The Engineering World #DataScience 9 & 10

May 31, 2018

AKKAL BAHADUR BIST
DATA SCIENTIST AT
KATHMANDU INSTITUTE OF APPLIED SCIENCES (KIAS)
Center for Conservation Biology (CCB)

1 CREATING LABELS AND ANNOTATIONS

1.0.1 Creating labels and annotations

```
In [1]: import numpy as np
        from numpy.random import randn
        import pandas as pd
        from pandas import Series, DataFrame
        import matplotlib.pyplot as plt
        from matplotlib import rcParams
        import seaborn as sb
```

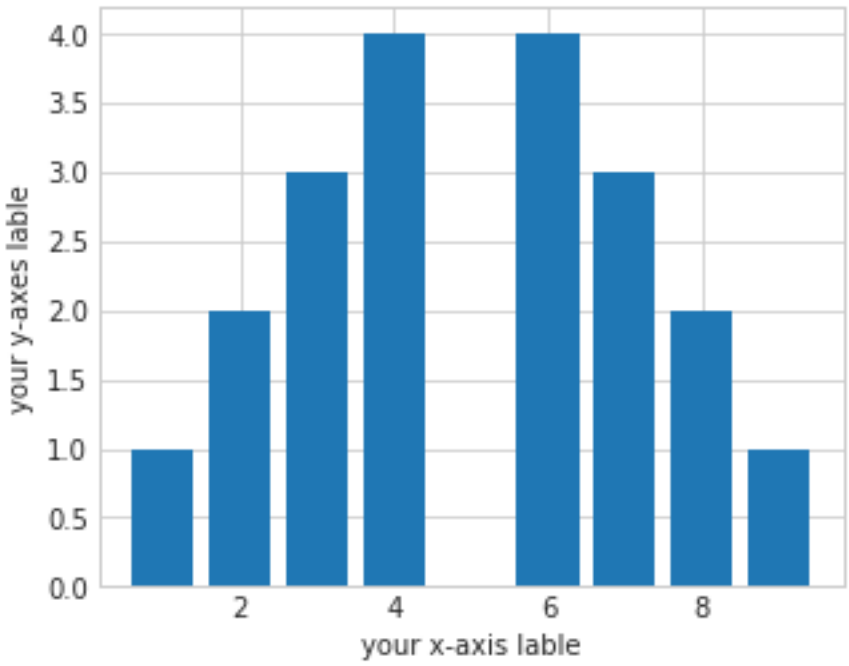
```
In [2]: %matplotlib inline
        rcParams ['figure.figsize'] = 5,4
        sb.set_style ('whitegrid')
```

1.0.2 Labeling plot features

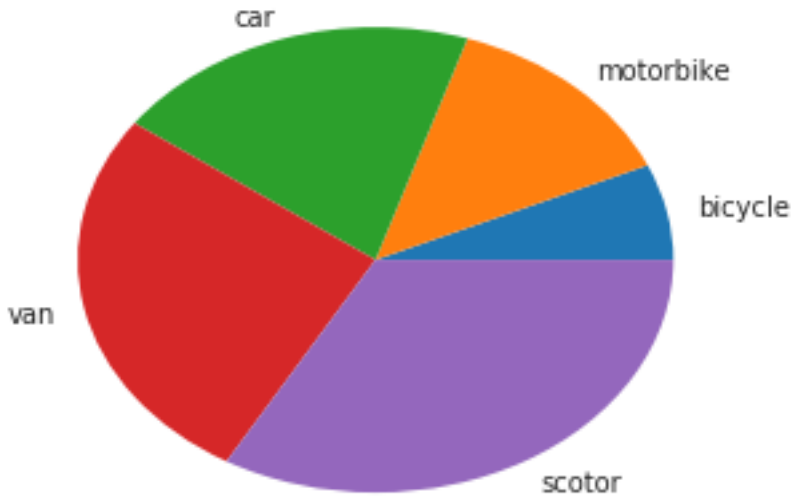
1.0.3 The functional method

```
In [3]: x = range(1,10)
        y = [1,2,3,4,0,4,3,2,1]

        plt.bar(x,y)
        plt.xlabel('your x-axis lable') #define axis lable in a graph side
        plt.ylabel('your y-axes lable')
```

```
Out[3]: Text(0,0.5,'your y-axes lable')
```



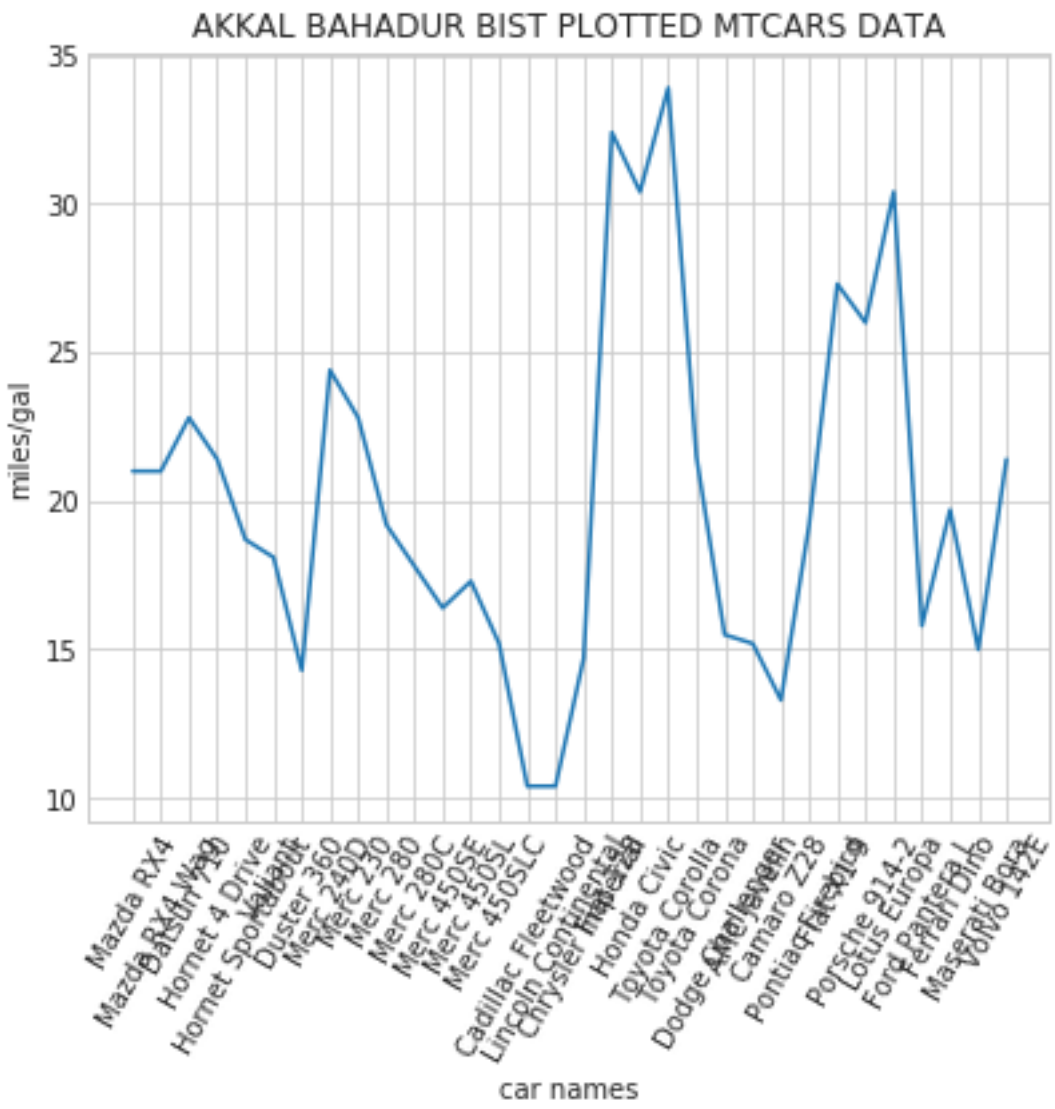

```
In [4]: z = [1,2,3,4,5]
        veh_type = ['bicycle', 'motorbike', 'car', 'van', 'scotor']
        plt.pie(z, labels = veh_type)
        plt.show()
```



1.0.4 The object-oriented method

In [5]:
```
address = 'mtcars.csv'
cars = pd.read_csv(address)
cars.columns = ['car_names', 'mpg', 'cyl', 'disp', 'hp', 'drat', 'wt', 'qsec', 'vs', 'am
mpg = cars.mpg

fig = plt.figure()

ax = fig.add_axes([.1,.1,1,1])
mpg.plot()

ax.set_xticks(range(32))
ax.set_xticklabels(cars.car_names, rotation = 60, fontsize = 'medium') #label name defin
ax.set_title('AKKAL BAHADUR BIST PLOTTED MTCARS DATA') #plot title name
ax.set_xlabel('car names') #xlabel name
ax.set_ylabel('miles/gal') #ylabel name
```

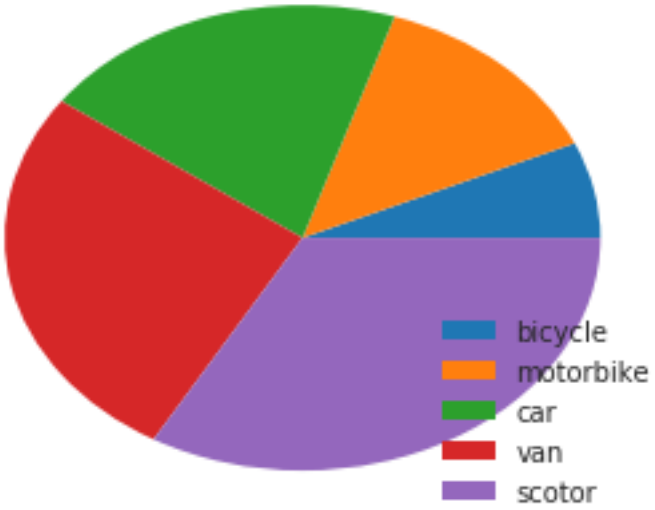
Out[5]:
```
Text(0,0.5,'miles/gal')
```



1.0.5 Adding a lagend to yur plot

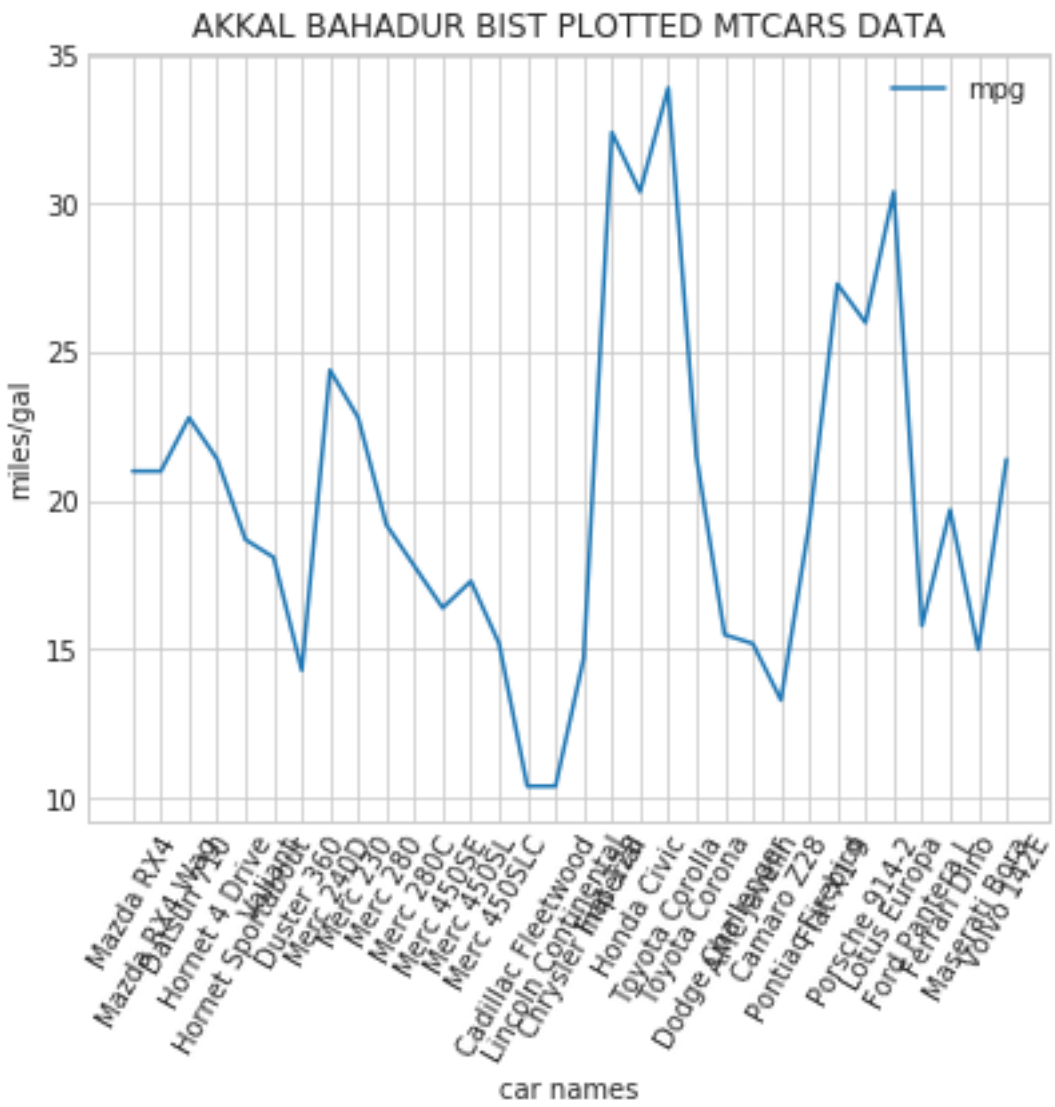
1.0.6 The functional method

```
In [6]: plt.pie(z)
        plt.legend(veh_type, loc = 'best') #lagend in pie plot
        plt.show()
```



1.0.7 The object-oriented method

```
In [7]: fig = plt.figure()
        ax = fig.add_axes([.1,.1,1,1])
        mpg.plot()

        ax.set_xticks(range(32))

        ax.set_xticklabels(cars.car_names, rotation = 60, fontsize = 'medium')
        ax.set_title('AKKAL BAHADUR BIST PLOTTED MTCARS DATA')
        ax.set_xlabel('car names')
        ax.set_ylabel('miles/gal')
        ax.legend (loc = 'best') #define legend in plot

Out[7]: <matplotlib.legend.Legend at 0x7f82142a2320>
```



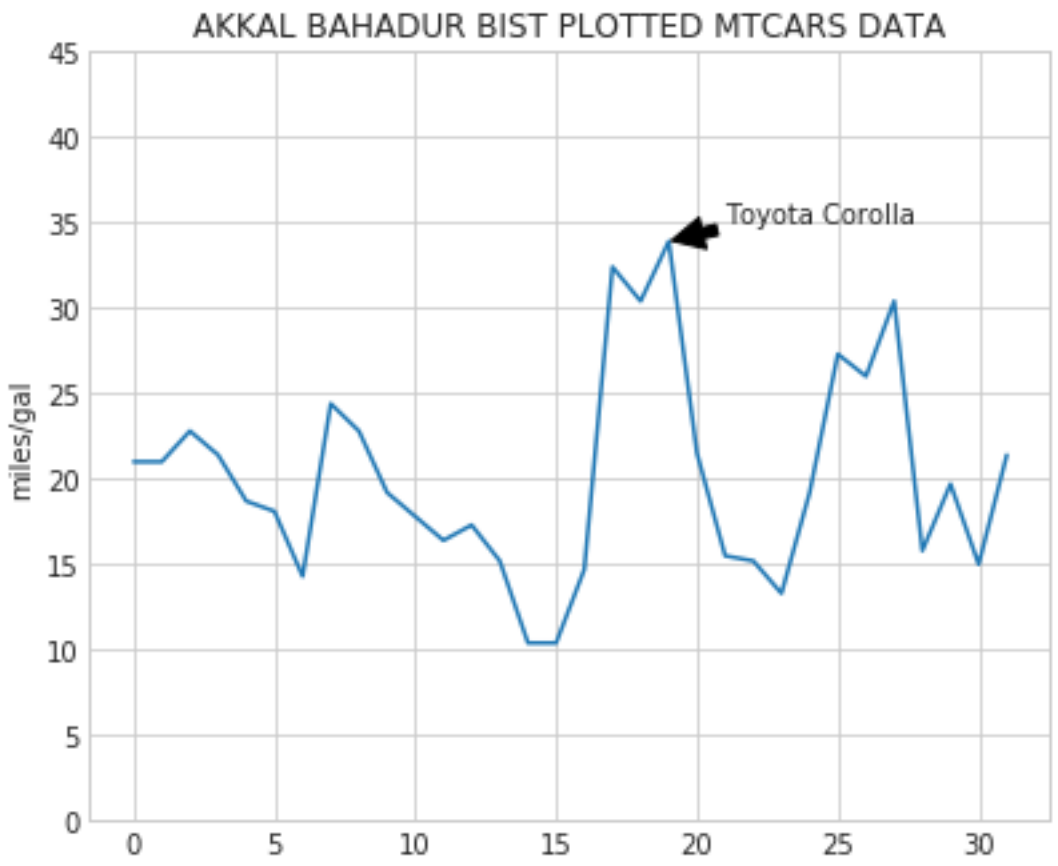
1.0.8 Annotating your plot

```
In [8]: mpg.max() #show data status in user required
```

```
Out[8]: 33.9
```

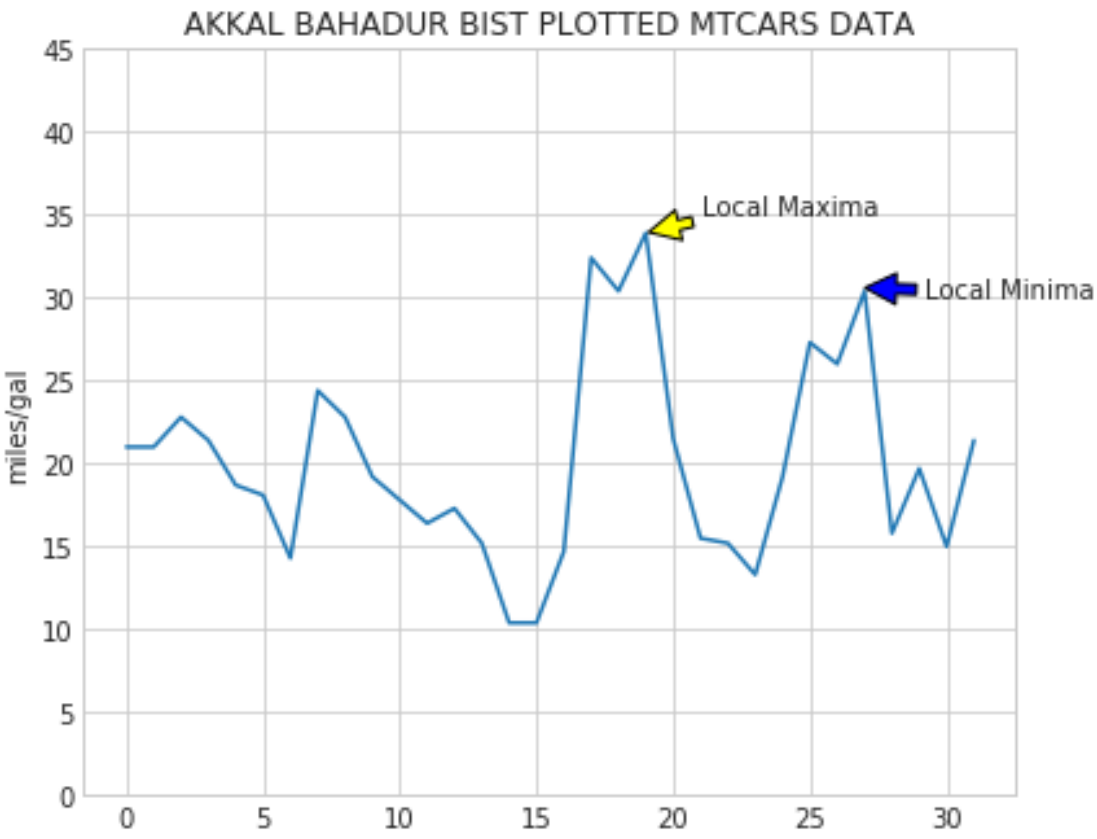
```
In [9]: fig = plt.figure()
        ax = fig.add_axes([.1,.1,1,1])
        mpg.plot()

        ax.set_title('AKKAL BAHADUR BIST PLOTTED MTCARS DATA')
        ax.set_ylabel('miles/gal')
        ax.set_ylim([0,45])
        ax.annotate('Toyota Corolla', xy = (19,33.9), xytext = (21,35), arrowprops = dict(faceco
```

Out[9]: Text(21,35,'Toyota Corolla')



In [10]:
```
 fig = plt.figure()
ax = fig.add_axes([.1,.1,1,1])
mpg.plot()

ax.set_title('AKKAL BAHADUR BIST PLOTTED MTCARS DATA')
ax.set_ylabel('miles/gal')
ax.set_ylim([0,45])
ax.annotate('Local Maxima', xy = (19,33.9), xytext = (21,35), arrowprops = dict(facecol
ax.annotate('Local Minima', xy = (26.9,30.6), xytext = (29.2,30), arrowprops = dict(fac
```

Out[10]: Text(29.2,30,'Local Minima')



2 TIME SERIES DATA VISUALIZATION

2.0.1 Creating visualization from a time series data

2.0.2 The simplest time series plot

```
In [11]: address = 'DATA.csv'
         df = pd.read_csv(address, index_col = 'Year', parse_dates = True)
         df.head
```

```
Out[11]: <bound method NDFrame.head of                Wildlife       WT    PE  NU  RED arrest_d
         Year
         2011-01-01         Tiger   15.000   0.0   0   EN          Parsa   Winter
         2011-01-01    Rhinoceros    1.000   0.0   0   VU        Syanjha   Winter
         2011-01-01       Leopard    0.000   1.0   0   VU       Lalitpur  Spring
         2011-01-01       Leopard    0.000   1.0   0   VU          Kavre  Spring
         2011-01-01         Tiger    8.800   1.0   0   EN       Makwanpur  Spring
         2011-01-01    Rhinoceros    0.000  12.0   0   VU       Kathmandu  Spring
         2011-01-01    Rhinoceros    1.000   0.0   0   VU       Lalitpur   Spring
         2011-01-01    Rhinoceros    1.000   0.0   0   VU       Kathmandu  Spring
         2011-01-01    Rhinoceros    1.000   0.0   0   VU       Kathmandu  Summer
```

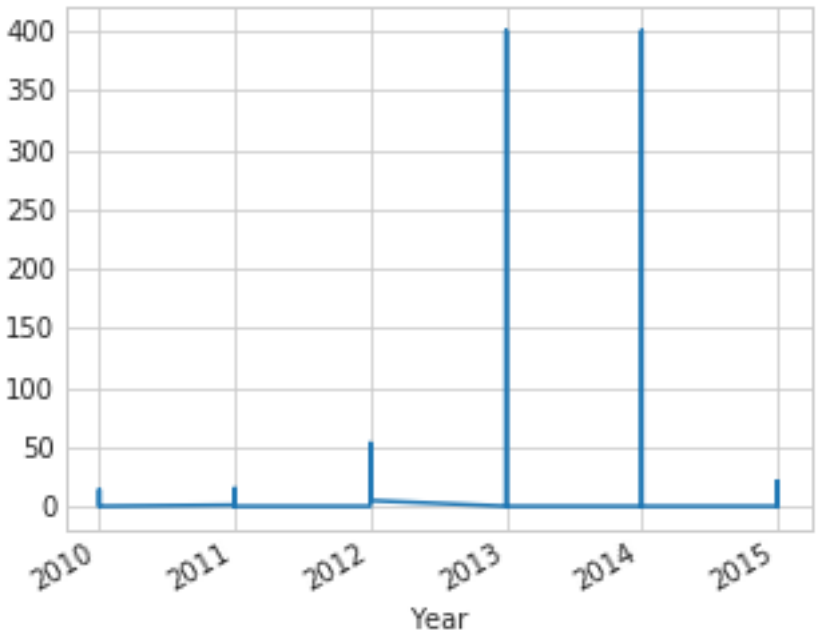
2011-01-01	Leopard	4.500	1.0	0	VU	Tanahun	Summer
2011-01-01	Red panda	0.000	2.0	0	EN	Kathmandu	Autumn
2012-01-01	Bear	0.073	0.0	0	VU	Kathmandu	Winter
2012-01-01	Leopard	0.000	2.0	0	VU	Kathmandu	Winter
2012-01-01	Leopard	0.000	1.0	0	VU	Kathmandu	Winter
2012-01-01	Bear	0.000	2.0	0	VU	Kathmandu	Winter
2012-01-01	Musk Deer	0.000	1.0	0	EN	Kathmandu	Spring
2012-01-01	Leopard	0.000	2.0	0	VU	Kavre	Spring
2012-01-01	Rhinoceros	1.000	0.0	0	VU	Kathmandu	Spring
2012-01-01	Rhinoceros	1.000	0.0	0	VU	Kaski	Spring
2012-01-01	elephant	0.000	2.0	0	EN	Kailali	Summer
2012-01-01	Bear	0.185	0.0	0	VU	Kathmandu	Summer
2012-01-01	Tiger	10.000	0.0	0	EN	Makwanpur	Summer
2012-01-01	Bear	2.000	0.0	0	VU	Kathmandu	Summer
2012-01-01	Leopard	0.000	2.0	0	VU	Lalitpur	Summer
2012-01-01	Tiger	0.000	1.0	0	EN	Kathmandu	Autumn
2012-01-01	Red panda	0.000	2.0	0	EN	Kathmandu	Autumn
2013-01-01	Leopard	0.000	1.0	0	VU	Sunsari	Winter
2013-01-01	Tiger	0.000	12.0	0	EN	Kathmandu	Spring
2013-01-01	Pangolin	0.000	1.0	0	EN	Kathmandu	Winter
2013-01-01	Red panda	0.000	1.0	0	EN	Nuwakot	Winter
...	...	...	...	..	...	...	...
2014-01-01	Elephant	21.000	0.0	0	EN	Lalitpur	FALSE
2013-01-01	Rhinoceros	0.000	1.0	0	VU	Mahottari	Spring
2015-01-01	Red Panda	0.000	2.0	0	EN	Bhaktapur	FALSE
2013-01-01	Pangolin	9.000	9.0	0	EN	Sindhupalchok	Spring
2013-01-01	Pangolin	80.000	80.0	0	EN	Sindhupalchok	Spring
2013-01-01	Tiger	1.240	0.0	0	EN	Bardiya	Spring
2013-01-01	Tiger	400.000	8.0	0	EN	Nuwakot	Spring
2013-01-01	Pangolin	40.000	40.0	0	EN	Darchula	Spring
2013-01-01	Red Panda	0.000	3.0	0	EN	Makwanpur	Summer
2013-01-01	Sambar deer	1.000	0.0	0	VU	Sindhupalchok	Summer
2013-01-01	Pangolin	48.000	48.0	0	EN	Bhaktapur	Summer
2013-01-01	Tiger	0.000	3.0	0	EN	Dang	Summer
2013-01-01	Leopard	0.000	3.0	0	VU	Dang	Summer
2013-01-01	Pangolin	40.000	40.0	0	EN	Dang	Summer
2013-01-01	Spotted deer	3.000	0.0	0	LC	Banke	Summer
2013-01-01	Python	0.000	1.0	0	VU	Kathmandu	Summer
2010-01-01	Pangolin	14.000	14.0	0	EN	Kathmandu	FALSE
2015-01-01	Leopard	2.445	1.0	0	VU	Surkhet	Spring
2010-01-01	Pangolin	10.000	10.0	0	EN	Sindhupalchok	FALSE
2011-01-01	Bear	0.000	5.0	0	VU	Sankhuwasabha	Summer
2011-01-01	Red Panda	0.000	2.0	0	EN	Kathmandu	Summer
2012-01-01	Bear	0.000	5.0	0	VU	Sankhuwasabha	Autumn
2011-01-01	Tiger	0.000	1.0	0	EN	Kathmandu	Spring
2011-01-01	Leopard	0.000	2.0	0	VU	Dolakha	Summer
2011-01-01	Tiger	7.500	0.0	0	EN	Bara	Autumn
2012-01-01	Rhinoceros	1.000	0.0	0	VU	Kathmandu	Winter

```
2012-01-01      Pangolin    4.000   4.0   0   EN            Ilam  Spring
2012-01-01          Bear    0.182   0.0   0   VU       Kathmandu  Summer
2011-01-01     Red Panda    0.000   1.0   0   EN       Kathmandu  Summer
2012-01-01    Rhinoceros    0.000   3.0   0   VU         Chitwan  Summer

             Volume
Year
2011-01-01   15.000
2011-01-01    1.000
2011-01-01    1.000
2011-01-01    1.000
2011-01-01    9.800
2011-01-01   12.000
2011-01-01    1.000
2011-01-01    1.000
2011-01-01    1.000
2011-01-01    5.500
2011-01-01    2.000
2012-01-01    0.073
2012-01-01    2.000
2012-01-01    1.000
2012-01-01    2.000
2012-01-01    1.000
2012-01-01    2.000
2012-01-01    1.000
2012-01-01    1.000
2012-01-01    2.000
2012-01-01    0.185
2012-01-01   10.000
2012-01-01    2.000
2012-01-01    2.000
2012-01-01    1.000
2012-01-01    2.000
2013-01-01    1.000
2013-01-01   12.000
2013-01-01    1.000
2013-01-01    1.000
...              ...
2014-01-01   21.000
2013-01-01    1.000
2015-01-01    2.000
2013-01-01   18.000
2013-01-01  160.000
2013-01-01    1.240
2013-01-01  408.000
2013-01-01   80.000
2013-01-01    3.000
2013-01-01    1.000
```

```
2013-01-01   96.000
2013-01-01    3.000
2013-01-01    3.000
2013-01-01   80.000
2013-01-01    3.000
2013-01-01    1.000
2010-01-01   28.000
2015-01-01    3.445
2010-01-01   20.000
2011-01-01    5.000
2011-01-01    2.000
2012-01-01    5.000
2011-01-01    1.000
2011-01-01    2.000
2011-01-01    7.500
2012-01-01    1.000
2012-01-01    8.000
2012-01-01    0.182
2011-01-01    1.000
2012-01-01    3.000

[444 rows x 8 columns]>
```

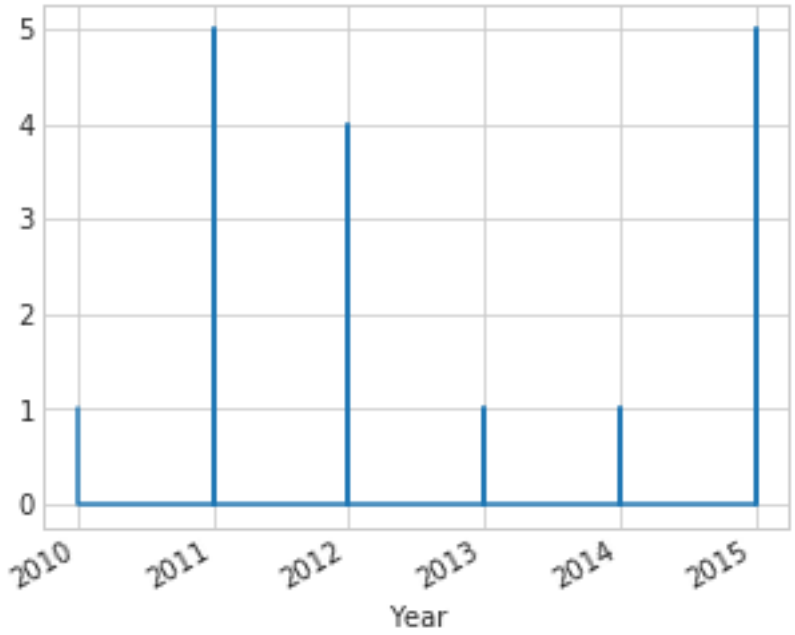
In [12]:
```
df['WT'].plot()
```

Out[12]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8214bb3b70>
```



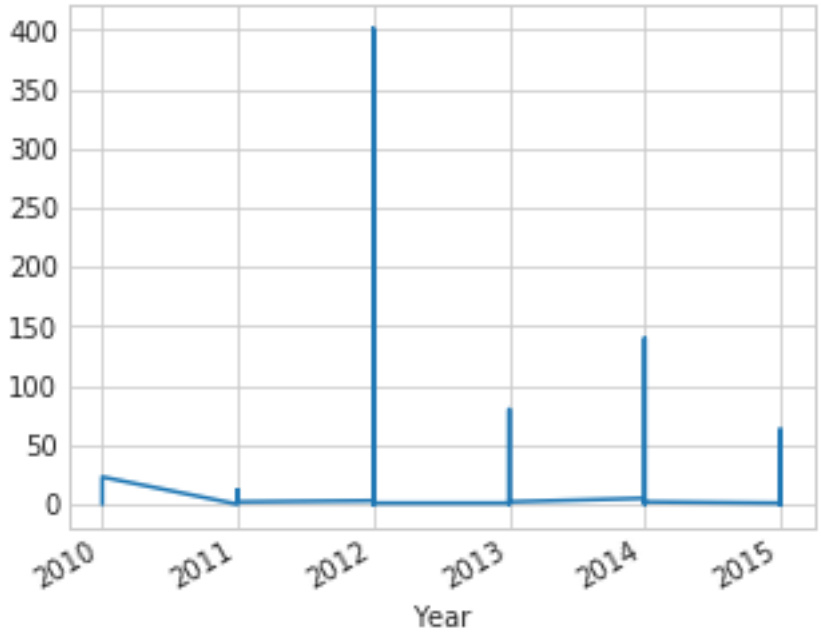
```
In [13]: df['NU'].plot()
```

```
Out[13]: <matplotlib.axes._subplots.AxesSubplot at 0x7f8214baca20>
```



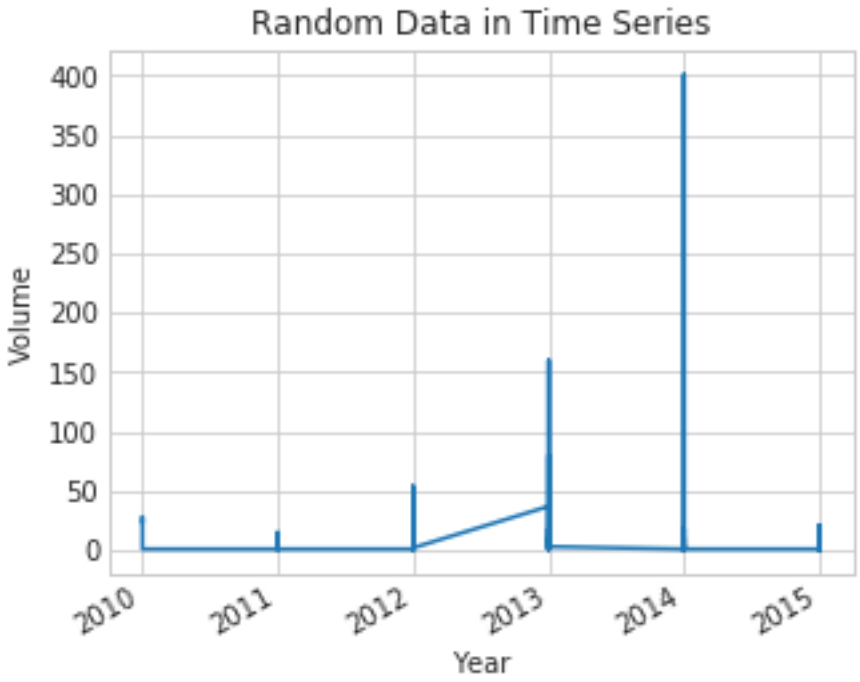

```
In [14]: df['PE'].plot()
```

```
Out[14]: <matplotlib.axes._subplots.AxesSubplot at 0x7f821410f4e0>
```



In [15]:
```
df2 = df.sample(n=100, random_state=25,axis=0)
plt.xlabel('Year')
plt.ylabel('Volume')
plt.title('Random Data in Time Series')
df2['Volume'].plot()
```

Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x7f8214b6eac8>



In [ ]: