ASSIGNMENT 10

KARTIK THKAUR

1. Create a Series from numpy array generated from random function randn and get the third place value.

In [71]:

```
import pandas as pd
import numpy as np

d=np.random.randn(5)

dp=pd.Series(d)
dp
```

Out[71]:

```
0    0.627457
1   -0.445774
2   -0.496568
3    0.087179
4   -1.476520
dtype: float64
```

In [72]:

```
# showing 3rd place value
d[3]
```

Out[72]:

```
0.08717875975930756
```

2. Create a Dataframe from dictionary having keys: Country, State, Population and values in the list upto 4 values. Population values = [200, 700,100, 500, 800, 50, 900]

In [73]:

```
country = {
    "Country": ["Nepal", "Bangladesh", "Pakistan", "Mauritius", "India", "Cana
    "State": [
        "Bagmati Province, Sudurpashchim Province, Karnali Province, Gandaki P
        "Barisal Division, Chittagong Division, Dhaka Division, Khulna Divisio
        "Azad Jammu and Kashmir, Balochistan, Gilgit-Baltistan, Islamabad Capi
        "Agaléga Islands, Black River, Cargados Carajos Shoals (Saint Brandon)
        "Andhra Pradesh, Arunachal Pradesh, Assam, Bihar, Chhattisgarh, Goa, G
        "Alberta, British Columbia, Manitoba, New Brunswick, Newfoundland and 
        "Anhui, Fujian, Gansu, Guangdong, Guizhou, Hainan, Hebei, Heilongjiang
    ],
    "Population": [200, 700, 100, 500, 800, 50, 900]
}

d=pd.DataFrame(country)
d
```

Out[73]:

	Country	State	Population
0	Nepal	Bagmati Province, Sudurpashchim Province, Karn...	200
1	Bangladesh	Barisal Division, Chittagong Division, Dhaka D...	700
2	Pakistan	Azad Jammu and Kashmir, Balochistan, Gilgit-Ba...	100
3	Mauritius	Agaléga Islands, Black River, Cargados Carajos...	500
4	India	Andhra Pradesh, Arunachal Pradesh, Assam, Biha...	800
5	Canada	Alberta, British Columbia, Manitoba, New Bruns...	50
6	China	Anhui, Fujian, Gansu, Guangdong, Guizhou, Hain...	900

a) Describe the dataframe by excluding object type data

In [74]:
```
d_desc = d.describe(exclude='object')
d_desc
```

Out[74]:

	Population
count	7.000000
mean	464.285714
std	349.659699
min	50.000000
25%	150.000000
50%	500.000000
75%	750.000000
max	900.000000

b) Sort the country according to the population

In [75]:
```
d_sorted = df.sort_values(by='Population') # Low to high
d_sorted
```

Out[75]:

	Country	Population
5	Canada	50
2	Pakistan	100
0	Nepal	200
3	Mauritius	500
1	Bangladesh	700
4	India	800
6	China	900

c) Get the values of second row

In [76]:
```
second_row = d.iloc[1]
print(second_row)
```

```
Country                                                     Bangladesh
State         Barisal Division, Chittagong Division, Dhaka D...
Population                                                         700
Name: 1, dtype: object
```

d) Sort the country names alphabetically

In [77]:

```
df = d.sort_values('Country')
print(df)
```

```
     Country                                              State  Population
1  Bangladesh  Barisal Division, Chittagong Division, Dhaka D...         700
5      Canada  Alberta, British Columbia, Manitoba, New Bruns...          50
6       China  Anhui, Fujian, Gansu, Guangdong, Guizhou, Hain...         900
4       India  Andhra Pradesh, Arunachal Pradesh, Assam, Biha...         800
3   Mauritius  Agaléga Islands, Black River, Cargados Carajos...         500
0       Nepal  Bagmati Province, Sudurpashchim Province, Karn...         200
2    Pakistan  Azad Jammu and Kashmir, Balochistan, Gilgit-Ba...         100
```

e) Create the new column with name capital

In [78]:

```
capitals = ["Kathmandu", "Dhaka", "Islamabad", "Port Louis", "New Delhi", "Ott

df['Capital'] = capitals
print(df)
```

```
     Country                                              State  Population
\
1  Bangladesh  Barisal Division, Chittagong Division, Dhaka D...         700
5      Canada  Alberta, British Columbia, Manitoba, New Bruns...          50
6       China  Anhui, Fujian, Gansu, Guangdong, Guizhou, Hain...         900
4       India  Andhra Pradesh, Arunachal Pradesh, Assam, Biha...         800
3   Mauritius  Agaléga Islands, Black River, Cargados Carajos...         500
0       Nepal  Bagmati Province, Sudurpashchim Province, Karn...         200
2    Pakistan  Azad Jammu and Kashmir, Balochistan, Gilgit-Ba...         100

     Capital
1  Kathmandu
5      Dhaka
6  Islamabad
4 Port Louis
3  New Delhi
0     Ottawa
2    Beijing
```

f) Get the population of country more than 500

In [79]:

```
pop_more_than_500 = d[d['Population'] > 500]['Population']
print(pop_more_than_500)
```

```
1    700
4    800
6    900
Name: Population, dtype: int64
```

g) Remove the state column from the dataframe

In [80]:

```
df = d.drop('State', axis=1)
print(df)
```

```
      Country  Population
0       Nepal         200
1  Bangladesh         700
2    Pakistan         100
3   Mauritius         500
4       India         800
5      Canada          50
6       China         900
```

h) Change the column name Population to %_Population

In [86]:

```
d.rename(columns={'Population': '%_Population'}, inplace=True)
```

Out[86]:

	Country	State	%_Population
0	Nepal	Bagmati Province, Sudurpashchim Province, Karn...	200
1	Bangladesh	Barisal Division, Chittagong Division, Dhaka D...	700
2	Pakistan	Azad Jammu and Kashmir, Balochistan, Gilgit-Ba...	100
3	Mauritius	Agaléga Islands, Black River, Cargados Carajos...	500
4	India	Andhra Pradesh, Arunachal Pradesh, Assam, Biha...	800
5	Canada	Alberta, British Columbia, Manitoba, New Bruns...	50
6	China	Anhui, Fujian, Gansu, Guangdong, Guizhou, Hain...	900

i) Find the maximum populated country

In [88]:

```
pop_by_country = d.groupby('Country')['%_Population'].sum()

# sort the resulting dataframe in descending order
pop_by_country = pop_by_country.sort_values(ascending=False)

# get the most populated country
most_populated_country = pop_by_country.index[0]

print('The most populated country is:', most_populated_country)
```

```
The most populated country is: China
```

In [ ]:

In [ ]: