

```
import matplotlib as mpl
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_excel("/content/Canada.xlsx",
    sheet_name='Canada by Citizenship',
    skiprows=range(20),
    skipfooter=2)
df
```

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	
1	Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	
2	Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	
3	Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	
4	Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	
...	
190	Immigrants	Foreigners	Viet Nam	935	Asia	920	South-Eastern Asia	902	Developing regions	1
191	Immigrants	Foreigners	Western Sahara	903	Africa	912	Northern Africa	902	Developing regions	
192	Immigrants	Foreigners	Yemen	935	Asia	922	Western Asia	902	Developing regions	
193	Immigrants	Foreigners	Zambia	903	Africa	910	Eastern Africa	902	Developing regions	
194	Immigrants	Foreigners	Zimbabwe	903	Africa	910	Eastern Africa	902	Developing regions	

195 rows × 43 columns



```
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],
      dtype='object')
```

```
df.info
```

<bound method DataFrame.info of										
	Type	Coverage	OdName	AREA	AreaName	REG	\			
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501				
1	Immigrants	Foreigners	Albania	908	Europe	925				
2	Immigrants	Foreigners	Algeria	903	Africa	912				
3	Immigrants	Foreigners	American Samoa	909	Oceania	957				
4	Immigrants	Foreigners	Andorra	908	Europe	925				
..				
190	Immigrants	Foreigners	Viet Nam	935	Asia	920				
191	Immigrants	Foreigners	Western Sahara	903	Africa	912				
192	Immigrants	Foreigners	Yemen	935	Asia	922				
193	Immigrants	Foreigners	Zambia	903	Africa	910				
194	Immigrants	Foreigners	Zimbabwe	903	Africa	910				
	RegName	DEV	DevName	1980	...	2004	2005	2006	\	
0	Southern Asia	902	Developing regions	16	...	2978	3436	3009		
1	Southern Europe	901	Developed regions	1	...	1450	1223	856		
2	Northern Africa	902	Developing regions	80	...	3616	3626	4807		
3	Polynesia	902	Developing regions	0	...	0	0	1		
4	Southern Europe	901	Developed regions	0	...	0	0	1		

..
190	South-Eastern Asia	902	Developing regions	1191	...	1816	1852	3153
191	Northern Africa	902	Developing regions	0	...	0	0	1
192	Western Asia	902	Developing regions	1	...	124	161	140
193	Eastern Africa	902	Developing regions	11	...	56	91	77
194	Eastern Africa	902	Developing regions	72	...	1450	615	454

	2007	2008	2009	2010	2011	2012	2013
0	2652	2111	1746	1758	2203	2635	2004
1	702	560	716	561	539	620	603
2	3623	4005	5393	4752	4325	3774	4331
3	0	0	0	0	0	0	0
4	1	0	0	0	0	1	1

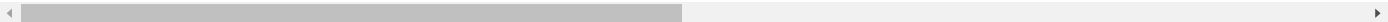
..
190	2574	1784	2171	1942	1723	1731	2112
191	0	0	0	0	0	0	0
192	122	133	128	211	160	174	217
193	71	64	60	102	69	46	59
194	663	611	508	494	434	437	407

[195 rows x 43 columns]>

df.describe()

	AREA	REG	DEV	1980	1981	1982	1983	1984	1985	1986
count	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000
mean	912.764103	1249.015385	901.753846	508.394872	566.989744	534.723077	387.435897	376.497436	358.861538	441.271795
std	13.082835	1185.526885	0.431878	1949.588546	2152.643752	1866.997511	1204.333597	1198.246371	1079.309600	1225.576630
min	903.000000	905.000000	901.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	903.000000	914.000000	902.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.500000
50%	908.000000	922.000000	902.000000	13.000000	10.000000	11.000000	12.000000	13.000000	17.000000	18.000000
75%	922.000000	925.500000	902.000000	251.500000	295.500000	275.000000	173.000000	181.000000	197.000000	254.000000
max	935.000000	5501.000000	902.000000	22045.000000	24796.000000	20620.000000	10015.000000	10170.000000	9564.000000	9470.000000

8 rows x 37 columns



df.rename(columns={'OdName':'Country', 'AreaName':'Continent', 'RegName':'Region'}, inplace=True)
df.columns

Index(['Type',	'Coverage',	'Country',	'AREA',	'Continent',
	'REG',	'Region',	'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],		
	dtype='object')				

df['Total'] = df.sum(axis=1)
df['Total']

<ipython-input-5-ebabe8175f7f>:1: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is de

df['Total'] = df.sum(axis=1)

0	65977
1	18433
2	72156
3	2774
4	2749
...	
190	99903
191	2719
192	5744
193	4392
194	11313

Name: Total, Length: 195, dtype: int64



df.isnull().sum()

```
Type      0
Coverage  0
OdName    0
AREA      0
AreaName  0
REG        0
RegName   0
DEV        0
DevName   0
1980      0
1981      0
1982      0
1983      0
1984      0
1985      0
1986      0
1987      0
1988      0
1989      0
1990      0
1991      0
1992      0
1993      0
1994      0
1995      0
1996      0
1997      0
1998      0
1999      0
2000      0
2001      0
2002      0
2003      0
2004      0
2005      0
2006      0
2007      0
2008      0
2009      0
2010      0
2011      0
2012      0
2013      0
dtype: int64

df[['OdName', 'AreaName']]

   OdName AreaName
0  Afghanistan  Asia
1    Albania  Europe
2    Algeria  Africa
3  American Samoa  Oceania
4    Andorra  Europe
...      ...     ...
190   Viet Nam  Asia
191  Western Sahara  Africa
192    Yemen  Asia
193    Zambia  Africa
194   Zimbabwe  Africa

195 rows x 2 columns

df.iloc[87, 36]

1250

yrs = list(map(str, range(1980, 2014)))
yrs

['1980',
 '1981',
```

<https://colab.research.google.com/drive/1Dko67VhrHGRk2KYe5SV5pRYVEpzMy3Wy#printMode=true>

3/9

```
'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']
```

```
df['OdName']
```

```
0      Afghanistan
1      Albania
2      Algeria
3      American Samoa
4      Andorra
...
190     Viet Nam
191  Western Sahara
192      Yemen
193      Zambia
194      Zimbabwe
```

```
Name: OdName, Length: 195, dtype: object
```

```
#VISUALIZATION
```

```
print('Matplotlib version', mpl.__version__)
```

```
Matplotlib version 3.7.1
```

```
print(plt.style.available)
```

```
mpl.style.use(['ggplot'])
```

```
['Solarize_Light2', '_classic_test_patch', '_mpl-gallery', '_mpl-gallery-nogrid', 'bmh', 'classic', 'dark_background', 'fast', 'fiveth
```

```
#Plot a line graph of immigration from Haiti using df.plot()
```

```
df = pd.read_csv("/content/Canada.csv")
```

```
df
```

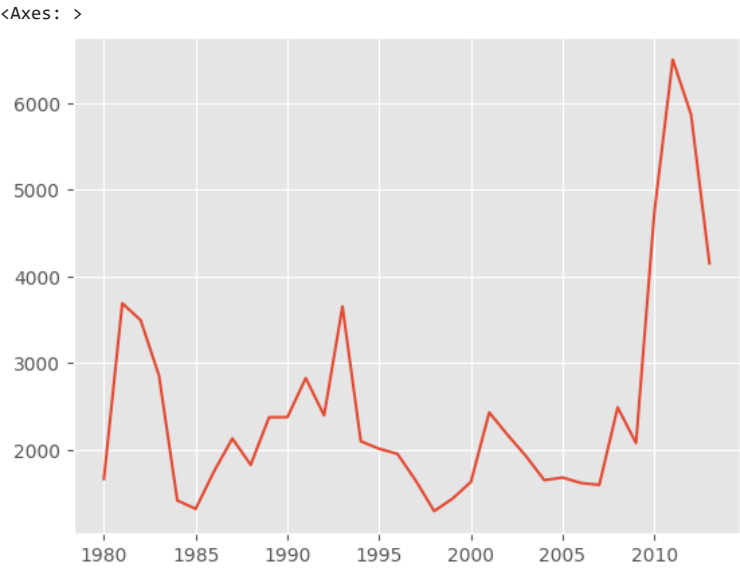
	Country	Continent	Region	DevName	1980	1981	1982	1983	1984	1985	...	2005	2006	2007	2008	2009	2010	2011	2012
0	Afghanistan	Asia	Southern Asia	Developing regions	16	39	39	47	71	340	...	3436	3009	2652	2111	1746	1758	2203	2635
1	Albania	Europe	Southern Europe	Developed regions	1	0	0	0	0	0	...	1223	856	702	560	716	561	539	620
2	Algeria	Africa	Northern Africa	Developing regions	80	67	71	69	63	44	...	3626	4807	3623	4005	5393	4752	4325	3774
3	American Samoa	Oceania	Polynesia	Developing regions	0	1	0	0	0	0	...	0	1	0	0	0	0	0	0
4	Andorra	Europe	Southern Europe	Developed regions	0	0	0	0	0	0	...	0	1	1	0	0	0	0	1
...
190	Viet Nam	Asia	South-Eastern	Developing regions	1191	1829	2162	3404	7583	5907	...	1852	3153	2574	1784	2171	1942	1723	1731

```
df.set_index('Country', inplace=True)
df.index.name = None

years = list(map(str, range(1980, 2014)))
#creating data series
haiti = df.loc['Haiti', years] # passing in years 1980 - 2013 to exclude the 'total' column
haiti.head()
```

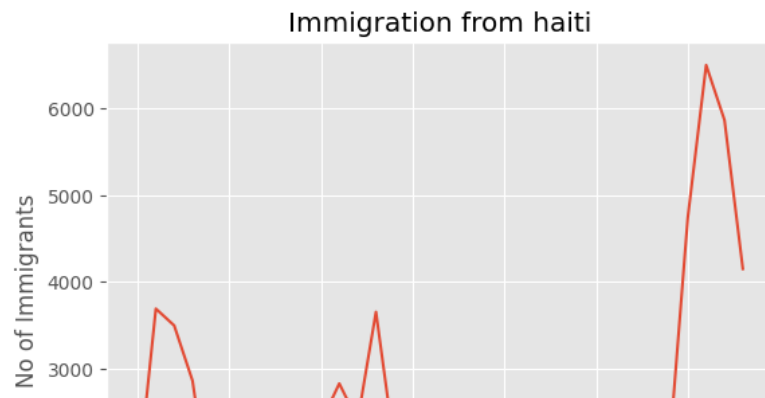
1980 1666
1981 3692
1982 3498
1983 2860
1984 1418
Name: Haiti, dtype: object

```
haiti.plot()
```



```
haiti.plot(kind='line')
plt.title('Immigration from haiti')
plt.xlabel('Years')
plt.ylabel('No of Immigrants')
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

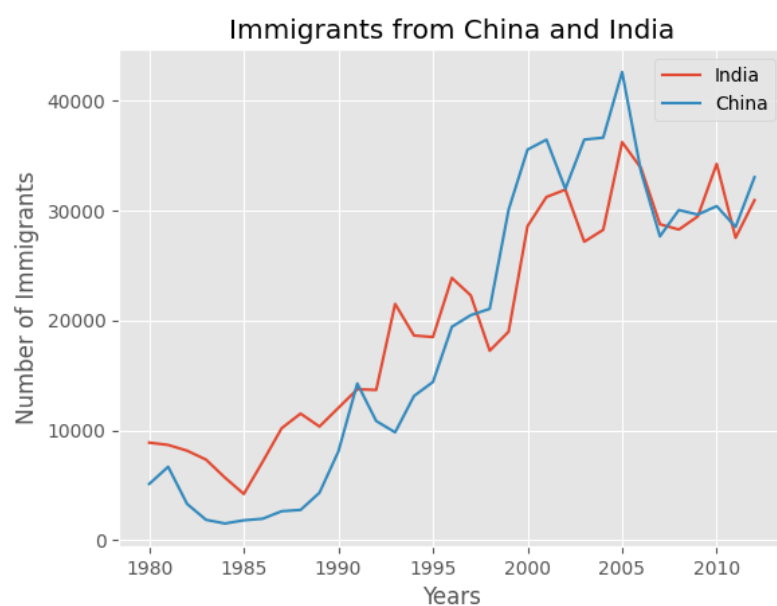


```
com1 =com.transpose()  
com1
```

	India	China
1980	8880	5123
1981	8670	6682
1982	8147	3308
1983	7338	1863
1984	5704	1527
1985	4211	1816
1986	7150	1960
1987	10189	2643

```
com1.plot(kind='line')
```

```
plt.title('Immigrants from China and India')
plt.ylabel('Number of Immigrants')
plt.xlabel('Years')
plt.show()
```



```
# Compare the trend of top 5 countries that contributed the most to immigration to Canada.
df.sort_values(by = 'Total', ascending = False, axis=0, inplace=True)
df_top5 = df.head(5)
```

```
-----
KeyError                                Traceback (most recent call last)
```

```
<ipython-input-83-900d155e7f9a> in <cell line: 2>()
```

```
1 # Compare the trend of top 5 countries that contributed the most to immigration to Canada.
```

```
----> 2 df.sort_values(by = 'Total', ascending = False, axis=0, inplace=True)
```

```
3 df_top5 = df.head(5)
```

```
↕ 2 frames
```

```
/usr/local/lib/python3.10/dist-packages/pandas/core/generic.py in _get_label_or_level_values(self, key, axis)
```

```
1848
```

```
)
```

```
1849
```

```
else:
```

```
-> 1850     raise KeyError(key)
```

```
1851
```

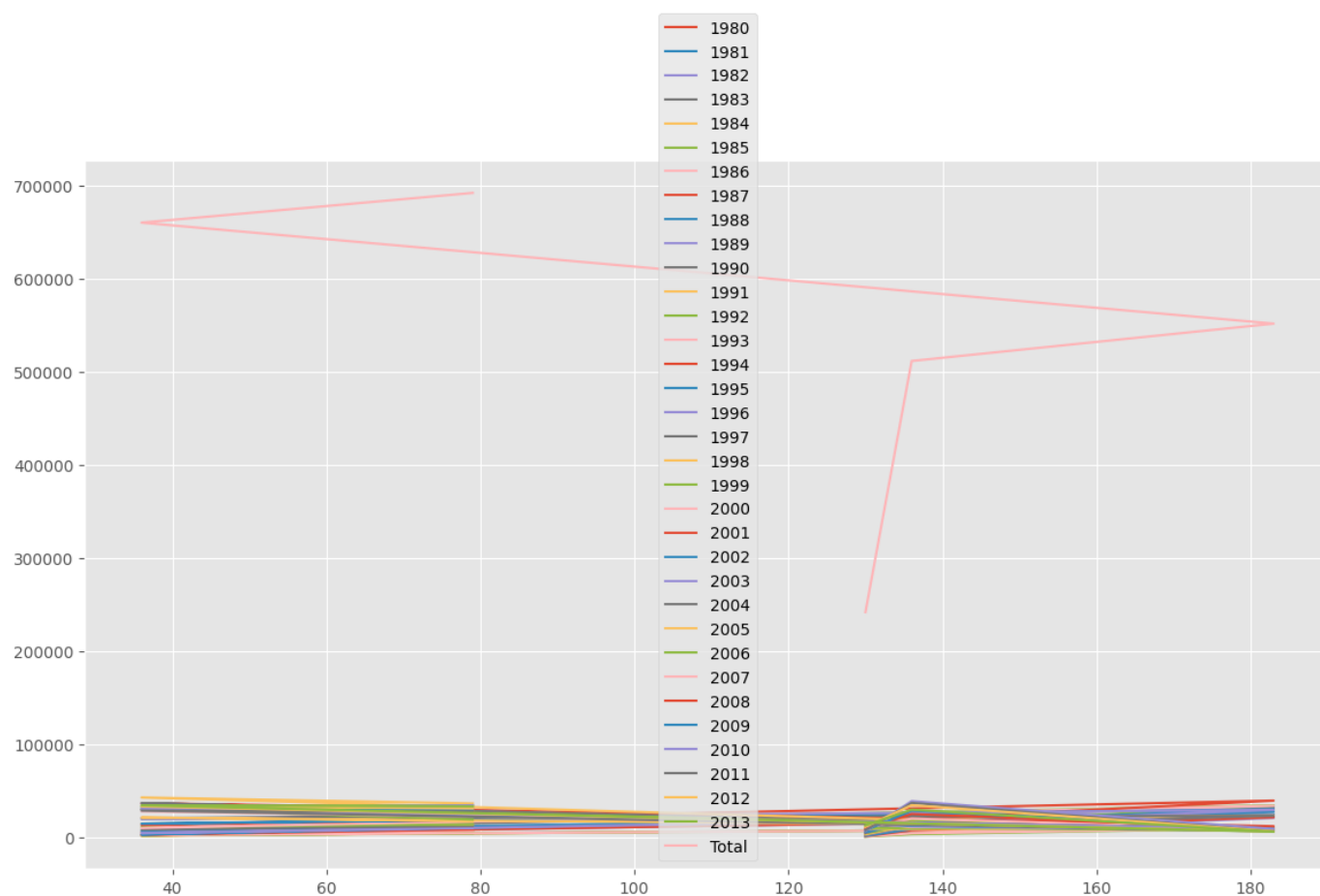
```
1852     # Check for duplicates
```

```
KeyError: 'Total'
```

SEARCH STACK OVERFLOW

```
df_top5.index = df_top5.index.map(int) # let's change the index values of df_top5 to type integer for plotting
df_top5.plot(kind='line', figsize=(14, 8)) # pass a tuple (x, y) size
```

<Axes: >



```
df_top5 = df_top5[years].transpose()
print(df_top5)
```

	79	36	183	136	130
1980	8880	5123	22045	6051	978
1981	8670	6682	24796	5921	972
1982	8147	3308	20620	5249	1201
1983	7338	1863	10015	4562	900
1984	5704	1527	10170	3801	668
1985	4211	1816	9564	3150	514
1986	7150	1960	9470	4166	691
1987	10189	2643	21337	7360	1072
1988	11522	2758	27359	8639	1334
1989	10343	4323	23795	11865	2261
1990	12041	8076	31668	12509	2470
1991	13734	14255	23380	12718	3079
1992	13673	10846	34123	13670	4071
1993	21496	9817	33720	20479	4777
1994	18620	13128	39231	19532	4666
1995	18489	14398	30145	15864	4994
1996	23859	19415	29322	13692	9125
1997	22268	20475	22965	11549	13073
1998	17241	21049	10367	8735	9068
1999	18974	30069	7045	9734	9979
2000	28572	35529	8840	10763	15400
2001	31223	36434	11728	13836	16708
2002	31889	31961	8046	11707	15110
2003	27155	36439	6797	12758	13205
2004	28235	36619	7533	14004	13399
2005	36210	42584	7258	18139	14314
2006	33848	33518	7140	18400	13127
2007	28742	27642	8216	19837	10124

2008	28261	30037	8979	24887	8994
2009	29456	29622	8876	28573	7217
2010	34235	30391	8724	38617	6811
2011	27509	28502	6204	36765	7468
2012	30933	33024	6195	34315	11227
2013	33087	34129	5827	29544	12603

```
df_top5.plot(kind='line')
plt.title('Immigration Trend of Top 5 Countries')
plt.ylabel('Number of Immigrants')
plt.xlabel('Years')
plt.show()
```

