

```
In [1]: import pandas as pd
```

```
In [8]: s1 = pd.Series([1,12,43,4,75])
```

```
In [9]: s1
```

```
Out[9]: 0      1  
        1     12  
        2     43  
        3      4  
        4     75  
        dtype: int64
```

```
In [10]: type(s1)
```

```
Out[10]: pandas.core.series.Series
```

```
In [11]: #changing index from numarical to alphabetic
```

```
In [12]: import pandas as pd
```

```
In [14]: s2 = pd.Series([22,33,55,66])  
s2
```

```
Out[14]: 0      22  
        1      33  
        2      55  
        3      66  
        dtype: int64
```

```
In [21]: s2 = pd.Series([22,33,55,66],index=['a','b','c','d'])  
s2
```

```
Out[21]: a      22
```

```
b    33
c    55
d    66
dtype: int64
```

```
In [22]: #series with dictionary
```

```
In [24]: s1= pd.Series({'a':11,'b':22,'c':33})
s1
```

```
Out[24]: a    11
b    22
c    33
dtype: int64
```

```
In [27]: s2= pd.Series({'a':32,'b':43,'c': 54},index = ['c','b','a','z'])
s2
```

```
Out[27]: c    54.0
b    43.0
a    32.0
z     NaN
dtype: float64
```

```
In [29]: import pandas as pd
sq = [2,4,6,8,9]
sq
```

```
Out[29]: [2, 4, 6, 8, 9]
```

```
In [32]: sq1 = pd.Series(sq)
sq1
```

```
Out[32]: 0    2
1    4
2    6
3    8
```

```
4    9
dtype: int64
```

```
In [34]: sql[4]
```

```
Out[34]: 9
```

```
In [36]: sql[2]
```

```
Out[36]: 6
```

```
In [38]: sql[-3:]
```

```
Out[38]: 2    6
         3    8
         4    9
dtype: int64
```

```
In [40]: sql[1:]
```

```
Out[40]: 1    4
         2    6
         3    8
         4    9
dtype: int64
```

```
In [41]: import pandas as pd
```

```
In [43]: l1 = [1,2,4,5,6,7,8]
         l2 = pd.Series(l1)
         l2
```

```
Out[43]: 0    1
         1    2
         2    4
         3    5
         4    6
         5    7
```

```
6      8
dtype: int64
```

```
In [45]: l2+5
```

```
Out[45]: 0      6
          1      7
          2      9
          3     10
          4     11
          5     12
          6     13
dtype: int64
```

```
In [56]: s1= [1,3,5,7]
          s2 = [2,4,6,8]
          s1= pd.Series(s1)
          s1
```

```
Out[56]: 0      1
          1      3
          2      5
          3      7
dtype: int64
```

```
In [51]: s2= pd.Series(s2)
          s2
```

```
Out[51]: 0      2
          1      4
          2      6
          3      8
dtype: int64
```

```
In [57]: s1+s2
```

```
Out[57]: 0      3
          1      7
```

```
2    11
3    15
dtype: int64
```

```
In [59]: 11
```

```
Out[59]: [1, 2, 4, 5, 6, 7, 8]
```

```
In [62]: 12-100 #can be done all operations -,+/,*
```

```
Out[62]: 0    -99
1    -98
2    -96
3    -95
4    -94
5    -93
6    -92
dtype: int64
```

```
In [64]: import pandas as pd
```

```
In [76]: pd.DataFrame({'Name': ['Anne', 'Bob', 'Matt'], 'Marks': [20, 30, 40]})
```

```
Out[76]:
```

	Name	Marks
0	Anne	20
1	Bob	30
2	Matt	40

```
In [77]: import pandas as pd
import os
```

```
In [78]: pwd
```

```
Out[78]: 'C:\\\\Users\\amolb'
```

```
v1 = pd.read_csv('C:\Users\amolb\testfile.csv')
```

```
In [80]: v1 = pd.read_csv('C:\Users\amolb\testfile.csv')
```

```
File "<ipython-input-80-81f0bc0bfb98>", line 1
    v1 = pd.read_csv('C:\Users\amolb\testfile.csv')
          ^
```

```
SyntaxError: (unicode error) 'unicodeescape' codec can't decode bytes i
n position 2-3: truncated \UXXXXXXXX escape
```

```
In [82]: data
```

```
-----
----
NameError                                Traceback (most recent call l
ast)
<ipython-input-82-c5d84736ba45> in <module>
----> 1 data
```

```
NameError: name 'data' is not defined
```

```
In [84]: import pandas as pd
```

```
In [86]: dell=pd.read_csv(r"C:\Users\amolb\.ipynb_checkpoints\delli_weather.csv"
)
```

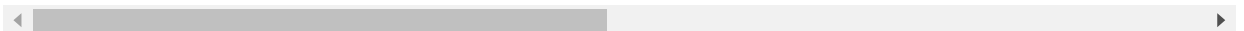
```
In [87]: dell
```

```
Out[87]:
```

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem
0	19961101-11:00	Smoke	9.0	0	0	NaN	27.0	NaN	1010.0
1	19961101-12:00	Smoke	10.0	0	0	NaN	32.0	NaN	-9999.0
2	19961101-13:00	Smoke	11.0	0	0	NaN	44.0	NaN	-9999.0

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem
3	19961101-14:00	Smoke	10.0	0	0	NaN	41.0	NaN	1010.0
4	19961101-16:00	Smoke	11.0	0	0	NaN	47.0	NaN	1011.0
...
100985	20170424-06:00	Haze	17.0	0	0	NaN	25.0	NaN	1005.0
100986	20170424-09:00	Haze	14.0	0	0	NaN	16.0	NaN	1003.0
100987	20170424-12:00	Haze	12.0	0	0	NaN	14.0	NaN	1002.0
100988	20170424-15:00	Haze	15.0	0	0	NaN	27.0	NaN	1004.0
100989	20170424-18:00	Haze	15.0	0	0	NaN	30.0	NaN	1005.0

100990 rows × 20 columns



In [89]: dell.head(4)

Out[89]:

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem	_rai
0	19961101-11:00	Smoke	9.0	0	0	NaN	27.0	NaN	1010.0	
1	19961101-12:00	Smoke	10.0	0	0	NaN	32.0	NaN	-9999.0	
2	19961101-13:00	Smoke	11.0	0	0	NaN	44.0	NaN	-9999.0	
3	19961101-14:00	Smoke	10.0	0	0	NaN	41.0	NaN	1010.0	



```
In [91]: dell.nunique() #will give u unique value in each column
```

```
Out[91]: datetime_utc    100990
         _conds          39
         _dewptm         51
         _fog            2
         _hail           2
         _heatindexm     193
         _hum            100
         _precipm        0
         _pressurem     139
         _rain           2
         _snow           2
         _tempm          50
         _thunder        2
         _tornado        2
         _vism           48
         _wdird          63
         _wdire          17
         _wgustm         22
         _windchillm     20
         _wspdm          90
dtype: int64
```

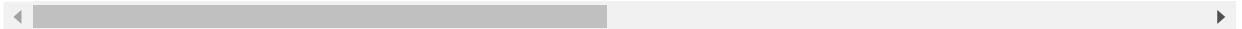
```
In [98]: dell
```

```
Out[98]:
```

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem
0	19961101-11:00	Smoke	9.0	0	0	NaN	27.0	NaN	1010.0
1	19961101-12:00	Smoke	10.0	0	0	NaN	32.0	NaN	-9999.0
2	19961101-13:00	Smoke	11.0	0	0	NaN	44.0	NaN	-9999.0
3	19961101-14:00	Smoke	10.0	0	0	NaN	41.0	NaN	1010.0

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem
4	19961101-16:00	Smoke	11.0	0	0	NaN	47.0	NaN	1011.0
...
100985	20170424-06:00	Haze	17.0	0	0	NaN	25.0	NaN	1005.0
100986	20170424-09:00	Haze	14.0	0	0	NaN	16.0	NaN	1003.0
100987	20170424-12:00	Haze	12.0	0	0	NaN	14.0	NaN	1002.0
100988	20170424-15:00	Haze	15.0	0	0	NaN	27.0	NaN	1004.0
100989	20170424-18:00	Haze	15.0	0	0	NaN	30.0	NaN	1005.0

100990 rows × 20 columns



```
In [101]: dell['datetime_utc'].unique()
```

```
Out[101]: array(['19961101-11:00', '19961101-12:00', '19961101-13:00', ...,
                '20170424-12:00', '20170424-15:00', '20170424-18:00'], dtype=object)
```

```
In [105]: #df=df.rename(columns = {'two':'new_name'})
dell=dell.rename(columns = {'_rain':'rain'})
```

```
In [106]: dell.head(2)
```

```
Out[106]:
```

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem	_rai
0	19961101-11:00	Smoke	9.0	0	0	NaN	27.0	NaN	1010.0	

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem	_rai
1	19961101-12:00	Smoke	10.0	0	0	NaN	32.0	NaN	-9999.0	



In [109]: `import pandas as pd`

In [111]: `dell=pd.read_csv(r"C:\Users\amolb\.ipynb_checkpoints\Austin_weather.csv")`

In [113]: `dell`

Out[113]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
0	2013-12-21	74	60	45	67	49	43
1	2013-12-22	56	48	39	43	36	28
2	2013-12-23	58	45	32	31	27	23
3	2013-12-24	61	46	31	36	28	21
4	2013-12-25	58	50	41	44	40	36
...
1314	2017-07-27	103	89	75	71	67	61
1315	2017-07-28	105	91	76	71	64	55
1316	2017-07-29	107	92	77	72	64	55
1317	2017-07-30	106	93	79	70	68	63

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
1318	2017-07-31	99	88	77	66	61	54

1319 rows × 21 columns



In [115]: `dell.nunique()`

```
Out[115]: Date          1319
TempHighF          74
TempAvgF           64
TempLowF           61
DewPointHighF      64
DewPointAvgF       66
DewPointLowF       73
HumidityHighPercent 58
HumidityAvgPercent  69
HumidityLowPercent  82
SeaLevelPressureHighInches 105
SeaLevelPressureAvgInches 101
SeaLevelPressureLowInches 105
VisibilityHighMiles  5
VisibilityAvgMiles   10
VisibilityLowMiles   12
WindHighMPH          22
WindAvgMPH           13
WindGustMPH          37
PrecipitationSumInches 114
Events               9
dtype: int64
```

In [120]: `dell["TempHighF"].nunique()` *#will give u count of that column*

Out[120]: 74

In [121]: `dell["TempHighF"].unique()` *#will show u all numbers of that column*

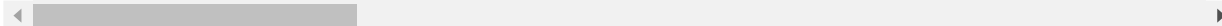
```
Out[121]: array([ 74,  56,  58,  61,  57,  60,  62,  64,  44,  55,  69,  71,  59,
                  36,  48,  53,  70,  72,  75,  67,  76,  82,  63,  40,  39,  49,
                  65,  80,  46,  32,  38,  86,  81,  83,  77,  52,  73,  47,  50,
                  68,  66,  89,  79,  85,  84,  87,  78,  88,  96,  93,  94,  91,
                  92,  90,  95,  97,  98, 100, 101,  99, 102, 104, 103,  51,  41,
                  45,  54,  42,  43,  37, 105,  34, 107, 106], dtype=int64)
```

```
In [123]: dell.head(5)
```

```
Out[123]:
```

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	Hui
0	2013-12-21	74	60	45	67	49	43	
1	2013-12-22	56	48	39	43	36	28	
2	2013-12-23	58	45	32	31	27	23	
3	2013-12-24	61	46	31	36	28	21	
4	2013-12-25	58	50	41	44	40	36	

5 rows × 21 columns



```
In [139]: #finding count
dell.Events.value_counts()
```

```
Out[139]:
```

	903
Rain	192
Rain , Thunderstorm	137
Fog , Rain , Thunderstorm	33
Fog	21
Thunderstorm	17
Fog , Rain	14
Rain , Snow	1

```
Fog , Thunderstorm      1
Name: Events, dtype: int64
```

```
In [142]: dell.Events == 'Rain' #Filtering - will show result in boolean
```

```
Out[142]: 0      False
          1      False
          2      False
          3      False
          4      False
          ...
        1314     False
        1315     False
        1316     False
        1317     False
        1318     False
Name: Events, Length: 1319, dtype: bool
```

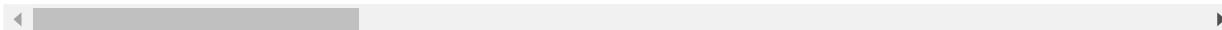
```
In [144]: dell[dell.Events == 'Rain'] #will show all the records which is havinr Rain
```

```
Out[144]:
```

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
18	2014-01-08	53	47	40	51	45	30
20	2014-01-10	72	65	57	64	61	54
22	2014-01-12	67	57	46	58	47	33
37	2014-01-27	60	47	34	43	23	5
38	2014-01-28	39	33	27	19	14	7
...
1257	2017-05-31	85	77	68	70	68	62

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
1269	2017-06-12	94	85	75	72	68	62
1281	2017-06-24	85	79	72	76	71	66
1282	2017-06-25	90	83	75	74	71	67
1284	2017-06-27	91	81	71	74	72	67

192 rows × 21 columns

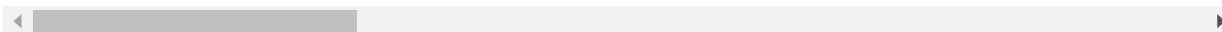


In [146]: `#groupby() This is
dell.head(3)`

Out[146]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	Hui
0	2013-12-21	74	60	45	67	49	43	
1	2013-12-22	56	48	39	43	36	28	
2	2013-12-23	58	45	32	31	27	23	

3 rows × 21 columns



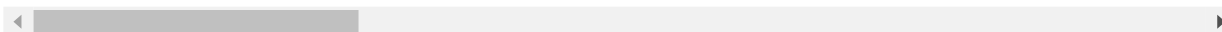
In [149]: `dell.groupby('Events').get_group('Rain')`

Out[149]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
18	2014-01-08	53	47	40	51	45	30

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
20	2014-01-10	72	65	57	64	61	54
22	2014-01-12	67	57	46	58	47	33
37	2014-01-27	60	47	34	43	23	5
38	2014-01-28	39	33	27	19	14	7
...
1257	2017-05-31	85	77	68	70	68	62
1269	2017-06-12	94	85	75	72	68	62
1281	2017-06-24	85	79	72	76	71	66
1282	2017-06-25	90	83	75	74	71	67
1284	2017-06-27	91	81	71	74	72	67

192 rows × 21 columns



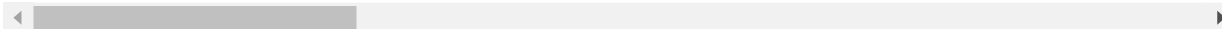
In [151]: `dell.head(5)`

Out[151]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	Hui
0	2013-12-21	74	60	45	67	49	43	
1	2013-12-22	56	48	39	43	36	28	
2	2013-12-23	58	45	32	31	27	23	

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	Humi
3	2013-12-24	61	46	31	36	28	21	
4	2013-12-25	58	50	41	44	40	36	

5 rows × 21 columns



In [162]: `dell['WindAvgMPH'] == 4` *#will give u boolean number*

Out[162]:

0	False
1	False
2	False
3	False
4	False
...	
1314	False
1315	False
1316	False
1317	False
1318	False

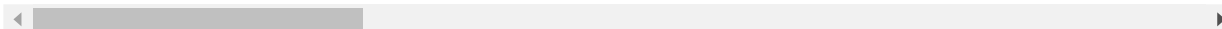
Name: WindAvgMPH, Length: 1319, dtype: bool

In [163]: `dell[dell['WindAvgMPH'] == 4]`

Out[163]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	Humi
--	------	-----------	----------	----------	---------------	--------------	--------------	------

0 rows × 21 columns



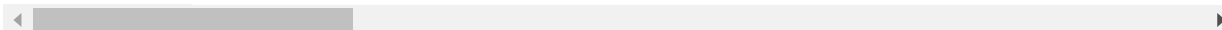
In [165]: `dell`

Out[165]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	Humi
--	------	-----------	----------	----------	---------------	--------------	--------------	------

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
0	2013-12-21	74	60	45	67	49	43
1	2013-12-22	56	48	39	43	36	28
2	2013-12-23	58	45	32	31	27	23
3	2013-12-24	61	46	31	36	28	21
4	2013-12-25	58	50	41	44	40	36
...
1314	2017-07-27	103	89	75	71	67	61
1315	2017-07-28	105	91	76	71	64	55
1316	2017-07-29	107	92	77	72	64	55
1317	2017-07-30	106	93	79	70	68	63
1318	2017-07-31	99	88	77	66	61	54

1319 rows × 21 columns



In [169]: `dell.WindHighMPH.value_counts()`

Out[169]:

13	212
12	195
14	162
10	140
15	113
16	99
9	98

```

17      80
8       74
18      43
7       27
20      25
21      22
22      10
24       5
23       4
25       3
-        2
28       2
29       1
6        1
26       1
Name: WindHighMPH, dtype: int64

```

```
In [177]: dell['WindAvgMPH'] == 4
```

```

Out[177]: 0      False
1      False
2      False
3      False
4      False
...
1314   False
1315   False
1316   False
1317   False
1318   False
Name: WindAvgMPH, Length: 1319, dtype: bool

```

```
In [179]: dell[dell['WindAvgMPH'] == 4]
```

```

Out[179]:
   Date  TempHighF  TempAvgF  TempLowF  DewPointHighF  DewPointAvgF  DewPointLowF  Humi
0 rows × 21 columns

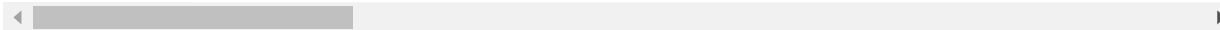
```

In [181]: dell

Out[181]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
0	2013-12-21	74	60	45	67	49	43
1	2013-12-22	56	48	39	43	36	28
2	2013-12-23	58	45	32	31	27	23
3	2013-12-24	61	46	31	36	28	21
4	2013-12-25	58	50	41	44	40	36
...
1314	2017-07-27	103	89	75	71	67	61
1315	2017-07-28	105	91	76	71	64	55
1316	2017-07-29	107	92	77	72	64	55
1317	2017-07-30	106	93	79	70	68	63
1318	2017-07-31	99	88	77	66	61	54

1319 rows × 21 columns



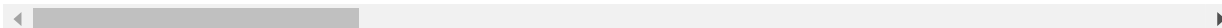
In [183]: dell.isnull()

Out[183]:

Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
------	-----------	----------	----------	---------------	--------------	--------------

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF
0	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False
...
1314	False	False	False	False	False	False	False
1315	False	False	False	False	False	False	False
1316	False	False	False	False	False	False	False
1317	False	False	False	False	False	False	False
1318	False	False	False	False	False	False	False

1319 rows × 21 columns



In [185]: dell.isnull().sum()

```
Out[185]: Date                                0
TempHighF                                    0
TempAvgF                                      0
TempLowF                                      0
DewPointHighF                                0
DewPointAvgF                                  0
DewPointLowF                                  0
HumidityHighPercent                           0
HumidityAvgPercent                            0
HumidityLowPercent                           0
SeaLevelPressureHighInches                    0
SeaLevelPressureAvgInches                     0
SeaLevelPressureLowInches                     0
VisibilityHighMiles                           0
VisibilityAvgMiles                            0
```

```
VisibilityLowMiles      0
WindHighMPH             0
WindAvgMPH              0
WindGustMPH             0
PrecipitationSumInches  0
Events                  0
dtype: int64
```

```
In [187]: dell.notnull().sum()
```

```
Out[187]: Date          1319
TempHighF             1319
TempAvgF              1319
TempLowF              1319
DewPointHighF         1319
DewPointAvgF          1319
DewPointLowF          1319
HumidityHighPercent   1319
HumidityAvgPercent    1319
HumidityLowPercent    1319
SeaLevelPressureHighInches 1319
SeaLevelPressureAvgInches 1319
SeaLevelPressureLowInches 1319
VisibilityHighMiles   1319
VisibilityAvgMiles    1319
VisibilityLowMiles    1319
WindHighMPH          1319
WindAvgMPH           1319
WindGustMPH          1319
PrecipitationSumInches 1319
Events               1319
dtype: int64
```

```
In [188]: import pandas as pd
```

```
In [190]: old=pd.read_csv(r"C:\Users\amolb\.ipynb_checkpoints\delli_weather.csv")
```

```
In [191]: old
```

Out[191]:

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem
0	19961101-11:00	Smoke	9.0	0	0	NaN	27.0	NaN	1010.0
1	19961101-12:00	Smoke	10.0	0	0	NaN	32.0	NaN	-9999.0
2	19961101-13:00	Smoke	11.0	0	0	NaN	44.0	NaN	-9999.0
3	19961101-14:00	Smoke	10.0	0	0	NaN	41.0	NaN	1010.0
4	19961101-16:00	Smoke	11.0	0	0	NaN	47.0	NaN	1011.0
...
100985	20170424-06:00	Haze	17.0	0	0	NaN	25.0	NaN	1005.0
100986	20170424-09:00	Haze	14.0	0	0	NaN	16.0	NaN	1003.0
100987	20170424-12:00	Haze	12.0	0	0	NaN	14.0	NaN	1002.0
100988	20170424-15:00	Haze	15.0	0	0	NaN	27.0	NaN	1004.0
100989	20170424-18:00	Haze	15.0	0	0	NaN	30.0	NaN	1005.0

100990 rows × 20 columns



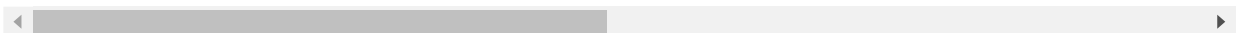
In [194]: `old.rename(columns= {'rain': 'Rain'})`

Out[194]:

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem
0	19961101-11:00	Smoke	9.0	0	0	NaN	27.0	NaN	1010.0

	datetime_utc	_conds	_dewptm	_fog	_hail	_heatindexm	_hum	_precipm	_pressurem
1	19961101-12:00	Smoke	10.0	0	0	NaN	32.0	NaN	-9999.0
2	19961101-13:00	Smoke	11.0	0	0	NaN	44.0	NaN	-9999.0
3	19961101-14:00	Smoke	10.0	0	0	NaN	41.0	NaN	1010.0
4	19961101-16:00	Smoke	11.0	0	0	NaN	47.0	NaN	1011.0
...
100985	20170424-06:00	Haze	17.0	0	0	NaN	25.0	NaN	1005.0
100986	20170424-09:00	Haze	14.0	0	0	NaN	16.0	NaN	1003.0
100987	20170424-12:00	Haze	12.0	0	0	NaN	14.0	NaN	1002.0
100988	20170424-15:00	Haze	15.0	0	0	NaN	27.0	NaN	1004.0
100989	20170424-18:00	Haze	15.0	0	0	NaN	30.0	NaN	1005.0

100990 rows × 20 columns



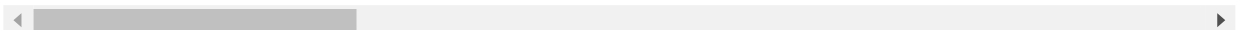
In [197]: `dell.rename(columns= {'DewPointHighF': 'DewPtHighF'})`

Out[197]:

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Hu
0	2013-12-21	74	60	45	67	49	43	
1	2013-12-22	56	48	39	43	36	28	
2	2013-12-23	58	45	32	31	27	23	

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Hu
3	2013-12-24	61	46	31	36	28	21	
4	2013-12-25	58	50	41	44	40	36	
...	
1314	2017-07-27	103	89	75	71	67	61	
1315	2017-07-28	105	91	76	71	64	55	
1316	2017-07-29	107	92	77	72	64	55	
1317	2017-07-30	106	93	79	70	68	63	
1318	2017-07-31	99	88	77	66	61	54	

1319 rows × 21 columns



In [198]: dell.head()

Out[198]:

	Date	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	Hu
0	2013-12-21	74	60	45	67	49	43	
1	2013-12-22	56	48	39	43	36	28	
2	2013-12-23	58	45	32	31	27	23	
3	2013-12-24	61	46	31	36	28	21	
4	2013-12-25	58	50	41	44	40	36	

5 rows × 21 columns

◀ ▶

```
In [199]: dell.rename(columns= {'DewPointHighF': 'DewPtHighF'}, inplace = True)
```

```
In [200]: dell.head(5)
```

Out[200]:

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Humid
0	2013-12-21	74	60	45	67	49	43	
1	2013-12-22	56	48	39	43	36	28	
2	2013-12-23	58	45	32	31	27	23	
3	2013-12-24	61	46	31	36	28	21	
4	2013-12-25	58	50	41	44	40	36	

5 rows × 21 columns

◀ ▶

```
In [210]: dell.TempHighF.std()
```

Out[210]: 14.766522914053345

```
In [211]: dell.TempAvgF.var()
```

Out[211]: 197.2874136727024

```
In [222]: #by using value_counts()
dell.Events.value_counts()
```

Out[222]:

	903
Rain	192

```

Rain , Thunderstorm      137
Fog , Rain , Thunderstorm 33
Fog                        21
Thunderstorm              17
Fog , Rain                14
Rain , Snow                1
Fog , Thunderstorm        1
Name: Events, dtype: int64

```

```

In [223]: #by using filtering
dell.Events == 'Fog'

```

```

Out[223]: 0      False
          1      False
          2      False
          3      False
          4      False
          ...
        1314    False
        1315    False
        1316    False
        1317    False
        1318    False
Name: Events, Length: 1319, dtype: bool

```

```

In [224]: dell[dell.Events == 'Fog']

```

```

Out[224]:

```

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Hu
19	2014-01-09	70	62	53	60	55	50	
50	2014-02-09	76	59	41	53	45	38	
57	2014-02-16	76	68	59	61	57	53	
70	2014-03-01	82	68	53	65	57	48	

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Hu
90	2014-03-21	80	67	54	60	54	41	
374	2014-12-30	46	43	39	43	40	32	
387	2015-01-12	46	43	40	45	41	36	
477	2015-04-12	82	75	67	69	67	65	
681	2015-11-02	79	68	57	60	58	55	
682	2015-11-03	80	69	57	62	59	55	
711	2015-12-02	68	55	42	48	40	31	
720	2015-12-11	79	66	52	65	61	51	
747	2016-01-07	73	61	49	53	50	46	
772	2016-02-01	77	66	54	61	54	37	
845	2016-04-14	77	66	54	58	56	53	
1045	2016-10-31	86	73	60	69	62	58	
1059	2016-11-14	82	69	56	62	58	55	
1081	2016-12-06	58	51	43	48	45	43	
1107	2017-01-01	73	63	53	60	52	39	
1127	2017-01-21	81	65	48	58	51	33	

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Hu
1142	2017-02-05	72	62	51	64	59	51	

21 rows × 21 columns



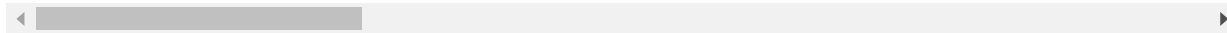
In [226]: `#group by()`

In [227]: `dell.head(10)`

Out[227]:

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Humid
0	2013-12-21	74	60	45	67	49	43	
1	2013-12-22	56	48	39	43	36	28	
2	2013-12-23	58	45	32	31	27	23	
3	2013-12-24	61	46	31	36	28	21	
4	2013-12-25	58	50	41	44	40	36	
5	2013-12-26	57	48	39	39	36	33	
6	2013-12-27	60	53	45	41	39	37	
7	2013-12-28	62	51	40	43	39	33	
8	2013-12-29	64	50	36	49	41	28	
9	2013-12-30	44	40	35	31	26	21	

10 rows × 21 columns

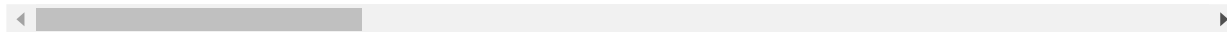


```
In [234]: #str.contains()  
dell[dell.Events.str.contains('Rain')].head(5)
```

Out[234]:

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Humi
0	2013-12-21	74	60	45	67	49	43	
18	2014-01-08	53	47	40	51	45	30	
20	2014-01-10	72	65	57	64	61	54	
22	2014-01-12	67	57	46	58	47	33	
33	2014-01-23	56	42	27	42	29	20	

5 rows × 21 columns



```
In [239]: dell.WindHighMPH.value_counts()
```

```
Out[239]: 13    212  
          12    195  
          14    162  
          10    140  
          15    113  
          16     99  
           9     98  
          17     80  
           8     74  
          18     43  
           7     27  
          20     25  
          21     22
```

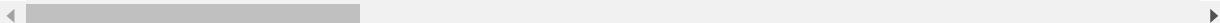
```
22      10
24       5
23       4
25       3
-        2
28       2
29       1
6        1
26       1
Name: WindHighMPH, dtype: int64
```

```
In [250]: #dell.head(5)
dell[(dell['TempHighF'] > 20) & (dell['VisibilityHighMiles'] == 10)]
```

```
Out[250]:
```

Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Humidity
------	-----------	----------	----------	------------	--------------	--------------	----------

0 rows × 21 columns



```
In [251]: dell.groupby('Events').mean()
```

```
Out[251]:
```

	TempHighF	TempAvgF	TempLowF
Events			
	82.807309	71.620155	59.895903
Fog	73.476190	62.809524	51.523810
Fog , Rain	71.500000	62.785714	53.714286
Fog , Rain , Thunderstorm	83.242424	74.757576	65.939394
Fog , Thunderstorm	89.000000	73.000000	57.000000
Rain	71.619792	64.484375	56.843750
Rain , Snow	56.000000	42.000000	27.000000
Rain , Thunderstorm	81.408759	72.824818	63.773723

	TempHighF	TempAvgF	TempLowF
Events			
Thunderstorm	90.764706	80.411765	69.470588

```
In [255]: dell.groupby('Events').min()
#dell.groupby('Events').max()
```

Out[255]:

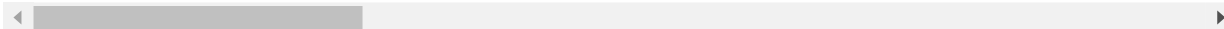
	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointL
Events							
	2013-12-22	32	29	19	-	-	
Fog	2014-01-09	46	43	39	43	40	
Fog , Rain	2014-03-15	50	42	34	49	45	
Fog , Rain , Thunderstorm	2014-04-14	59	52	34	55	46	
Fog , Thunderstorm	2015-04-19	89	73	57	69	58	
Rain	2014-01-08	36	30	23	19	12	
Rain , Snow	2014-01-23	56	42	27	42	29	
Rain , Thunderstorm	2013-12-21	36	34	24	27	20	
Thunderstorm	2014-03-28	66	55	44	57	50	

```
In [258]: dell[dell['Events'] == 'Fog'].head(5)
```

Out[258]:

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Humi
19	2014-01-09	70	62	53	60	55	50	
50	2014-02-09	76	59	41	53	45	38	
57	2014-02-16	76	68	59	61	57	53	
70	2014-03-01	82	68	53	65	57	48	
90	2014-03-21	80	67	54	60	54	41	

5 rows × 21 columns



In [261]: `dell[(dell['Events'] == 'Rain') | (dell['TempHighF'] > 70)]`

Out[261]:

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Hu
0	2013-12-21	74	60	45	67	49	43	
14	2014-01-04	71	57	42	55	48	38	
18	2014-01-08	53	47	40	51	45	30	
20	2014-01-10	72	65	57	64	61	54	
21	2014-01-11	75	62	48	54	37	20	
...	
1314	2017-07-27	103	89	75	71	67	61	
1315	2017-07-28	105	91	76	71	64	55	

	Date	TempHighF	TempAvgF	TempLowF	DewPtHighF	DewPointAvgF	DewPointLowF	Hu
1316	2017-07-29	107	92	77	72	64	55	
1317	2017-07-30	106	93	79	70	68	63	
1318	2017-07-31	99	88	77	66	61	54	

1098 rows × 21 columns

In [265]: `dell[(dell['Events'] == 'Thunderstrom') & (dell['DewPtHighF'] > 50)|(dell['VisibilityHighMiles'] >5)]`

```

-----
----
TypeError                                Traceback (most recent call l
ast)
<ipython-input-265-5ceef4a28e9> in <module>
----> 1 dell[(dell['Events'] == 'Thunderstrom') & (dell['DewPtHighF'] >
50)|(dell['VisibilityHighMiles'] >5)]

~\anaconda3\lib\site-packages\pandas\core\ops\common.py in new_method(s
elf, other)
    62         other = item_from_zerodim(other)
    63
--> 64         return method(self, other)
    65
    66         return new_method

~\anaconda3\lib\site-packages\pandas\core\ops\__init__.py in wrapper(se
lf, other)
    524         rvalues = extract_array(other, extract_numpy=True)
    525
--> 526         res_values = comparison_op(lvalues, rvalues, op)
    527
    528         return _construct_result(self, res_values, index=self.i
ndex, name=res_name)

```

```

~\anaconda3\lib\site-packages\pandas\core\ops\array_ops.py in compariso
n_op(left, right, op)
    245
    246     elif is_object_dtype(lvalues.dtype):
--> 247         res_values = comp_method_OBJECT_ARRAY(op, lvalues, rval
ues)
    248
    249     else:

~\anaconda3\lib\site-packages\pandas\core\ops\array_ops.py in comp_meth
od_OBJECT_ARRAY(op, x, y)
    55         result = libops.vec_compare(x.ravel(), y, op)
    56     else:
---> 57         result = libops.scalar_compare(x.ravel(), y, op)
    58     return result.reshape(x.shape)
    59

pandas\_libs\ops.pyx in pandas._libs.ops.scalar_compare()

```

TypeError: '>' not supported between instances of 'str' and 'int'

In [266]: `#car data analysis`

In [267]: `import pandas as pd`

In [281]: `car=pd.read_csv(r"C:\Users\amolb\.ipynb_checkpoints\Automobile_data.csv")`
`car`

Out[281]:

	symboling	normalized- losses	make	fuel- type	aspiration	num- of- doors	body- style	drive- wheels	engine- location	whe
0	3	?	alfa- romero	gas	std	two	convertible	rwd	front	8
1	3	?	alfa- romero	gas	std	two	convertible	rwd	front	8

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94
3	2	164	audi	gas	std	four	sedan	fwd	front	99
4	2	164	audi	gas	std	four	sedan	4wd	front	99
...
200	-1	95	volvo	gas	std	four	sedan	rwd	front	109
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109
202	-1	95	volvo	gas	std	four	sedan	rwd	front	109
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	109
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109

205 rows × 26 columns



In [283]: `car.isnull().sum()`

```
Out[283]: symboling          0
normalized-losses  0
make              0
fuel-type         0
aspiration        0
num-of-doors      0
body-style        0
drive-wheels      0
engine-location   0
wheel-base       0
length           0
width            0
height           0
curb-weight       0
engine-type       0
```

```
num-of-cylinders    0
engine-size         0
fuel-system         0
bore                0
stroke              0
compression-ratio   0
horsepower          0
peak-rpm            0
city-mpg            0
highway-mpg         0
price               0
dtype: int64
```

```
In [277]: car.shape
```

```
Out[277]: (205, 23)
```

```
In [280]: #car.isnull()
car.isnull().sum()
```

```
Out[280]: Unnamed: 0    0
make          0
fuel_type     0
aspiration    0
num_of_doors  0
body_style    0
drive_wheels  0
engine_location 0
wheel_base    0
length        0
width          0
height         0
curb_weight    0
engine_type    0
num_of_cylinders 0
engine_size    0
fuel_system    0
compression_ratio 0
horsepower     0
```

```
peak_rpm          0
city_mpg           0
highway_mpg        0
price             0
dtype: int64
```

```
In [286]: #car.make.value_counts()
car['make'].value_counts()
```

```
Out[286]: toyota          32
nissan              18
mazda              17
honda              13
mitsubishi         13
subaru             12
volkswagen         12
peugot             11
volvo              11
dodge              9
bmw                8
mercedes-benz      8
audi               7
plymouth           7
saab               6
porsche            5
isuzu              4
chevrolet          3
jaguar             3
alfa-romero        3
renault            2
mercury            1
Name: make, dtype: int64
```

```
In [330]: car
```

```
Out[330]:
```

symboling	normalized- losses	make	fuel- type	aspiration	num- of- doors	body- style	drive- wheels	engine- location	whe ba
-----------	-----------------------	------	---------------	------------	----------------------	----------------	------------------	---------------------	-----------

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.4
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.4
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8
4	2	164	audi	gas	std	four	sedan	4wd	front	99.8
...
200	-1	95	volvo	gas	std	four	sedan	rwd	front	100.0
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	100.0
202	-1	95	volvo	gas	std	four	sedan	rwd	front	100.0
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	100.0
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	100.0

205 rows × 26 columns



In [292]: `#car['body-style'].isin(['hatchback','sedan'])` #will show u boolean values
`car[car['body-style'].isin(['hatchback','sedan'])]` #will show u all the records

Out[292]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8

4	2	164	audi	gas	std	four	sedan	4wd	front	99
	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
5	2	?	audi	gas	std	two	sedan	fwd	front	99
6	1	158	audi	gas	std	four	sedan	fwd	front	105
...
200	-1	95	volvo	gas	std	four	sedan	rwd	front	109
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109
202	-1	95	volvo	gas	std	four	sedan	rwd	front	109
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	109
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109

166 rows × 26 columns



```
In [328]: #car.head(2)
car.shape
```

```
Out[328]: (205, 26)
```

```
In [313]: car['engine-location'].value_counts()
```

```
Out[313]: front    202
rear         3
Name: engine-location, dtype: int64
```

```
In [315]: car[car['engine-location'] == 'rear']
```

```
Out[315]:
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
--	-----------	-------------------	------	-----------	------------	--------------	------------	--------------	-----------------	------------

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
126	3	?	porsche	gas	std	two	hardtop	rwd	rear	86.6
127	3	?	porsche	gas	std	two	hardtop	rwd	rear	86.6
128	3	?	porsche	gas	std	two	convertible	rwd	rear	86.6

3 rows × 26 columns

◀		▶
---	--	---

In [318]: `car[~(car['engine-location'] == 'rear')]`

Out[318]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	86.6
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	86.6
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8
4	2	164	audi	gas	std	four	sedan	4wd	front	99.8
...
200	-1	95	volvo	gas	std	four	sedan	rwd	front	100.0
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	100.0
202	-1	95	volvo	gas	std	four	sedan	rwd	front	100.0
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	100.0
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	100.0

202 rows × 26 columns

◀		▶
---	--	---


```
In [326]: car[~(car['engine-location'] == 'rear')]
```

Out[326]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	whe ba
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94
3	2	164	audi	gas	std	four	sedan	fwd	front	99
4	2	164	audi	gas	std	four	sedan	4wd	front	99
...
200	-1	95	volvo	gas	std	four	sedan	rwd	front	109
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109
202	-1	95	volvo	gas	std	four	sedan	rwd	front	109
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	109
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109

202 rows × 26 columns



```
In [329]: car.shape
```

Out[329]: (205, 26)

```
In [337]: car.head(5)
```

Out[337]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8
4	2	164	audi	gas	std	four	sedan	4wd	front	99.4

5 rows × 26 columns



In [338]: `car['highway-mpg'] = car['highway-mpg'].apply(lambda x:x+3)`

In [339]: `car`

Out[339]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8
4	2	164	audi	gas	std	four	sedan	4wd	front	99.4
...
200	-1	95	volvo	gas	std	four	sedan	rwd	front	109.0

	symboling	normalized- losses	make	fuel- type	aspiration	num- of- doors	body- style	drive- wheels	engine- location	whe ba
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109
202	-1	95	volvo	gas	std	four	sedan	rwd	front	109
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	109
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109

205 rows × 26 columns



In []: