Introduction to Big Data

- Developed by Dr. Keungoui KIM
- https://awekim.github.io/portfolio/

Lecture 4. Data Manipulation with Pandas I

```
from google.colab import drive
drive.mount('/content/drive')
```

Review

- Write down the expected result of the following Python codes

```
value = 5
while 0 < value:
  value = value - 1
  print(value)
print("Hello")
```

```
value = 5
while 0 < value:
  print(value)
  value = value - 1
print("Hello")
```

```
myList = [ Dance','Ballad ,'HipHop',1,2,'3','four']
for i in range(7,0,-1):
  print("Index', i, "-", myList[i-1])
```

DataFrame

```
import numpy as np
import pandas as pd
```

```
dir(pd.Series)
```

```
set(dir(pd.Series))
```

Create Dataframe with List & Array

```
numberList = [1, 2, 3]
numberList
```

```
numberArray = np.array(numberList)
numberArray
```

```
pd.DataFrame(numberList)
```

```
pd.DataFrame(numberArray)
```

```
pd.DataFrame(numberList,
             columns=['Numbers ],
             index=[ one','two ,'three'])
```

```
pd.DataFrame(numberArray,
             columns=['Numbers ],
             index=[ one','two ,'three'])
```

```
variableName = ['Numbers']
variableName
```

```
pd.DataFrame(numberList,
             columns=variableName)
```

Create Dataframe with Dictionary

```
numberDict = {'Numbers':[1,2,3]}
numberDict
```

```
pd.DataFrame(numberDict)
```

```
pd.DataFrame(numberDict,
             index=[ one','two ,'three'])
```

```
pd.DataFrame(numberDict,
             columns=['Numbers ],
             index=[ one','two ,'three'])
```

```
pd.DataFrame(numberDict,
             columns=['numbers ],
             index=[ one','two ,'three'])
```

Create Dataframe with Dictionary

```
myClass={'city': ['Dublin ,'Dublin','Dublin',
                  'London ,'London','London',
                  'Paris','Paris','Paris'],
         'year': [2018,2019,2020,
                  2018,2019,2020,
                  2018,2019,2020],
         'pop': [2.3,3.4,3.2,
                 4.3,4.4,4.2,
                 4.8,5.0,5.2]}
myClass
```

```
pd.DataFrame(myClass)
```

```
pd.DataFrame(myClass,
             columns=['city'])
```

```
pd.DataFrame(myClass,
             columns=['GDP'])
```

```
myClass_df = pd.DataFrame(myClass)
myClass_df
```

DataFrame Methods

Checking the overview of data

```
myClass_df.shape
```

```
myClass_df.dtypes
```

```
myClass_df.head()
```

```
myClass_df[['city','pop']].head()
```

```
myClass_df.values
```

```
myClass_df.columns
```

```
myClass_df.index
```

```
type(myClass_df)
```

```
myClass_df['city']
```

```
type(myClass_df['city'])
```

```
myClass_df[['city','pop']]
```

```
type(myClass_df[['city','pop']])
```

```
myClass_df['city'].unique()
```

```
myClass_df['city'].nunique()
```

```
myClass_df
```

```
myClass_df['city'].value_counts()
```

```
myClass_df['city'].value_counts(normalize=True)
```

Checking the values of specific column

```
myClass_df = pd.DataFrame(myClass)
myClass_df
```

```
myClass_df.rename(columns =
                  {'city':'capitcal city',
                   'pop': population'},
                  inplace=False)
```

```
myClass_df.head()
```

```
myClass_df.rename(columns =
                  {'city':'capitcal city',
                   'pop': population'},
                  inplace=True)
```

```
myClass_df.head()
```

```
myClass_df.rename(index = {0:'zero',1:'one'},
                  inplace=False).head()
```

```
myClass_df = pd.DataFrame(myClass)
```

```
myClass_df['city'].head()
```

```
myClass_df.city.head()
```

```
myClass_df_ed = myClass_df.rename(
    columns = { city :'capital city','pop':'population'},
    inplace=False)
myClass_df_ed
```

```
myClass_df_ed.columns
```

```
myClass_df_ed['capital city'].head()
```

```
myClass_df_ed['year'].head()
```

```
myClass_df_ed.year.head()
```

```
myClass_df_ed.capital city.head()
```

.iloc (using index number) & loc (using label)

```
myClass_df = pd.DataFrame(myClass)
myClass_df.head()
```

```
myClass_df.iloc[0]
```

```
myClass_df.iloc[[0]]
```

```
myClass_df.iloc[:2]
```

```
myClass_df.iloc[:2,1:3]
```

```
myClass_df.iloc[[0,3],1:3]
```

```
myClass_df.iloc[[0,3],[0,2]]
```

```
myClass_index=pd.DataFrame(
    { city : ['Dublin','Dublin ,'Dublin',
              'London','London ,'London',
              'Paris ,'Paris', Paris'],
      year : [2018,2019,2020,
              2018,2019,2020,
              2018,2019,2020],
      pop': [2.3,3.4,3.2,
             4.3,4.4,4.2,
             4.8,5.0,5.2]},
     index=['Dublin2018', Dublin2019','Dublin2020',
            'London2018', London2019','London2020',
            'Paris2018','Paris2019','Paris2020'])
myClass_index
```

```
myClass_index.loc['Dublin2018']
myClass_index.loc[:, city ]
```

```
myClass_index['city']
```

```
myClass_index.loc[['Dublin2018 ]]
```

```
myClass_index.loc[['London2018 ,'Paris2018']]
```

```
myClass_index.loc['Dublin2018':'Paris2018','pop']
```

```
myClass_index.city=="Dublin"
```

```
myClass_index.loc[myClass_index.city=="Dublin"]
```

```
myClass_index.year==2018
```

```
myClass_index.loc[(myClass_index.city=="Dublin ) & (myClass_index.year==2018)]
```

```
myClass_index.loc[myClass_index.city=="Dublin" & myClass_index.year==2018]
```

Filtering with isin()

```
myClass={'city': ['Dublin ,'Dublin','Dublin',
                  'London ,'London','London',
                  'Paris','Paris','Paris'],
         'year': [2018,2019,2020,
                  2018,2019,2020,
                  2018,2019,2020],
         'pop': [2.3,3.4,3.2,
                 4.3,4.4,4.2,
                 4.8,5.0,5.2]}
myClass_df = pd.DataFrame(myClass)
myClass_df
```

```
myClass_df[myClass_df.city.isin(['Dublin'])]
```

```
myClass_df[~myClass_df.city.isin(['Dublin'])]
```

```
myClass_df.loc[ myClass_df.city.isin(['Dublin']) , : ]
```

```
myClass_df.loc[ ~myClass_df.city.isin(['Dublin']) , : ]
```

Review

Given YoutubeSub, a data frame about YouTube Subscription, answer the following questions:

1) Which country owns the greatest number of YouTube Channels?

2) What are the list of music-realted YouTube Channels?

3) How many subscribers does Blankink have?

4) What is the most popular YouTube Channel?

```
import pandas as pd

YoutubeSub = pd.read_csv("/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation
YoutubeSub.head()
```

```
# 1
YoutubeSub.Country.value_counts()
```

```
# 2
YoutubeSub[YoutubeSub.Category=='Music'].Name
```

```
# 3
YoutubeSub.loc[YoutubeSub.Name=="Blackpink"]
```

```
# 4
YoutubeSub.loc[YoutubeSub['Subscribers (millions)'] == max(YoutubeSub['Subscribers (millions)'])]
```

Data Import & Export with pandas

```
import pandas as pd
```

Data import with pandas

```
sample_1 = pd.read_table( /content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation
                         sep=',')
sample_1
```

```
sample_1 = pd.read_csv('/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation/s
sample_1
```

```
sample_1 = pd.read_csv('/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation/s
                       header=None)
sample_1
```

```
header_name=['FullName','Age', Major']
sample_1 = pd.read_csv('/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation/s
                       names=header_name)
sample_1
```

Data export with pandas

```
import pandas as pd
sample_2 = pd.read_csv('/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation/s
sample_2
```

```
# export sample_3.csv
sample_2.to_csv('/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation/sample_3
```

```
# export sample_4.csv
sample_2.to_csv('/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation/sample_4
```

```
# export sample_5.csv
sample_2.to_csv('/content/drive/MyDrive/[Lecture]/IntBigData/BigData_Python/04_DataManipulation/sample_5
```