DS4UX: How to get data with Python (3)

[HCDE598] RESTful APIs and data crawling

Sungsoo (Ray) Hong & Johnathan Morgan

Python Function (a.k.a., Subroutine, Procedure)

Why use Function

- Function:
  "A function is a block of organized, reusable code that is used to perform a single, related action. Functions provide better modularity for your application and a high degree of code reusing. As you already know, Python gives you many built-in functions like print(), etc. but you can also create your own functions. These functions are called user-defined functions." from Tutorial Points
- Why use functions?
  a) Make your code less redundant b) helps you to organize yourself while you implement the code c) already benefit from using several procedures built by other programmers

Structure of a Python function

- Structure:

```
def ProcedureName (parameters):
    #Do something with parameters and find results
    something = results
    #Return something so that the main procedure can use that
    return something

#set parameters to hand over to the function ProcedureName
parameters = { }
Results = ProcedureName(parameters)
#Use Results
…
```

Structure of a Python function

- Structure:

```
def ProcedureName (parameters):
    #Do something with parameters and find results
    something = results
    #Return something so that the main procedure can use it
    return something
```

Procedure named *ProcedureName*

```
#set parameters to hand over to the function ProcedureName
parameters = { }
Results = ProcedureName(parameters)
#Use Results
```

***Main* procedure: The computer interpret your code from here**

Structure of a Python function

Defining function #2:
A name should be define

- Structure:

Defining function #3:
A function can receive one or multiple parameters

Defining function #1
"def " is a keyword

```
def ProcedureName (paramet
    …
    …
    …
  return somethin    #This is the end of the Procedure
```

Defining function #4:
A function can return a data

Use function #2:
Input parameter(s) in the parenthesis

```
parameters = { }
HeyProcedureGetTheRes lt = ProcedureName(parar
…
```

Use function #3:
Save the results in the main module

Use function #1:
Write down a function name in main module

Structure of a Python function

- Structure:

def *ProcedureName* (paramet

…

…

…

return *somethin* #This is the end of the Procedure

parameters = { }

HeyProcedureGetTheRes lt = ProcedureName(parar

…

Function

example

Function

Demonstration #1.

Write a function called *getDayToll*

which returns a daily toll from bgt_traffic dataset

A parameter: a string of a date (e.g., “07/03/2014”)

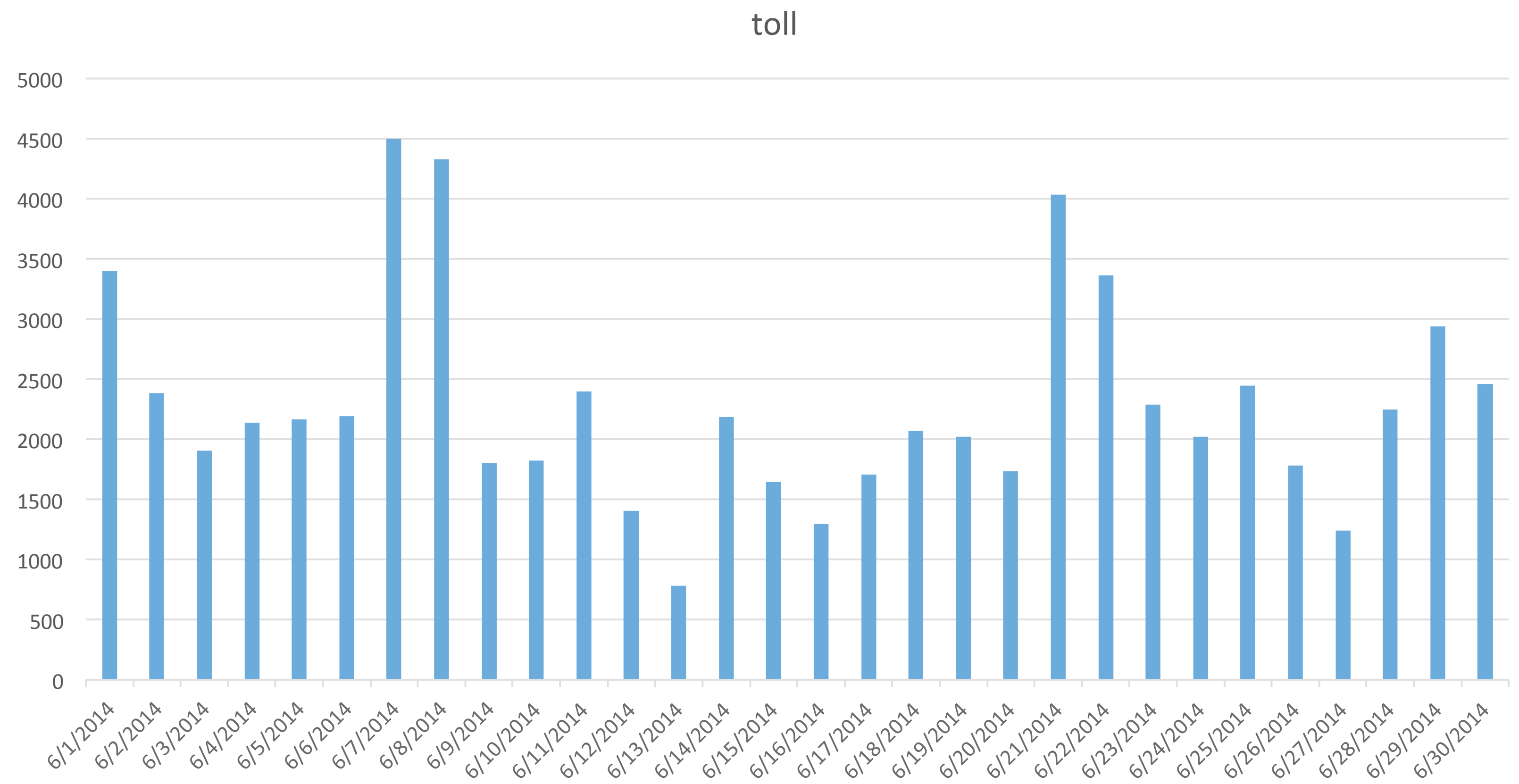
Return type: integer of a total toll for the date

Function

Demonstration #2.
Write a function called *writeMonthToll* that saves a csv file that each line has date and daily toll for a given month.
Parameter: a string of month and a string of year.
(e.g., "06", "2014“ make “06-2014.csv” and save a string like: “06/01/2014, xxxx \n 06/02/2014, xxxx\n … 06/30/2014, xxxx”

Function

Example: 06-2014.csv



Function

Code challenge activity #1.
Write a function called *getMonthToll*
which returns monthly toll from bgt_traffic dataset
Parameters: string of a month and a year
(e.g., month = “07”, year = “2014”)
Return type: integer of total toll for the month.
You may use “getDayToll()” to get monthly toll.

Function

Use challenge01_traffic_getMonthToll.py

Function

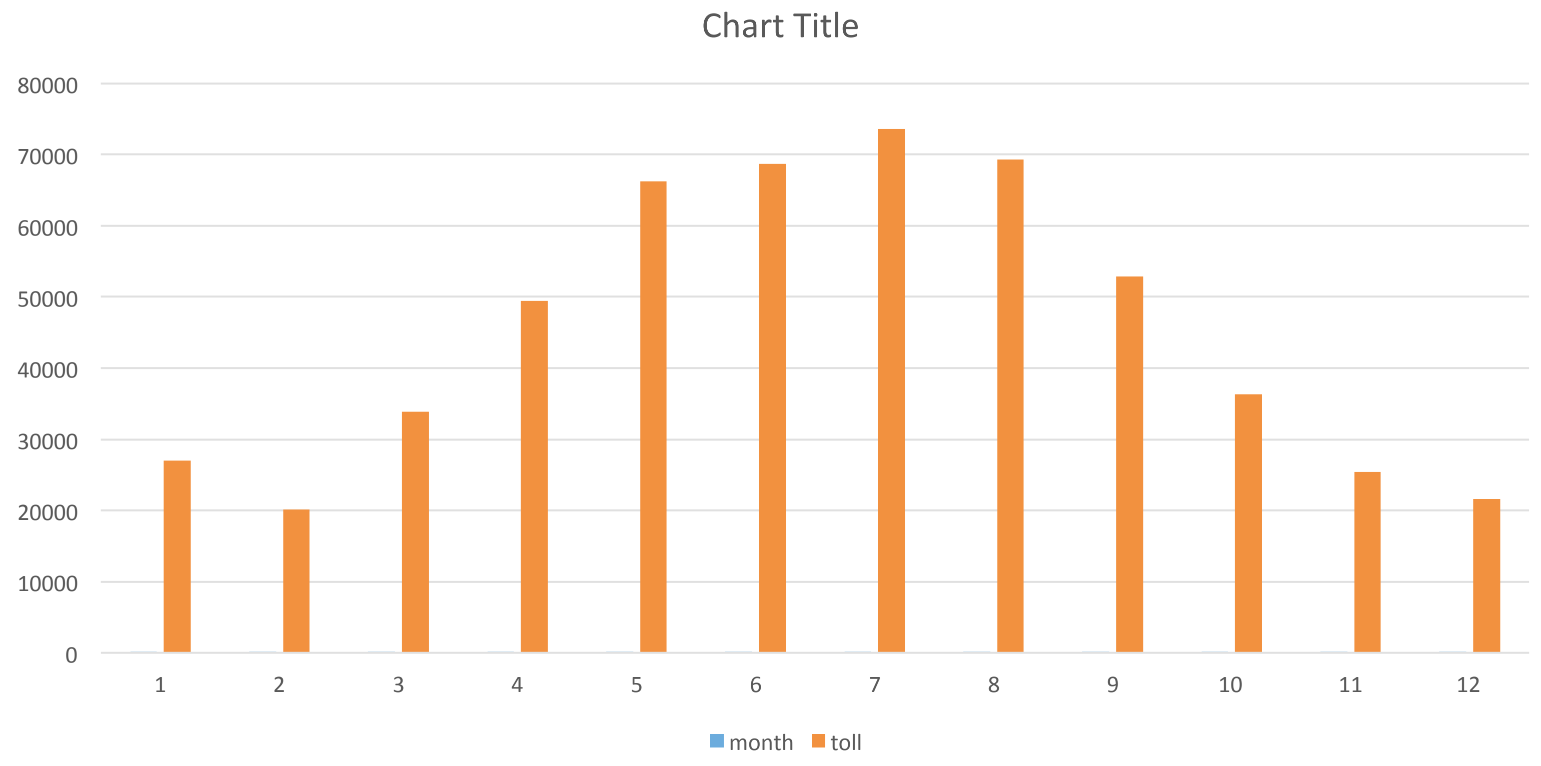
Code challenge activity #2.
Write a function called *writeYearToll*
which saves monthly toll of a given year.
Parameter: a string of year (e.g., “2014”)
(e.g., "2014“ makes “2014.csv” and save a string
like: “06/01, xxxx \n 02, xxxx\n … 12, xxxx”

Function

Use challenge02_traffic_writeYearToll.py

Function

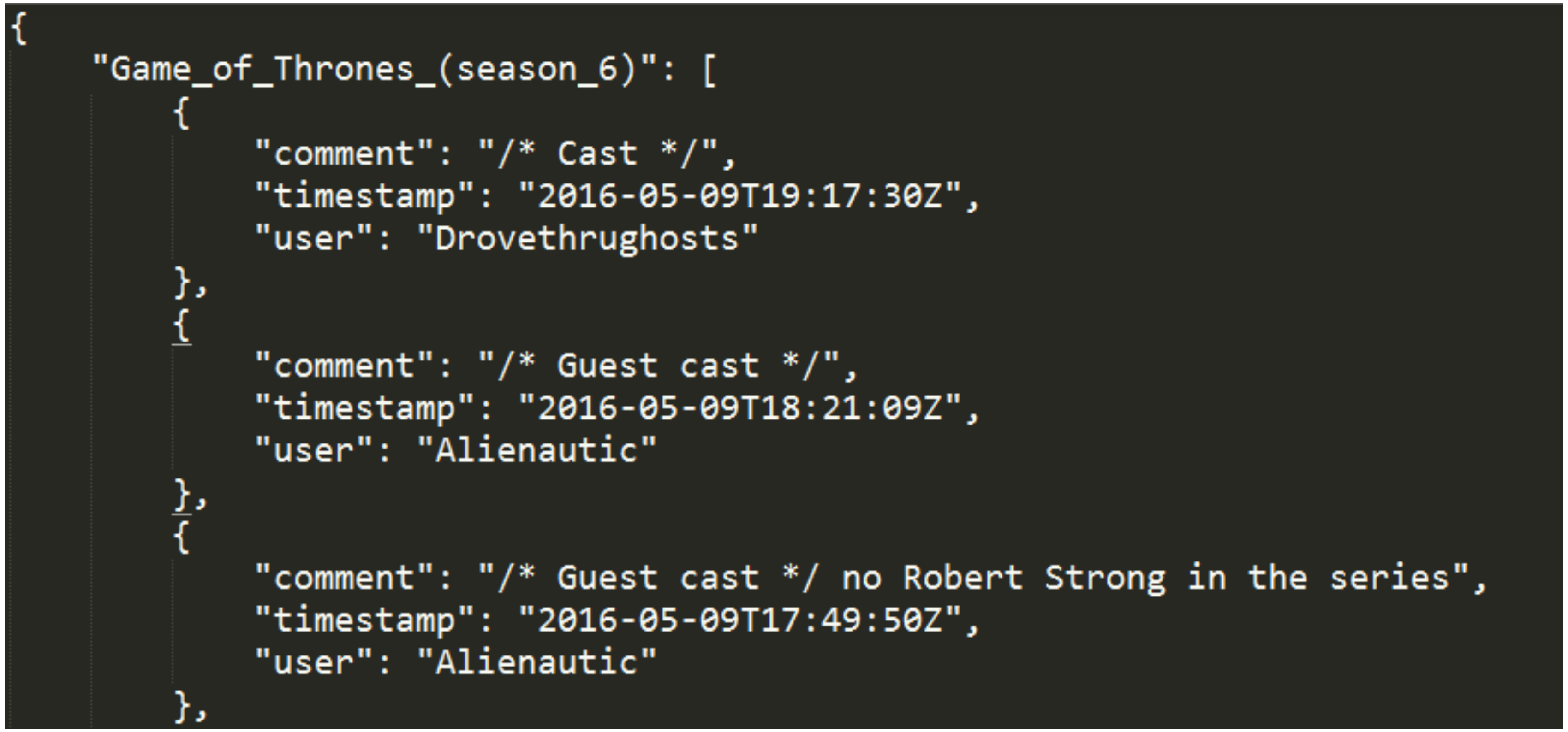
Example: 2014.csv



Function

Demonstration #3.
Write a function called *getRevisions(keyword)* which receives a keyword as an input.
A function get every revision record from Wikepedia and return the result as dictionary.

Function



```
{
    "Game_of_Thrones_(season_6)": [
        {
            "comment": "/* Cast */",
            "timestamp": "2016-05-09T19:17:30Z",
            "user": "Drovethrughosts"
        },
        {
            "comment": "/* Guest cast */",
            "timestamp": "2016-05-09T18:21:09Z",
            "user": "Alienautic"
        },
        {
            "comment": "/* Guest cast */ no Robert Strong in the series",
            "timestamp": "2016-05-09T17:49:50Z",
            "user": "Alienautic"
        },
```

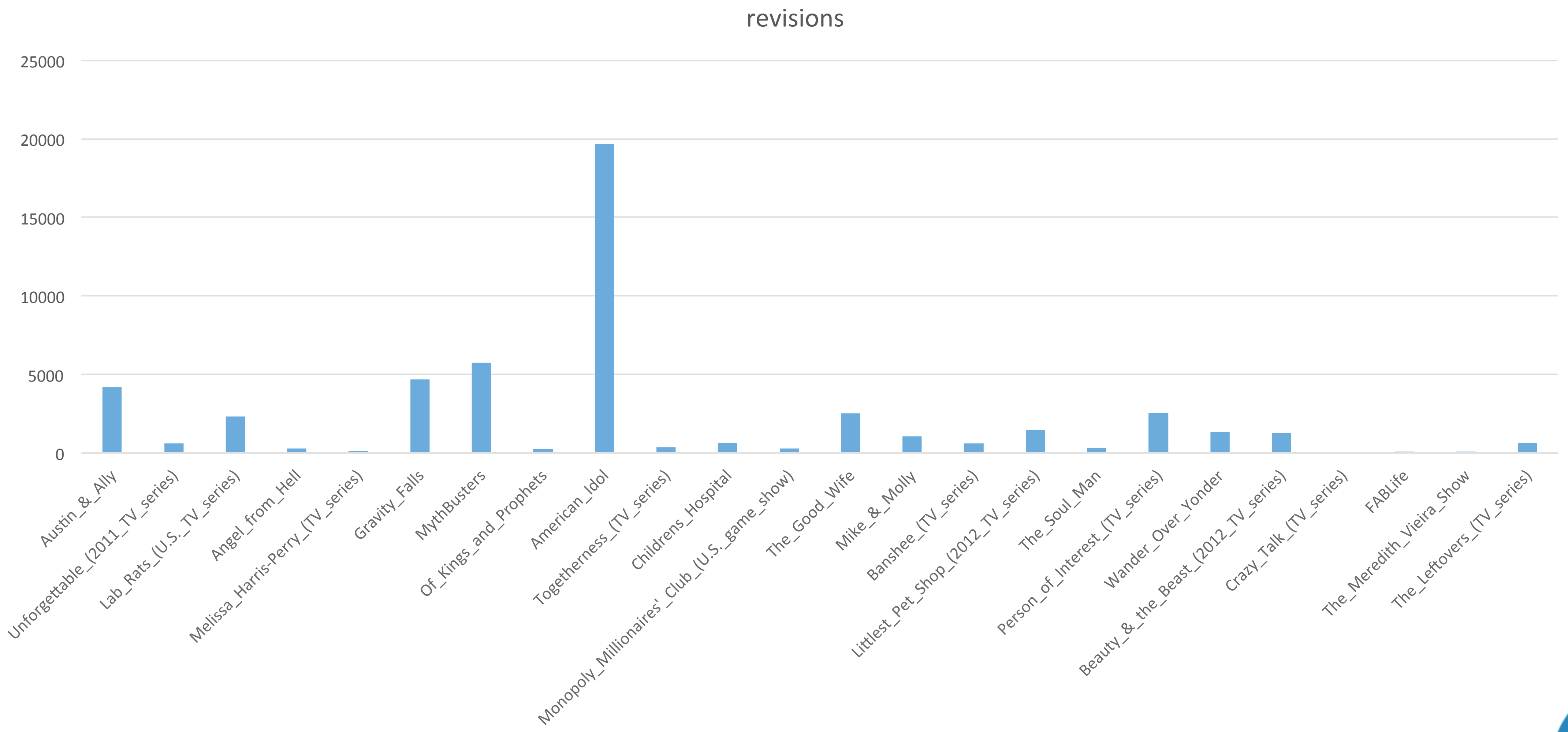
Function

Code challenge activity #3.
Write a function called *theMostRevised()*
which receives a list of keywords and return the
most highly revised keyword in Wikipedia.

Function

Use challenge03_wiki_theMostRevised.py

Function


revisions
25000
20000
15000
10000
5000
0
Austin_&_Ally
Unforgettable_(2011_TV_series)
Lab_Rats_(U.S._TV_series)
Angel_from_Hell
Melissa_Harris-Perry_(TV_series)
Gravity_Falls
MythBusters
Of_Kings_and_Prophets
American_Idol
Togetherness_(TV_series)
Childrens_Hospital
Monopoly_Millionaires'_Club_(U.S._game_show)
The_Good_Wife
Mike_&_Molly
Banshee_(TV_series)
Littlest_Pet_Shop_(2012_TV_series)
The_Soul_Man
Person_of_Interest_(TV_series)
Wander_Over_Yonder
Beauty_&_the_Beast_(2012_TV_series)
Crazy_Talk_(TV_series)
FABLife
The_Meredith_Vieira_Show
The_Leftovers_(TV_series)