Different Date time setups

Daily

In [4]:
```
df.head()
```

Out[4]:

	Date	Positive	Hospitalized	Death
0	17-03-2020	10021	325	124
1	18-03-2020	13385	416	155
2	19-03-2020	16085	617	203
3	20-03-2020	24197	1042	273
4	21-03-2020	31013	1492	335

In [5]:
```
df.tail()
```

Out[5]:

	Date	Positive	Hospitalized	Death
260	02-12-2020	13711156	100322	264522
261	03-12-2020	13921360	100755	267228
262	04-12-2020	14146191	101276	269791
263	05-12-2020	14357264	101190	272236
264	06-12-2020	14534035	101487	273374

Data is collecetd for the period of 17th March 2020 to 06th December 2020. Data is collected on daily basis for Positive cases, Hospitalized and Death count

converting data into time series

In [6]:
```
date = pd.date_range(start='3/17/2020', end='12/6/2020', freq='D')
date
```

Out[6]:
```
DatetimeIndex(['2020-03-17', '2020-03-18', '2020-03-19', '2020-03-20',
               '2020-03-21', '2020-03-22', '2020-03-23', '2020-03-24',
               '2020-03-25', '2020-03-26',
               ...
               '2020-11-27', '2020-11-28', '2020-11-29', '2020-11-30',
               '2020-12-01', '2020-12-02', '2020-12-03', '2020-12-04',
               '2020-12-05', '2020-12-06'],
              dtype='datetime64[ns]', length=265, freq='D')
```

In [7]:
```
df['Time_Stamp'] = pd.DataFrame(date)
```

In [8]:
```
df=df.set_index('Time_Stamp')
```

In [9]:
```
df.head()
```

Out[9]:

Month

read AirTemp data and convert it to time series

In [2]:
```
df= pd.read_csv('AirTemp.csv',parse_dates=True)
```

In [3]:
```
df.head()
```

Out[3]:

	Year	Month	AvgTemp
0	1920	Jan	40.6
1	1920	Feb	40.8
2	1920	Mar	44.4
3	1920	Apr	46.7
4	1920	May	54.1

In [4]:
```
df.tail()
```

Out[4]:

	Year	Month	AvgTemp
235	1939	Aug	61.8
236	1939	Sep	58.2
237	1939	Oct	46.7
238	1939	Nov	46.6
239	1939	Dec	37.8

In [5]:
```
date_rng=  pd.date_range(start='1/1/1920', end='31/12/1939', freq='M')
df['TimeIndex'] = pd.DataFrame(date_rng)
df.drop(['Year','Month'],axis=1,inplace=True)
df.set_index('TimeIndex',inplace=True)
df.head()
```

Out[5]:

TimeIndex	AvgTemp
1920-01-31	40.6
1920-02-29	40.8
1920-03-31	44.4
1920-04-30	46.7
1920-05-31	54.1

Quarter

Read retail turnover data

In [4]:
```
df = pd.read_csv('RetailTurnover.csv')
df.head()
```

Out[4]:

	Year	Quarter	Turnover
0	1982	Q3	13423.2
1	1982	Q4	13128.8
2	1983	Q1	15398.8
3	1983	Q2	12964.2
4	1983	Q3	13133.5

In [5]:
```
df.tail()
```

Out[5]:

	Year	Quarter	Turnover
34	1991	Q1	17115.2
35	1991	Q2	14284.9
36	1991	Q3	14558.8
37	1991	Q4	14914.3
38	1992	Q1	17342.3

In [6]:
```
quarters= pd.date_range(start='1/7/1982', end='31/3/1992', freq='Q')
```

In [7]:
```
df['Time_Stamp']=pd.DataFrame(quarters)
df.drop(['Year','Quarter'],axis=1,inplace=True)
df=df.set_index('Time_Stamp')
```

Year Month

In [5]:
```
#Check data types
df1.dtypes
```

Out[5]:
```
Year-Month    object
Pax            int64
dtype: object
```

Year-Month column is not seen as a date object

In [6]:
```
#We are providing inputs to tell pandas that we are trying to work with time series.
df1 = pd.read_csv('AirPassenger-E.csv', parse_dates = ['Year-Month'])
```

In [7]:
```
df1.dtypes
```

Out[7]:
```
Year-Month    datetime64[ns]
Pax                    int64
dtype: object
```

Now the time series reference is appropriately identified.

In [8]:
```
#It is recommended that we make our time series reference as the index
df1 = pd.read_csv('AirPassenger.csv', parse_dates = ['Year-Month'], index_col = 'Year-Month')
```

In [9]:
```
df1.head()
```

Out[9]:

Year-Month	Pax
1949-01-01	112
1949-02-01	118
1949-03-01	132
1949-04-01	129
1949-05-01	121

In [10]:
```
#We can conveniently do slicing i.e. obtain data for a specific time period.
df1['1951-04-01':'1952-03-01']
```

Out[10]:

Year-Month	Pax
1951-04-01	163
1951-05-01	172
1951-06-01	178
1951-07-01	199

One More Year Month

In [5]:
```
df.tail()
```

Out[5]:

	Month	heater	ice cream
175	2018-08	21	66
176	2018-09	24	49
177	2018-10	39	39
178	2018-11	53	34
179	2018-12	48	36

convert data into time series

In [6]:
```
date = pd.date_range(start='1/1/2004', end='1/1/2019', freq='M')
date
```

Out[6]:
```
DatetimeIndex(['2004-01-31', '2004-02-29', '2004-03-31', '2004-04-30',
               '2004-05-31', '2004-06-30', '2004-07-31', '2004-08-31',
               '2004-09-30', '2004-10-31',
               ...
               '2018-03-31', '2018-04-30', '2018-05-31', '2018-06-30',
               '2018-07-31', '2018-08-31', '2018-09-30', '2018-10-31',
               '2018-11-30', '2018-12-31'],
              dtype='datetime64[ns]', length=180, freq='M')
```

In [7]:
```
df['Time_Stamp'] = pd.DataFrame(date)
df=df.set_index('Time_Stamp')
```

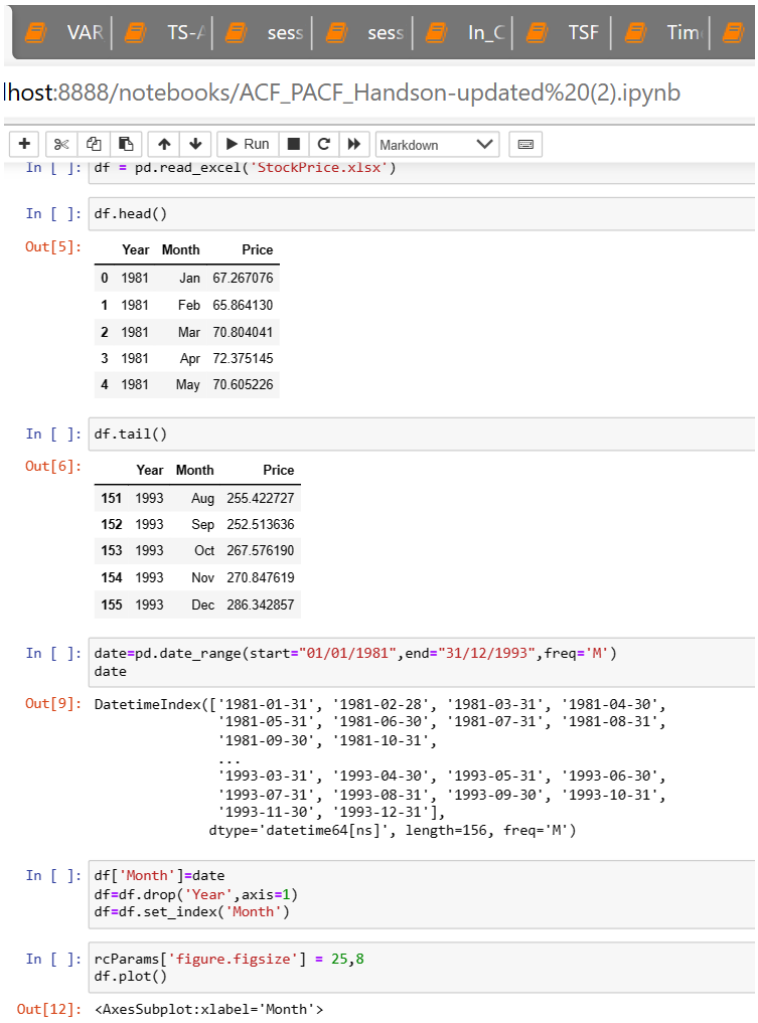
In [8]:
```
df.head()
```

Out[8]:

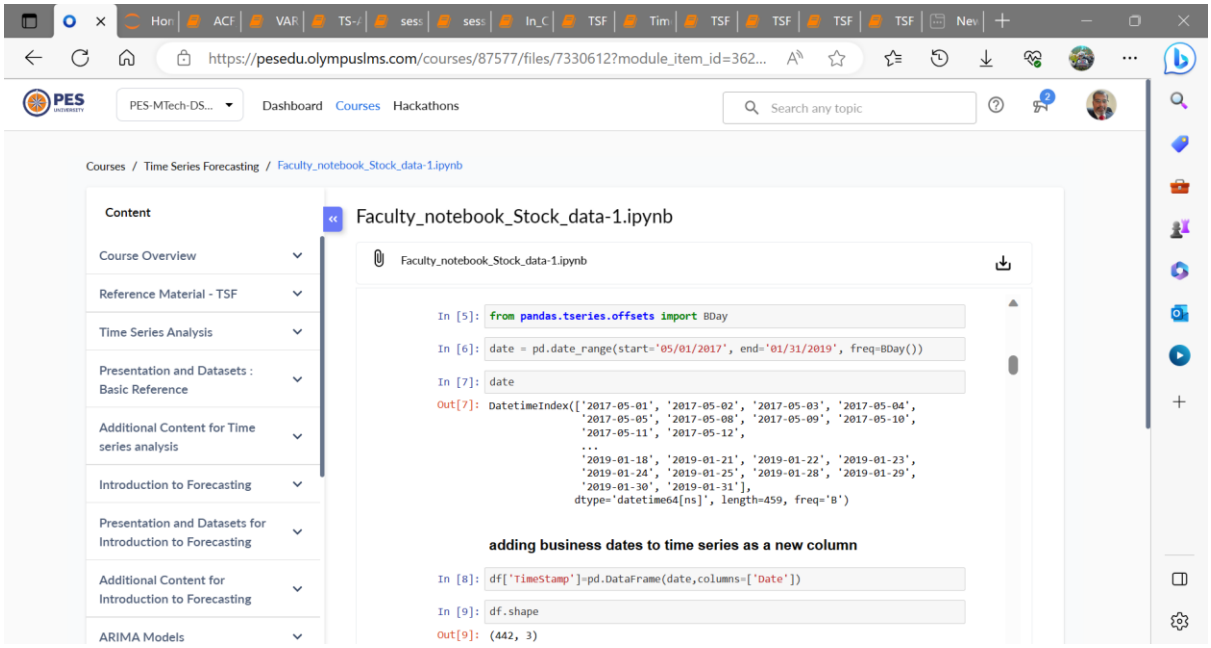
Time_Stamp	Month	heater	ice cream
2004-01-31	2004-01	27	13
2004-02-29	2004-02	18	15
2004-03-31	2004-03	14	16
2004-04-30	2004-04	13	19
2004-05-31	2004-05	13	21

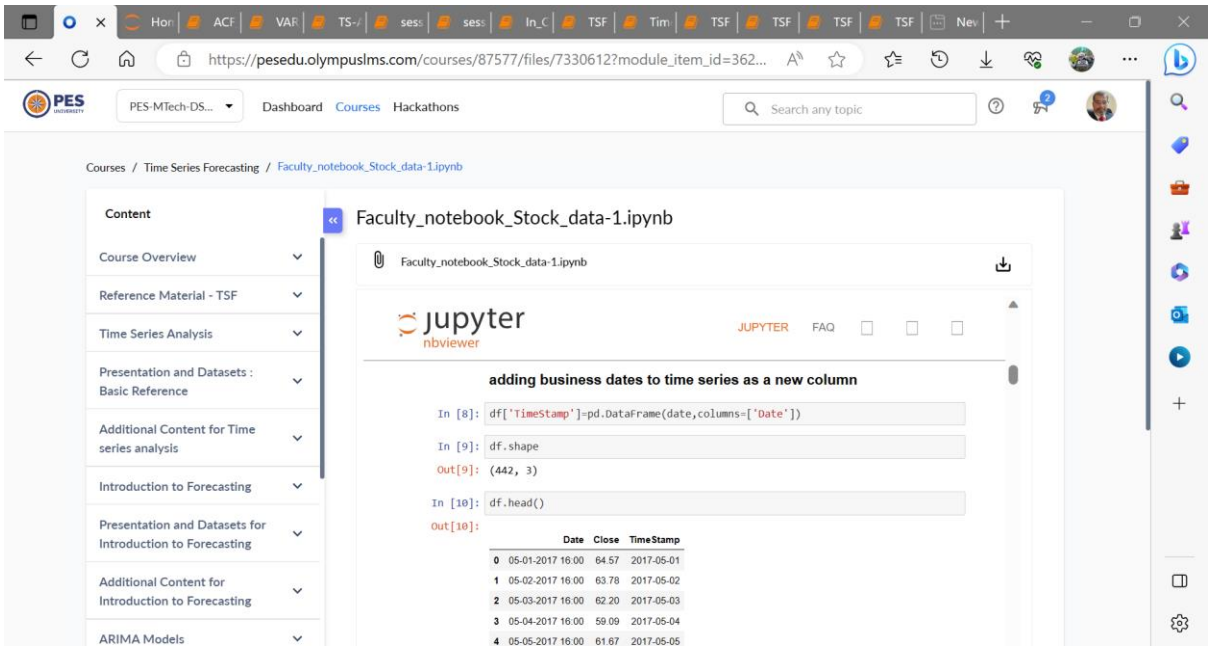
In [9]:
```
df.drop('Month',axis=1,inplace=True)
```

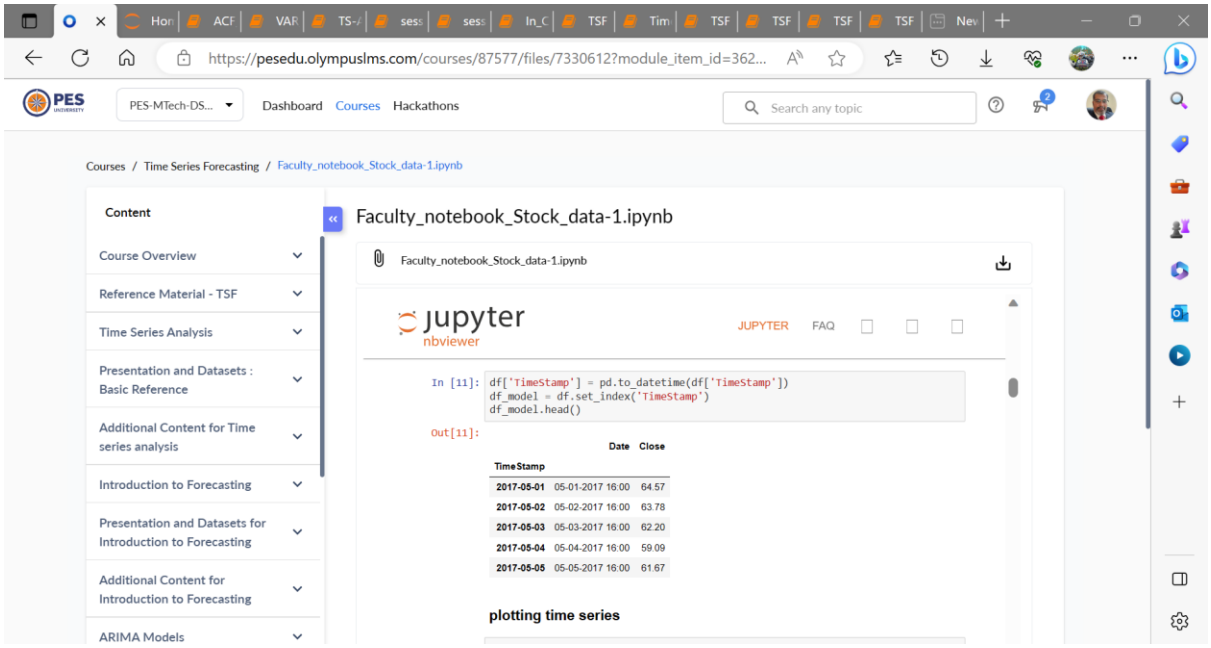
Year and Month given – Converting to Year Month (Monthly)



Business Day


https://pesedu.olympuslms.com/courses/87577/files/7330612?module_item_id=362...
PES-MTech-DS...
Dashboard
Courses
Hackathons
Search any topic
Courses / Time Series Forecasting / Faculty_notebook_Stock_data-1.ipynb
Content
Course Overview
Reference Material - TSF
Time Series Analysis
Presentation and Datasets : Basic Reference
Additional Content for Time series analysis
Introduction to Forecasting
Presentation and Datasets for Introduction to Forecasting
Additional Content for Introduction to Forecasting
ARIMA Models
Faculty_notebook_Stock_data-1.ipynb
Faculty_notebook_Stock_data-1.ipynb
In [5]: from pandas.tseries.offsets import BDay
In [6]: date = pd.date_range(start='05/01/2017', end='01/31/2019', freq=BDay())
In [7]: date
Out[7]: DatetimeIndex(['2017-05-01', '2017-05-02', '2017-05-03', '2017-05-04',
'2017-05-05', '2017-05-08', '2017-05-09', '2017-05-10',
'2017-05-11', '2017-05-12',
...
'2019-01-18', '2019-01-21', '2019-01-22', '2019-01-23',
'2019-01-24', '2019-01-25', '2019-01-28', '2019-01-29',
'2019-01-30', '2019-01-31'],
dtype='datetime64[ns]', length=459, freq='B')
adding business dates to time series as a new column
In [8]: df['TimeStamp']=pd.DataFrame(date,columns=['Date'])
In [9]: df.shape
Out[9]: (442, 3)


https://pesedu.olympuslms.com/courses/87577/files/7330612?module_item_id=362...
PES-MTech-DS...
Dashboard
Courses
Hackathons
Search any topic
Courses / Time Series Forecasting / Faculty_notebook_Stock_data-1.ipynb
Content
Course Overview
Reference Material - TSF
Time Series Analysis
Presentation and Datasets : Basic Reference
Additional Content for Time series analysis
Introduction to Forecasting
Presentation and Datasets for Introduction to Forecasting
Additional Content for Introduction to Forecasting
ARIMA Models
Faculty_notebook_Stock_data-1.ipynb
Faculty_notebook_Stock_data-1.ipynb
jupyter
nbviewer
JUPYTER
FAQ
adding business dates to time series as a new column
In [8]: df['TimeStamp']=pd.DataFrame(date,columns=['Date'])
In [9]: df.shape
Out[9]: (442, 3)
In [10]: df.head()
Out[10]:
Date Close TimeStamp
0 05-01-2017 16:00 64.57 2017-05-01
1 05-02-2017 16:00 63.78 2017-05-02
2 05-03-2017 16:00 62.20 2017-05-03
3 05-04-2017 16:00 59.09 2017-05-04
4 05-05-2017 16:00 61.67 2017-05-05


https://pesedu.olympuslms.com/courses/87577/files/7330612?module_item_id=362...
PES-MTech-DS...
Dashboard
Courses
Hackathons
Search any topic
Courses / Time Series Forecasting / Faculty_notebook_Stock_data-1.ipynb
Content
Course Overview
Reference Material - TSF
Time Series Analysis
Presentation and Datasets : Basic Reference
Additional Content for Time series analysis
Introduction to Forecasting
Presentation and Datasets for Introduction to Forecasting
Additional Content for Introduction to Forecasting
ARIMA Models
Faculty_notebook_Stock_data-1.ipynb
Faculty_notebook_Stock_data-1.ipynb
jupyter
nbviewer
JUPYTER
FAQ
In [11]:
df['TimeStamp'] = pd.to_datetime(df['TimeStamp'])
df_model = df.set_index('TimeStamp')
df_model.head()
Out[11]:
Date
Close
TimeStamp
2017-05-01 05-01-2017 16:00 64.57
2017-05-02 05-02-2017 16:00 63.78
2017-05-03 05-03-2017 16:00 62.20
2017-05-04 05-04-2017 16:00 59.09
2017-05-05 05-05-2017 16:00 61.67
plotting time series