

## Friday, 4 September 2020

### 1 Tuple and List (30 mins)

#### 1.1 Objectives

1. Know the concept of tuples
2. Be able to store tuples in a list
3. Can use the function **sorted**

#### 1.2 Description

Read the score.csv file which is attached in the lab zip file. Print all students one line for one student sorted by id ascending. And print all students sorted by sum of score descending. The file format is same as in the previous lab as shown below.

```
5995778521,4,0,4,6,7
5974831121,6,5,0,6,4
5992211621,9,2,0,0,4
5987519321,3,5,1,4,3
5979431521,5,7,1,8,2
5976635721,5,6,5,5,1
5911249921,0,6,1,3,7
5906374521,9,0,7,3,8
5946204221,8,4,4,1,6
5963287221,4,0,5,5,9
```

#### 1.3 Procedure

Read all lines of student scores. Store each line in one tuple. Store all tuples in a list with summation of all quizzes. Then sort the list using sorted function.

*Hint:* You can use the `append` function to add a new item to a list. And you can use the `sum` function to find the summation of all elements in a list.

*Option:* You can use `lst[i:j]` to refer a sublist starting from index  $i$  to  $j - 1$ . Moreover, if you use `lst[i:]`, python will refer a sublist starting from index  $i$  to the end of `lst`. But if you use `lst[:j]`, python will refer to a sublist starting from index 0 to  $j - 1$ .

For example, the source code

```
1 f=open('score.csv')
2 for line in f.readlines():
3     tokens = line.split(',')
4     tokens = [int(e) for e in tokens] # convert all
5     print(tokens, sum(tokens[1:]))
6 f.close()
```

will print the output below.

```
[5963032021, 10, 2, 8, 9, 10] 39
[5920552721, 1, 1, 2, 0, 9] 13
[5995635821, 8, 8, 6, 10, 7] 39
[5941861521, 7, 1, 8, 4, 8] 28
[5950684221, 1, 4, 7, 2, 9] 23
...
```

If you use the code below, you can create a list which stores all tuples.

```
1 f=open('score.csv')
2 by_id = []
3 for line in f.readlines():
4     tokens = line.split(',')
5     tokens = [int(e) for e in tokens]
6     by_id.append(tuple(tokens))
7 f.close()
```

*Hint:* We can convert a list to tuple using `tuple(lst)` and convert a tuple to list using `list(tpl)`. So that the code

```
1 list_data=[1,2,3,5]
2 tuple_data=('a','b',1)
3
4 print(tuple(list_data))
5 print(list(tuple_data))
```

will print the output below.

```
(1, 2, 3, 5)
['a', 'b', 1]
```

*Hint:* You can use `sorted(list)` to sort a list of tuples as well. It will sort all data by comparing the first element of tuple first. If the first elements are equal, it will sort by the next element and so on.

## 1.4 Sample Output Screen

Sort ascending by id.

```
5901150821 1 6 4 3 9 23
5901383821 9 7 1 2 0 19
5901486221 10 4 4 8 6 32
5901611321 7 1 7 7 8 30
5901701921 3 10 5 10 9 37
...
```

Sort descending by sum of score.

```
5976531021 9 9 10 7 9 44
5971168221 10 5 10 9 10 44
5984437321 7 10 8 8 10 43
5992466421 7 10 6 9 10 42
5940000821 8 10 10 7 7 42
...
```

[illegible]

### 3 API and json (30 mins)

#### 3.1 Objectives

1. Be able to apply dictionary to real world programming
2. Be able to use api
3. Be able to use the library json and requests

#### 3.2 Description

From the openweathermap api, please find the time of the highest temperature and the highest temperature in the weather forecast return from the api.

Given the url that you can use to retrieve the data as below.

```
http://api.openweathermap.org/data/2.5/forecast?zip=10330,th&APPID=7743f38ce634083abe786e2d679955e3&units=metric
```

#### 3.3 Procedure

Same as in the lecture

#### 3.4 Sample Output Screen

```
2018-09-03 06:00:00 35.44
```