Ринок-центр-головна пошта-політехніка-вокзал',  
10: 'вокзал-площа Кропивницького-політехніка-головна пошта-центр-площа Ринок-  
Личаківська-Мечникова-Пасічна',  
11: 'вокзал-площа Кропивницького-Бандери-Вітовського-Франка-Стуса' }
```

```
# крок 3: операції з словником, друкування результатів  
numrt = list(tramway.keys()) # складемо список ключів
```

```
[1, 2, 3, 5, 6, 7, 9, 10, 11]
```

```
# початкові і кінцеві пункти відправлень  
for rt in numrt :  
    liststation = tramway[rt].split('-') # список зупинок  
    print(rt, ':', liststation[0], '-', liststation[len(liststation)-1] )
```

```
1 : вокзал (через центр) - вокзал  
2 : Коновальця - Пасічна  
3 : гідропарк - центр  
5 : гідропарк - Липинського  
6 : вокзал - Липинського  
7 : Погулянка - Татарбунарська  
9 : вокзал (через Вітовського) - вокзал  
10 : вокзал - Пасічна  
11 : вокзал - Стуса
```

```
# маршрути до політехніки  
for rt in numrt :  
    liststation = tramway[rt].split('-') # список зупинок  
    if 'політехніка' in liststation :  
        print(rt, ':', tramway[rt])
```

```
1 : вокзал (через центр)-політехніка-головна пошта-центр-площа Ринок-Франка-  
Вітовського-Бандери-площа Кропивницького-вокзал  
9 : вокзал (через Вітовського)-площа Кропивницького-Бандери-Вітовського-Франка-  
площа Ринок-центр-головна пошта-політехніка-вокзал  
10 : вокзал-площа Кропивницького-політехніка-головна пошта-центр-площа Ринок-  
Личаківська-Мечникова-Пасічна
```