More About PANDAS (Python Data Analysis Library)

This Notebooks contains:

- Installation of pandas
- creating object
- working with pandas different functions
- Calculating Average of dataframe

In [ ]:
```
import pandas as pd
import numpy as np

#object creation
S= pd.Series([1,3,np.nan,5,7,8,9]) #np.nan will create a null value in series
S
```

Out[ ]:
```
0    1.0
1    3.0
2    NaN
3    5.0
4    7.0
5    8.0
6    9.0
dtype: float64
```

In [ ]:
```
#display date range
dates =pd.date_range("20220101", periods =6)
dates
```

Out[ ]:
```
DatetimeIndex(['2022-01-01', '2022-01-02', '2022-01-03', '2022-01-04',
               '2022-01-05', '2022-01-06'],
              dtype='datetime64[ns]', freq='D')
```

In [ ]:
```
#np.random.randn(rows, col), columns = list(colname)
dates =pd.date_range("20220101", periods =20)
```

```
dates
df = pd.DataFrame(np.random.randn(20,4), index=dates, columns = list("ABCD"))
df
```

Out[ ]:

	A	B	C	D
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-05	-0.979584	0.461584	-0.665996	1.194677
2022-01-06	0.159986	-0.678950	-2.291788	1.138026
2022-01-07	-0.442468	1.082353	-2.133290	0.217281
2022-01-08	-1.283584	1.408376	-0.924864	0.653886
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353
2022-01-11	-1.386227	0.161764	0.827493	-0.074207
2022-01-12	1.421998	-0.280355	-0.012034	-0.098590
2022-01-13	1.554924	-0.268234	-0.429994	0.923679
2022-01-14	-1.837905	-1.461330	0.402138	-1.256550
2022-01-15	0.088351	0.008429	1.079971	0.541435
2022-01-16	-1.778927	1.583108	-0.221253	-0.611673
2022-01-17	2.220360	-0.132794	1.143537	0.639487
2022-01-18	0.846034	0.807254	-1.586870	0.404676
2022-01-19	0.152072	0.197637	-0.782905	-1.716974
2022-01-20	-0.378237	0.363469	0.003681	0.391145

In [ ]:

```
import pandas as pd
import numpy as np
# creating dataset using dictionaries
df2 =pd.DataFrame(
    {
        "A": 1.0,
        "B": pd.Timestamp("20220111"),
        "C": pd.Series(1, index=list(range(4)), dtype="float32"),
        "D": np.array([3] * 4, dtype ="int32"),
        "E": pd.Categorical(["girl", "woman","girl", "woman" ]),
        "F": "females",
    }
)
df2
```

Out[ ]:

	A	B	C	D	E	F
0	1.0	2022-01-11	1.0	3	girl	females
1	1.0	2022-01-11	1.0	3	woman	females
2	1.0	2022-01-11	1.0	3	girl	females
3	1.0	2022-01-11	1.0	3	woman	females

In [ ]:

```
#Checking datatype of all columns
df2.dtypes
```

Out[ ]:

```
A           float64
B    datetime64[ns]
C           float32
D             int32
E          category
F            object
dtype: object
```

In [ ]:

```
# display first five rows of the dataframe
df.head()
```

Out[ ]:

	A	B	C	D
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-05	-0.979584	0.461584	-0.665996	1.194677

In [ ]:
```
#df.head(4) shows first four row's data
df.head(4)
```

Out[ ]:

	A	B	C	D
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208
2022-01-04	0.644303	0.509191	0.565688	1.725495

In [ ]:
```
#last two row of data
df.tail(2)
```

Out[ ]:

	A	B	C	D
2022-01-19	0.152072	0.197637	-0.782905	-1.716974
2022-01-20	-0.378237	0.363469	0.003681	0.391145

In [ ]:
```
#show index column
df.index
```

Out[ ]:
```
DatetimeIndex(['2022-01-01', '2022-01-02', '2022-01-03', '2022-01-04',
               '2022-01-05', '2022-01-06', '2022-01-07', '2022-01-08',
               '2022-01-09', '2022-01-10', '2022-01-11', '2022-01-12',
               '2022-01-13', '2022-01-14', '2022-01-15', '2022-01-16',
               '2022-01-17', '2022-01-18', '2022-01-19', '2022-01-20'],
              dtype='datetime64[ns]', freq='D')
```

In [ ]:
```
df2.index
```

Out[ ]:
```
Int64Index([0, 1, 2, 3], dtype='int64')
```

In [ ]:
```
#converting df to array
df.to_numpy()
```

Out[ ]:
```
array([[-1.01541149, -0.68587768,  0.14653965,  0.4884602 ],
       [ 0.42319345, -0.6211333 , -0.09324943,  0.39310133],
       [-0.41257089, -1.43664762,  0.57332639,  0.30920777],
       [ 0.6443031 ,  0.50919095,  0.56568826,  1.72549491],
       [-0.97958433,  0.46158409, -0.66599577,  1.1946775 ],
       [ 0.15998606, -0.67895046, -2.2917881 ,  1.13802592],
       [-0.44246843,  1.08235274, -2.13328993,  0.21728091],
       [-1.28358414,  1.40837572, -0.92486445,  0.65388573],
       [ 0.4611936 ,  0.49853419,  0.17186612,  1.6827729 ],
       [-0.75056027, -0.24570749, -0.00448594,  1.0703527 ],
       [-1.38622704,  0.16176377,  0.82749254, -0.07420728],
       [ 1.42199771, -0.28035532, -0.0120336 , -0.09858977],
       [ 1.55492448, -0.26823368, -0.4299944 ,  0.923679  ],
       [-1.83790527, -1.46133001,  0.40213757, -1.2565498 ],
       [ 0.08835132,  0.00842912,  1.07997082,  0.54143458],
       [-1.77892692,  1.58310812, -0.22125275, -0.61167294],
       [ 2.22036021, -0.13279437,  1.14353653,  0.63948746],
       [ 0.84603395,  0.80725434, -1.58687014,  0.40467573],
       [ 0.15207177,  0.19763714, -0.78290483, -1.71697375],
       [-0.37823696,  0.36346899,  0.00368106,  0.39114548]])
```

In [ ]:
```
df2.to_numpy()
```

Out[ ]:
```
array([[1.0, Timestamp('2022-01-11 00:00:00'), 1.0, 3, 'girl', 'females'],
       [1.0, Timestamp('2022-01-11 00:00:00'), 1.0, 3, 'woman',
        'females'],
       [1.0, Timestamp('2022-01-11 00:00:00'), 1.0, 3, 'girl', 'females'],
       [1.0, Timestamp('2022-01-11 00:00:00'), 1.0, 3, 'woman',
        'females']], dtype=object)
```

In [ ]:
```
# display details of all numeric columns
df.describe()
```

Out[ ]:

	A	B	C	D
count	20.000000	20.000000	20.000000	20.000000
mean	-0.114653	0.063533	-0.211625	0.400784
std	1.120642	0.823844	0.962465	0.865002
min	-1.837905	-1.461330	-2.291788	-1.716974
25%	-0.988541	-0.365550	-0.695223	0.144409
50%	-0.144943	0.085096	-0.008260	0.446568
75%	0.506971	0.501198	0.443025	0.960347
max	2.220360	1.583108	1.143537	1.725495

In [ ]:
```
#Transpose of data
df2.T
```

Out[ ]:

	0	1	2	3
A	1.0	1.0	1.0	1.0
B	2022-01-11 00:00:00	2022-01-11 00:00:00	2022-01-11 00:00:00	2022-01-11 00:00:00
C	1.0	1.0	1.0	1.0
D	3	3	3	3
E	girl	woman	girl	woman
F	females	females	females	females

In [ ]:

```
# sorting values in index column
df.sort_index()
```

Out[ ]:

	A	B	C	D
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-05	-0.979584	0.461584	-0.665996	1.194677
2022-01-06	0.159986	-0.678950	-2.291788	1.138026
2022-01-07	-0.442468	1.082353	-2.133290	0.217281
2022-01-08	-1.283584	1.408376	-0.924864	0.653886
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353
2022-01-11	-1.386227	0.161764	0.827493	-0.074207
2022-01-12	1.421998	-0.280355	-0.012034	-0.098590
2022-01-13	1.554924	-0.268234	-0.429994	0.923679
2022-01-14	-1.837905	-1.461330	0.402138	-1.256550
2022-01-15	0.088351	0.008429	1.079971	0.541435
2022-01-16	-1.778927	1.583108	-0.221253	-0.611673
2022-01-17	2.220360	-0.132794	1.143537	0.639487
2022-01-18	0.846034	0.807254	-1.586870	0.404676
2022-01-19	0.152072	0.197637	-0.782905	-1.716974
2022-01-20	-0.378237	0.363469	0.003681	0.391145

In [ ]:

```
#Sorting Data
df.sort_index(axis=1, ascending = True)
```

Out[ ]:

	A	B	C	D
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-05	-0.979584	0.461584	-0.665996	1.194677
2022-01-06	0.159986	-0.678950	-2.291788	1.138026
2022-01-07	-0.442468	1.082353	-2.133290	0.217281
2022-01-08	-1.283584	1.408376	-0.924864	0.653886
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353
2022-01-11	-1.386227	0.161764	0.827493	-0.074207
2022-01-12	1.421998	-0.280355	-0.012034	-0.098590
2022-01-13	1.554924	-0.268234	-0.429994	0.923679
2022-01-14	-1.837905	-1.461330	0.402138	-1.256550
2022-01-15	0.088351	0.008429	1.079971	0.541435
2022-01-16	-1.778927	1.583108	-0.221253	-0.611673
2022-01-17	2.220360	-0.132794	1.143537	0.639487
2022-01-18	0.846034	0.807254	-1.586870	0.404676
2022-01-19	0.152072	0.197637	-0.782905	-1.716974
2022-01-20	-0.378237	0.363469	0.003681	0.391145

In [ ]:

```
df.sort_values(by="B", ascending=True)
```

Out[ ]:

	A	B	C	D
2022-01-14	-1.837905	-1.461330	0.402138	-1.256550
2022-01-03	-0.412571	-1.436648	0.573326	0.309208
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-06	0.159986	-0.678950	-2.291788	1.138026
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-12	1.421998	-0.280355	-0.012034	-0.098590
2022-01-13	1.554924	-0.268234	-0.429994	0.923679
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353
2022-01-17	2.220360	-0.132794	1.143537	0.639487
2022-01-15	0.088351	0.008429	1.079971	0.541435
2022-01-11	-1.386227	0.161764	0.827493	-0.074207
2022-01-19	0.152072	0.197637	-0.782905	-1.716974
2022-01-20	-0.378237	0.363469	0.003681	0.391145
2022-01-05	-0.979584	0.461584	-0.665996	1.194677
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-18	0.846034	0.807254	-1.586870	0.404676
2022-01-07	-0.442468	1.082353	-2.133290	0.217281
2022-01-08	-1.283584	1.408376	-0.924864	0.653886
2022-01-16	-1.778927	1.583108	-0.221253	-0.611673

In [ ]:

```
df["D"]
```

Out[ ]:
```
2022-01-01    0.488460
2022-01-02    0.393101
2022-01-03    0.309208
2022-01-04    1.725495
2022-01-05    1.194677
2022-01-06    1.138026
2022-01-07    0.217281
2022-01-08    0.653886
2022-01-09    1.682773
2022-01-10    1.070353
2022-01-11   -0.074207
2022-01-12   -0.098590
2022-01-13    0.923679
2022-01-14   -1.256550
2022-01-15    0.541435
2022-01-16   -0.611673
2022-01-17    0.639487
2022-01-18    0.404676
2022-01-19   -1.716974
2022-01-20    0.391145
Freq: D, Name: D, dtype: float64
```

In [ ]:
```
#row wise selection
df[0:10]
```

Out[ ]:

	A	B	C	D
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-05	-0.979584	0.461584	-0.665996	1.194677
2022-01-06	0.159986	-0.678950	-2.291788	1.138026
2022-01-07	-0.442468	1.082353	-2.133290	0.217281
2022-01-08	-1.283584	1.408376	-0.924864	0.653886
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353

In [ ]:

```
# display first two row
df[1:3]
```

Out[ ]:

	A	B	C	D
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208

In [ ]:

```
#  displays values of row at index 0
df.loc[dates[0]]
```

Out[ ]:

```
A   -1.015411
B   -0.685878
C    0.146540
D    0.488460
Name: 2022-01-01 00:00:00, dtype: float64
```

In [ ]:
```
#  displays values of row at index 2
df.loc[dates[2]]
```

Out[ ]:
```
A   -0.412571
B   -1.436648
C    0.573326
D    0.309208
Name: 2022-01-03 00:00:00, dtype: float64
```

In [ ]:
```
#  displays values of row at index 15
df.loc[dates[15]]
```

Out[ ]:
```
A   -1.778927
B    1.583108
C   -0.221253
D   -0.611673
Name: 2022-01-16 00:00:00, dtype: float64
```

In [ ]:
```
#  displays values of col A and B
df.loc[:,["A","B"]]
```

Out[ ]:

	A	B
2022-01-01	-1.015411	-0.685878
2022-01-02	0.423193	-0.621133
2022-01-03	-0.412571	-1.436648
2022-01-04	0.644303	0.509191
2022-01-05	-0.979584	0.461584
2022-01-06	0.159986	-0.678950
2022-01-07	-0.442468	1.082353
2022-01-08	-1.283584	1.408376
2022-01-09	0.461194	0.498534
2022-01-10	-0.750560	-0.245707
2022-01-11	-1.386227	0.161764
2022-01-12	1.421998	-0.280355
2022-01-13	1.554924	-0.268234
2022-01-14	-1.837905	-1.461330
2022-01-15	0.088351	0.008429
2022-01-16	-1.778927	1.583108
2022-01-17	2.220360	-0.132794
2022-01-18	0.846034	0.807254
2022-01-19	0.152072	0.197637
2022-01-20	-0.378237	0.363469

In [ ]:

```
df.loc[:,["C","D"]]
```

Out[ ]:

	C	D
2022-01-01	0.146540	0.488460
2022-01-02	-0.093249	0.393101
2022-01-03	0.573326	0.309208
2022-01-04	0.565688	1.725495
2022-01-05	-0.665996	1.194677
2022-01-06	-2.291788	1.138026
2022-01-07	-2.133290	0.217281
2022-01-08	-0.924864	0.653886
2022-01-09	0.171866	1.682773
2022-01-10	-0.004486	1.070353
2022-01-11	0.827493	-0.074207
2022-01-12	-0.012034	-0.098590
2022-01-13	-0.429994	0.923679
2022-01-14	0.402138	-1.256550
2022-01-15	1.079971	0.541435
2022-01-16	-0.221253	-0.611673
2022-01-17	1.143537	0.639487
2022-01-18	-1.586870	0.404676
2022-01-19	-0.782905	-1.716974
2022-01-20	0.003681	0.391145

In [ ]:

```
#  displays values of specific rows from col A and B
df.loc["20220102":"20220104",["A","B"]]
```

Out[ ]:

	A	B
2022-01-02	0.423193	-0.621133
2022-01-03	-0.412571	-1.436648
2022-01-04	0.644303	0.509191

In [ ]:

```
#  displays values of specific rows from col A, B and C
df.loc["20220102":"20220110",["A","B","C"]]
```

Out[ ]:

	A	B	C
2022-01-02	0.423193	-0.621133	-0.093249
2022-01-03	-0.412571	-1.436648	0.573326
2022-01-04	0.644303	0.509191	0.565688
2022-01-05	-0.979584	0.461584	-0.665996
2022-01-06	0.159986	-0.678950	-2.291788
2022-01-07	-0.442468	1.082353	-2.133290
2022-01-08	-1.283584	1.408376	-0.924864
2022-01-09	0.461194	0.498534	0.171866
2022-01-10	-0.750560	-0.245707	-0.004486

In [ ]:

```
df.loc["20220102",["A","B","C"]]
```

Out[ ]:

```
A    0.423193
B   -0.621133
C   -0.093249
Name: 2022-01-02 00:00:00, dtype: float64
```

In [ ]:

```
#to find value at specific date in specific column
```

```
df.at[dates[0],"A"]
```

Out[ ]:
```
-1.0154114936695335
```

In [ ]:
```
df.at[dates[0],"B"]
```

Out[ ]:
```
-0.6858776811731602
```

In [ ]:
```
#shows row wise data at specific date i.e. at index 4 date
df.iloc[4]
```

Out[ ]:
```
A   -0.979584
B    0.461584
C   -0.665996
D    1.194677
Name: 2022-01-05 00:00:00, dtype: float64
```

In [ ]:
```
# show data via given range
df.iloc[3:10]
```

Out[ ]:

	A	B	C	D
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-05	-0.979584	0.461584	-0.665996	1.194677
2022-01-06	0.159986	-0.678950	-2.291788	1.138026
2022-01-07	-0.442468	1.082353	-2.133290	0.217281
2022-01-08	-1.283584	1.408376	-0.924864	0.653886
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353

In [ ]:
```
#df.iloc[row_range,col_range]
```

```
df.iloc[0:5, 0:3]
```

Out[ ]:

	A	B	C
2022-01-01	-1.015411	-0.685878	0.146540
2022-01-02	0.423193	-0.621133	-0.093249
2022-01-03	-0.412571	-1.436648	0.573326
2022-01-04	0.644303	0.509191	0.565688
2022-01-05	-0.979584	0.461584	-0.665996

In [ ]:

```
#show data with rows only
df.iloc[0:3, :]
```

Out[ ]:

	A	B	C	D
2022-01-01	-1.015411	-0.685878	0.146540	0.488460
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-03	-0.412571	-1.436648	0.573326	0.309208

In [ ]:

```
#show data with cols only
df.iloc[:, 0:2]
```

Out[ ]:

	A	B
2022-01-01	-1.015411	-0.685878
2022-01-02	0.423193	-0.621133
2022-01-03	-0.412571	-1.436648
2022-01-04	0.644303	0.509191
2022-01-05	-0.979584	0.461584
2022-01-06	0.159986	-0.678950
2022-01-07	-0.442468	1.082353
2022-01-08	-1.283584	1.408376
2022-01-09	0.461194	0.498534
2022-01-10	-0.750560	-0.245707
2022-01-11	-1.386227	0.161764
2022-01-12	1.421998	-0.280355
2022-01-13	1.554924	-0.268234
2022-01-14	-1.837905	-1.461330
2022-01-15	0.088351	0.008429
2022-01-16	-1.778927	1.583108
2022-01-17	2.220360	-0.132794
2022-01-18	0.846034	0.807254
2022-01-19	0.152072	0.197637
2022-01-20	-0.378237	0.363469

In [ ]:

```
#Boolean Operators
df[df["A"]>0]
```

Out[ ]:

	A	B	C	D
2022-01-02	0.423193	-0.621133	-0.093249	0.393101
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-06	0.159986	-0.678950	-2.291788	1.138026
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-12	1.421998	-0.280355	-0.012034	-0.098590
2022-01-13	1.554924	-0.268234	-0.429994	0.923679
2022-01-15	0.088351	0.008429	1.079971	0.541435
2022-01-17	2.220360	-0.132794	1.143537	0.639487
2022-01-18	0.846034	0.807254	-1.586870	0.404676
2022-01-19	0.152072	0.197637	-0.782905	-1.716974

In [ ]:

```
#Boolean Operators
df[df>0]
```

Out[ ]:

	A	B	C	D
2022-01-01	NaN	NaN	0.146540	0.488460
2022-01-02	0.423193	NaN	NaN	0.393101
2022-01-03	NaN	NaN	0.573326	0.309208
2022-01-04	0.644303	0.509191	0.565688	1.725495
2022-01-05	NaN	0.461584	NaN	1.194677
2022-01-06	0.159986	NaN	NaN	1.138026
2022-01-07	NaN	1.082353	NaN	0.217281
2022-01-08	NaN	1.408376	NaN	0.653886
2022-01-09	0.461194	0.498534	0.171866	1.682773
2022-01-10	NaN	NaN	NaN	1.070353
2022-01-11	NaN	0.161764	0.827493	NaN
2022-01-12	1.421998	NaN	NaN	NaN
2022-01-13	1.554924	NaN	NaN	0.923679
2022-01-14	NaN	NaN	0.402138	NaN
2022-01-15	0.088351	0.008429	1.079971	0.541435
2022-01-16	NaN	1.583108	NaN	NaN
2022-01-17	2.220360	NaN	1.143537	0.639487
2022-01-18	0.846034	0.807254	NaN	0.404676
2022-01-19	0.152072	0.197637	NaN	NaN
2022-01-20	NaN	0.363469	0.003681	0.391145

In [ ]:

```
#copy a dataframe
df3= df2.copy()
```

In [ ]:

```
#Adding new col in df2 dataset
df["E"]= ["one","one","two","three","four","three",
"one","one","two","three","four","three",
"one","one","two","three","four","three",
"four","three"]
df
```

Out[ ]:

	A	B	C	D	E
2022-01-01	-1.015411	-0.685878	0.146540	0.488460	one
2022-01-02	0.423193	-0.621133	-0.093249	0.393101	one
2022-01-03	-0.412571	-1.436648	0.573326	0.309208	two
2022-01-04	0.644303	0.509191	0.565688	1.725495	three
2022-01-05	-0.979584	0.461584	-0.665996	1.194677	four
2022-01-06	0.159986	-0.678950	-2.291788	1.138026	three
2022-01-07	-0.442468	1.082353	-2.133290	0.217281	one
2022-01-08	-1.283584	1.408376	-0.924864	0.653886	one
2022-01-09	0.461194	0.498534	0.171866	1.682773	two
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353	three
2022-01-11	-1.386227	0.161764	0.827493	-0.074207	four
2022-01-12	1.421998	-0.280355	-0.012034	-0.098590	three
2022-01-13	1.554924	-0.268234	-0.429994	0.923679	one
2022-01-14	-1.837905	-1.461330	0.402138	-1.256550	one
2022-01-15	0.088351	0.008429	1.079971	0.541435	two
2022-01-16	-1.778927	1.583108	-0.221253	-0.611673	three
2022-01-17	2.220360	-0.132794	1.143537	0.639487	four
2022-01-18	0.846034	0.807254	-1.586870	0.404676	three
2022-01-19	0.152072	0.197637	-0.782905	-1.716974	four
2022-01-20	-0.378237	0.363469	0.003681	0.391145	three

Calculating Average of DF Data

In [ ]:

```
X= df["A"]+df["B"]+df["C"]+df["D"]
df["Average"]= X/4
#df.drop('Mean', inplace=True, axis=1)
df
```

Out[ ]:

	A	B	C	D	E	Average
2022-01-01	-1.015411	-0.685878	0.146540	0.488460	one	-0.266572
2022-01-02	0.423193	-0.621133	-0.093249	0.393101	one	0.025478
2022-01-03	-0.412571	-1.436648	0.573326	0.309208	two	-0.241671
2022-01-04	0.644303	0.509191	0.565688	1.725495	three	0.861169
2022-01-05	-0.979584	0.461584	-0.665996	1.194677	four	0.002670
2022-01-06	0.159986	-0.678950	-2.291788	1.138026	three	-0.418182
2022-01-07	-0.442468	1.082353	-2.133290	0.217281	one	-0.319031
2022-01-08	-1.283584	1.408376	-0.924864	0.653886	one	-0.036547
2022-01-09	0.461194	0.498534	0.171866	1.682773	two	0.703592
2022-01-10	-0.750560	-0.245707	-0.004486	1.070353	three	0.017400
2022-01-11	-1.386227	0.161764	0.827493	-0.074207	four	-0.117795
2022-01-12	1.421998	-0.280355	-0.012034	-0.098590	three	0.257755
2022-01-13	1.554924	-0.268234	-0.429994	0.923679	one	0.445094
2022-01-14	-1.837905	-1.461330	0.402138	-1.256550	one	-1.038412
2022-01-15	0.088351	0.008429	1.079971	0.541435	two	0.429546
2022-01-16	-1.778927	1.583108	-0.221253	-0.611673	three	-0.257186
2022-01-17	2.220360	-0.132794	1.143537	0.639487	four	0.967647
2022-01-18	0.846034	0.807254	-1.586870	0.404676	three	0.117773
2022-01-19	0.152072	0.197637	-0.782905	-1.716974	four	-0.537542
2022-01-20	-0.378237	0.363469	0.003681	0.391145	three	0.095015