Lab IA1 - Program (5)
Submitted By: Sintu Boro
USN: 17BTCS050

In [3]:

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

In [5]:

```
series = pd.Series([0.2, 2.5, 3.5, 1.2])
type(series)
```

Out[5]:

```
pandas.core.series.Series
```

In [6]:

```
series.index
```

Out[6]:

```
RangeIndex(start=0, stop=4, step=1)
```

In [8]:

```
series.values
```

Out[8]:

```
array([0.2, 2.5, 3.5, 1.2])
```

In [10]:

```
s[0]
```

Out[10]:

```
1.0
```

In [11]:

```
s[1]
```

Out[11]:

```
2.0
```

In [12]:

```
s[2]
```

Out[12]:

```
0.3
```

In [15]:

```
# importing numpy library
import numpy as np
```

In [18]:

```
np_array = np.array([1,2,3,4])
```

In [19]:

```
series_2 = pd.Series(np_array, index=['w', 'x', 'y', 'z'])
series_2
```

Out[19]:

```
w    1
x    2
y    3
z    4
dtype: int64
```

In [20]:

```
series_2['z']
```

Out[20]:

```
4
```

In [41]:

```
dictionary = {'USA': 81.3, 'France': 11.3, 'Colombia': 64.3, 'United Kingdom': 6
4.9, 'Netherlands': 16.9}
population = pd.Series(dictionary)
population
```

Out[41]:

```
USA               81.3
France            11.3
Colombia          64.3
United Kingdom    64.9
Netherlands       16.9
dtype: float64
```

In [42]:

```
population['USA']
```

Out[42]:

```
81.3
```

In [43]:

```
population['Netherlands']
```

Out[43]:

```
16.9
```

In [44]:

```
population * 10
```

Out[44]:

```
USA               813.0
France            113.0
Colombia          643.0
United Kingdom    649.0
Netherlands       169.0
dtype: float64
```

In [45]:

```
dict_3 = {
               "country": ["Brazil", "Russia", "India", "China", "South Africa"],
               "population": [200.4, 143.5, 1252, 1357, 52.98],
               "capital": ["Brasilia", "Moscow", "New Dehli", "Beijing", "Pretori
a"],
               "area": [8.516, 17.10, 3.286, 9.597, 1.221],
              }
```

In [46]:

```
countries = pd.DataFrame(dict_3)
countries
```

Out[46]:

	country	population	capital	area
0	Brazil	200.40	Brasilia	8.516
1	Russia	143.50	Moscow	17.100
2	India	1252.00	New Dehli	3.286
3	China	1357.00	Beijing	9.597
4	South Africa	52.98	Pretoria	1.221

In [47]:

```
countries.index
```

Out[47]:

```
RangeIndex(start=0, stop=5, step=1)
```

In [48]:

```
countries.columns
```

Out[48]:

```
Index(['country', 'population', 'capital', 'area'], dtype='object')
```

In [49]:

```
countries.dtypes
```

Out[49]:

```
country        object
population    float64
capital        object
area          float64
dtype: object
```

In [50]:

```
countries.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5 entries, 0 to 4
Data columns (total 4 columns):
country       5 non-null object
population    5 non-null float64
capital       5 non-null object
area          5 non-null float64
dtypes: float64(2), object(2)
memory usage: 240.0+ bytes
```

In [51]:

```
countries.values
```

Out[51]:

```
array([['Brazil', 200.4, 'Brasilia', 8.516],
       ['Russia', 143.5, 'Moscow', 17.1],
       ['India', 1252.0, 'New Dehli', 3.286],
       ['China', 1357.0, 'Beijing', 9.597],
       ['South Africa', 52.98, 'Pretoria', 1.221]], dtype=object)
```

In [52]:

```
countries = countries.set_index('country')
countries
```

Out[52]:

country	population	capital	area
Brazil	200.40	Brasilia	8.516
Russia	143.50	Moscow	17.100
India	1252.00	New Dehli	3.286
China	1357.00	Beijing	9.597
South Africa	52.98	Pretoria	1.221

In [53]:

```
countries['area']
```

Out[53]:

```
country
Brazil           8.516
Russia          17.100
India            3.286
China            9.597
South Africa     1.221
Name: area, dtype: float64
```

In [54]:

```
countries['capital']
```

Out[54]:

```
country
Brazil           Brasilia
Russia             Moscow
India           New Dehli
China             Beijing
South Africa     Pretoria
Name: capital, dtype: object
```

In [56]:

```
population_1 = pd.Series({'USA': 81.3, 'France': 11.3, 'Colombia': 64.3, 'United
Kingdom': 64.9, 'Netherlands': 16.9})

countries_1 = pd.DataFrame({
    'country': ['Belgium', 'France', 'Germany', 'Netherlands', 'United Kingdom'
],
    'population': [11.3, 64.3, 81.3, 16.9, 64.9],
    'area': [30510, 671308, 357050, 41526, 244820],
    'capital': ['Brussels', 'Paris', 'Berlin', 'Amsterdam', 'London']
})
```

In [58]:

```
population_1 / 100
```

Out[58]:

```
USA              0.813
France           0.113
Colombia         0.643
United Kingdom   0.649
Netherlands      0.169
dtype: float64
```

In [59]:

```
countries_1['space']=countries_1['area']/countries_1['population']
countries_1
```

Out[59]:

	country	population	area	capital	space
0	Belgium	11.3	30510	Brussels	2700.000000
1	France	64.3	671308	Paris	10440.248834
2	Germany	81.3	357050	Berlin	4391.758918
3	Netherlands	16.9	41526	Amsterdam	2457.159763
4	United Kingdom	64.9	244820	London	3772.265023

In [60]:

```
countries_1['population'] / countries_1['area']
```

Out[60]:

```
0    0.000370
1    0.000096
2    0.000228
3    0.000407
4    0.000265
dtype: float64
```

In [67]:

```
s1 = population[['USA', 'France']]
s2 = population[['France', 'United Kingdom']]
```

In [68]:

```
s1
```

Out[68]:

```
USA       81.3
France    11.3
dtype: float64
```

In [69]:

```
s2
```

Out[69]:

```
France            11.3
United Kingdom    64.9
dtype: float64
```

In [70]:

```
s1 + s2
```

Out[70]:

```
France            22.6
USA                NaN
United Kingdom     NaN
dtype: float64
```

In [71]:

```
population_1.mean()
```

Out[71]:

```
47.739999999999995
```

In [72]:

```
countries_1['space'].max()
```

Out[72]:

```
10440.248833592535
```

In [73]:

```
countries_1.median()
```

Out[73]:

```
population        64.300000
area          244820.000000
space           3772.265023
dtype: float64
```

In [74]:

```
countries_1['population']=  countries_1['population'] * 1000
```

Calculate the population numbers relative to USA.

In [75]:

```
population/population['USA'].mean()
```

Out[75]:

```
USA               1.000000
France            0.138991
Colombia          0.790898
United Kingdom    0.798278
Netherlands       0.207872
dtype: float64
```

Calculated the population density for each country and added as a new column to the dataframe.

In [76]:

```
countries_1['density'] = countries_1['population'] / countries_1['area']
```

In [78]:

```
countries_1
```

Out[78]:

	country	population	area	capital	space	density
0	Belgium	11300.0	30510	Brussels	2700.000000	0.370370
1	France	64300.0	671308	Paris	10440.248834	0.095783
2	Germany	81300.0	357050	Berlin	4391.758918	0.227699
3	Netherlands	16900.0	41526	Amsterdam	2457.159763	0.406974
4	United Kingdom	64900.0	244820	London	3772.265023	0.265093

In [79]:

```
countries_1.sort_values('density', ascending = True)
```

Out[79]:

	country	population	area	capital	space	density
1	France	64300.0	671308	Paris	10440.248834	0.095783
2	Germany	81300.0	357050	Berlin	4391.758918	0.227699
4	United Kingdom	64900.0	244820	London	3772.265023	0.265093
0	Belgium	11300.0	30510	Brussels	2700.000000	0.370370
3	Netherlands	16900.0	41526	Amsterdam	2457.159763	0.406974

In [80]:

```
countries_1.describe()
```

Out[80]:

	population	area	space	density
count	5.000000	5.000000	5.000000	5.000000
mean	47740.000000	269042.800000	4752.286508	0.273184
std	31519.644668	264012.827994	3275.775647	0.123441
min	11300.000000	30510.000000	2457.159763	0.095783
25%	16900.000000	41526.000000	2700.000000	0.227699
50%	64300.000000	244820.000000	3772.265023	0.265093
75%	64900.000000	357050.000000	4391.758918	0.370370
max	81300.000000	671308.000000	10440.248834	0.406974

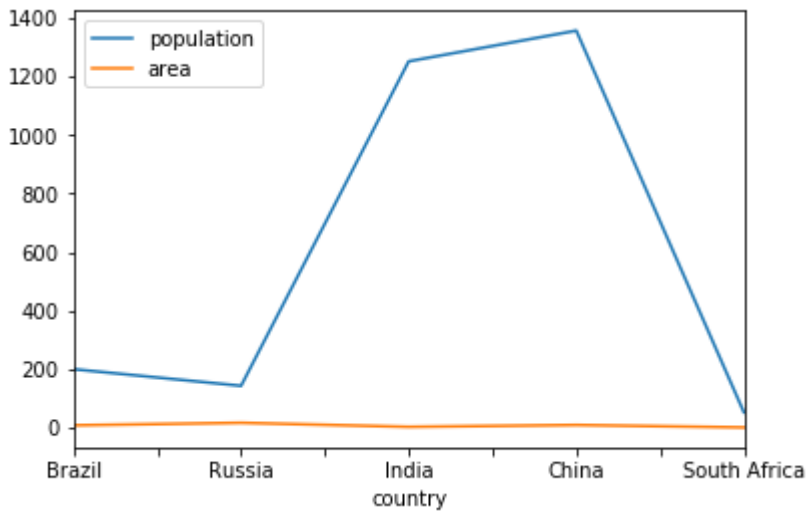
Visualizing Data using graph plots

In [84]:

```
countries.plot()
```

Out[84]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fdfe97f0550>
```

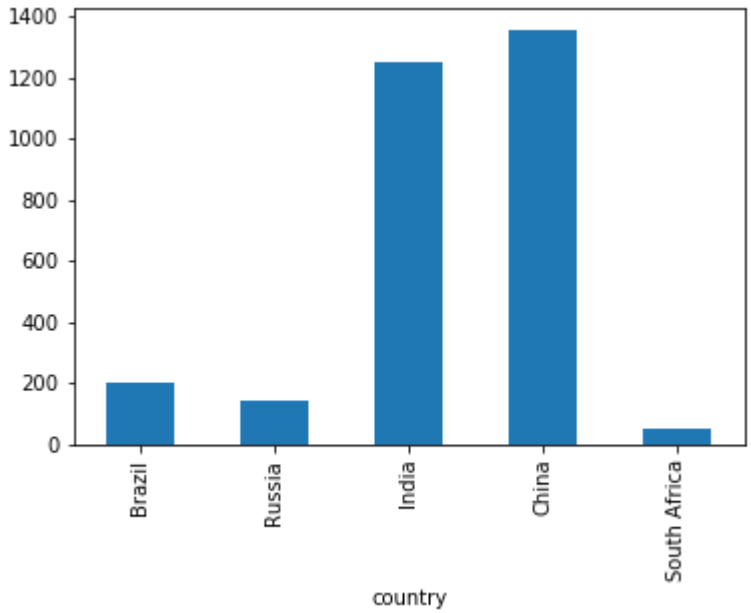


In [85]:

```
countries['population'].plot(kind='bar')
```

Out[85]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fdfe976e650>
```



***