Final Assignment
ICT 6011: Data Science
Topic : Python Programming & Data Analysis

Bangladesh Agriculture University

Submitted to

Mr. Machbah Uddin
Assistant Professor
PGD in ICT Program
Department of Computer Science and Mathematics
Bangladesh Agriculture University

Submitted By

Md. Zahidur Rahman - ID: PGD1727
Mohammad Munirul Islam - ID: PGD1725
Md. Ashraful Alam - ID: PGD1726
Abul Kalam Azad - ID: PGD1728

Google Collab Source Code Link :
https://colab.research.google.com/drive/1mXL7QHTUcLl2ezE3m4LbEQRFjR2jEtJS

In this Assignment we are going to run data analysis over csv files. The csv files consist of data which contains a lot of information about canada imigration.

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	1981	1982	1983	1984	1985	1986
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	39	39	47	71	340	496
1	Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	0	0	0	0	0	1
2	Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	67	71	69	63	44	69
3	Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	1	0	0	0	0	0
4	Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	0	0	0	0	0	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
192	Immigrants	Foreigners	Yemen	935	Asia	922	Western Asia	902	Developing regions	1	2	1	6	0	18	7
193	Immigrants	Foreigners	Zambia	903	Africa	910	Eastern Africa	902	Developing regions	11	17	11	7	16	9	15
194	Immigrants	Foreigners	Zimbabwe	903	Africa	910	Eastern Africa	902	Developing regions	72	114	102	44	32	29	43
195	Immigrants	Foreigners	Unknown	999	World	999	World	999	World	44000	18078	16904	13635	14855	14368	13303
196	Immigrants	Both	Total	999	World	999	World	999	World	143137	128641	121175	89185	88272	84346	99351

Figure : data figure

We can see in the figure that there is so much datas and columns and rows. We are going to run python data analysis functions, methods & attributes to run the data analysis.

Tools

We are going to use Google Colab . a Research platform created by google for python developer and researchers.

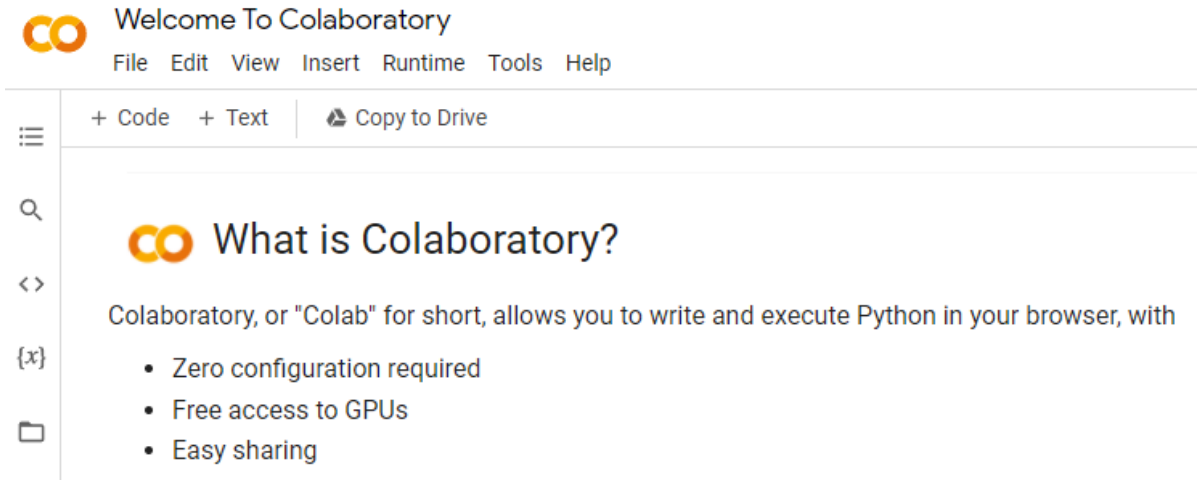

Figure : Colaboratory

Chapter One : Our Research And Analysis

We are going to import the pandas library for data analysis.Pandas help to add data structures and tools designed to work with table-like data.

```
#importing the pandas library
import pandas as pd
```

Figure : importing library

Then we are going to upload the CSV file in our Google colab platform from *google.colab import file*.

```
#loading the CSV files into Google Co
from google.colab import files
uploaded = files.upload()
```

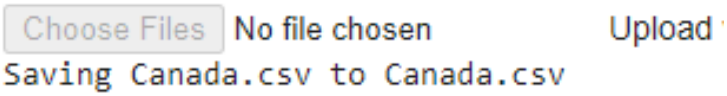

Figure : uploading the files

After uploading the files we can set the file path from google collab file section. So that we can make a dataframe from it.

Now we are going to load the CSV file into the dataframe.

```
#read the csv file in dataframe
path='/content/Canada.csv'
df = pd.read_csv(path)
```

Figure : reading data into the data frame

We can find the dataframe shape by using shape attribute

```
#finding the shape of the dataframe
df.shape
```

```
(197, 43)
```

Figure : Shape of the datas

We can see that there are 197 rows in the dataframe.

We use df.head() function to get the 5 element of data

```
df.head(5)
```

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	1981	1982
0	Immigrants	Foreigners	Afghanistan	935.0	Asia	5501.0	Southern Asia	902.0	Developing regions	16.0	39.0	39.0
1	Immigrants	Foreigners	Albania	908.0	Europe	925.0	Southern Europe	901.0	Developed regions	1.0	0.0	0.0
2	Immigrants	Foreigners	Algeria	903.0	Africa	912.0	Northern Africa	902.0	Developing regions	80.0	67.0	71.0
3	Immigrants	Foreigners	American Samoa	909.0	Oceania	957.0	Polynesia	902.0	Developing regions	0.0	1.0	0.0
4	Immigrants	Foreigners	Andorra	908.0	Europe	925.0	Southern Europe	901.0	Developed regions	0.0	0.0	0.0

Figure : first 5 data

Now we use dtypes attribute to find the data types of the columns & rows

```
df.dtypes
```

```
Type          object
Coverage      object
OdName        object
AREA         float64
AreaName      object
REG          float64
RegName       object
DEV          float64
DevName       object
1980         float64
1981         float64
1982         float64
1983         float64
1984         float64
```

Figure : Datatypes of Given data

Now we are using data frame columns attribute to get the all column names in array

```
df.columns
```

```
Index(['Type', 'Coverage', 'OdName', 'AREA', 'AreaName', 'REG', 'RegName',
       'DEV', 'DevName', '1980', '1981', '1982', '1983', '1984', '1985',
       '1986', '1987', '1988', '1989', '1990', '1991', '1992', '1993', '1994',
       '1995', '1996', '1997', '1998', '1999', '2000', '2001', '2002', '2003',
       '2004', '2005', '2006', '2007', '2008', '2009', '2010', '2011', '2012',
       '2013', 'Row_Total'],
      dtype='object')
```

Figure : Column list in array

Now by setting data frame dimension attribute we can get the dimension of the dataframe

```
[17] df.ndim

2
```

Figure : Data frame dimension

Chapter Two : Main Task :

Finding the name of the country from which the lowest number of peoples immigrated to Canada

First we are going to find the sum of all numeric data in the data frame. And add a column that contains the value of total sum of the data from data frame

```
df.loc['Column_Total']= df.sum(numeric_only=True, axis=0)
df.loc[:,'Row_Total'] = df.sum(numeric_only=True, axis=1)
print(df)
```

```
                 Type    Coverage  ...       2013    Row_Total
0           Immigrants  Foreigners  ...     2004.0     263908.0
1           Immigrants  Foreigners  ...      603.0      73732.0
2           Immigrants  Foreigners  ...     4331.0     288624.0
3           Immigrants  Foreigners  ...        0.0      11096.0
4           Immigrants  Foreigners  ...        1.0      10996.0
...                ...         ...  ...        ...         ...
193         Immigrants  Foreigners  ...       59.0      17568.0
194         Immigrants  Foreigners  ...      407.0      45252.0
195         Immigrants  Foreigners  ...     1484.0    2072224.0
196         Immigrants        Both  ...   259021.0   27708836.0
Column_Total       NaN         NaN  ...  2072168.0  231228912.0
```

Diagram : Adding new new column that contains the sum of each row

Now finding the sum of each numeric row in the Row_Total column . We are going to find out the minimum value from the Row_Total column .

```
[26] min_row=df['Row_Total'].min()
     print(min_row)

     2719.0
```

Diagram : value of lowest number in Row_Total column

Here sum total is 2719 . As AREA , REG and DEV are numerical values. Those are added to the sum. So we can consider that. The lowest value of the row is the lowest immigrant value over time

Now we are going to filter the row data by the minimum value from the Row_Total column. Thus we get the lowest sum value rows and total information of the rows

```
df_sub = df[ df['Row_Total'] ==min_row ]
df_sub.head()
```

	Type	Coverage	OdName	AREA
191	Immigrants	Foreigners	Western Sahara	903.0

Diagram : minimum row details information.

Now we are getting the country name , area name & region name

```
[20]  df_sub.OdName

191    Western Sahara
Name: OdName, dtype: object
```

```
[21]  df_sub.AreaName

191    Africa
Name: AreaName, dtype: object
```

```
[22] df_sub.RegName

191    Northern Africa
Name: RegName, dtype: object
```

By this implementation we can find that

The lowest Immigrant

Country ⇒ **Western Sahara**

Arean Name ⇒ **Africa**

Region Name ⇒ **Northern Africa**