

The Weather Dataset is a country based data containing date, precipitation, snow_depth, tavg, tmin and tmax .

This data is available as a CSV file. We are going to analyze this data set using the Pandas DataFrame.

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

```
In [4]: data=pd.read_csv(r"C:\Users\KIIT\Downloads\archive\nordics_weather.csv")
```

```
In [5]: data
```

```
Out[5]:
```

	country	date	precipitation	snow_depth	tavg	tmax	tmin
0	Finland	1/1/2015	1.714141	284.545455	1.428571	2.912739	-1.015287
1	Finland	1/2/2015	10.016667	195.000000	0.553571	2.358599	-0.998718
2	Finland	1/3/2015	3.956061	284.294118	-1.739286	0.820382	-3.463871
3	Finland	1/4/2015	0.246193	260.772727	-7.035714	-3.110828	-9.502581
4	Finland	1/5/2015	0.036364	236.900000	-17.164286	-8.727564	-19.004487
...
5473	Sweden	12/27/2019	0.105079	141.220930	-4.277778	-2.391204	-8.993458
5474	Sweden	12/28/2019	1.117688	142.955224	-3.866667	-3.006542	-11.593056
5475	Sweden	12/29/2019	1.758669	136.725146	1.755556	3.692056	-4.516204
5476	Sweden	12/30/2019	0.561996	120.740741	4.900000	6.135648	1.859070
5477	Sweden	12/31/2019	0.848161	131.583333	1.722222	4.376606	-2.290278

5478 rows × 7 columns

```
In [6]: data.head()
```

```
Out[6]:
```

	country	date	precipitation	snow_depth	tavg	tmax	tmin
0	Finland	1/1/2015	1.714141	284.545455	1.428571	2.912739	-1.015287
1	Finland	1/2/2015	10.016667	195.000000	0.553571	2.358599	-0.998718
2	Finland	1/3/2015	3.956061	284.294118	-1.739286	0.820382	-3.463871
3	Finland	1/4/2015	0.246193	260.772727	-7.035714	-3.110828	-9.502581
4	Finland	1/5/2015	0.036364	236.900000	-17.164286	-8.727564	-19.004487

```
In [8]: data.shape
```

```
Out[8]: (5478, 7)
```

```
In [9]: data.index
```

```
Out[9]: RangeIndex(start=0, stop=5478, step=1)
```

In [10]: `data.columns`

```
Out[10]: Index(['country', 'date', 'precipitation', 'snow_depth', 'tavg', 'tmax',  
              'tmin'],  
              dtype='object')
```

In [11]: `data.dtypes`

```
Out[11]: country      object  
date      object  
precipitation  float64  
snow_depth  float64  
tavg      float64  
tmax      float64  
tmin      float64  
dtype: object
```

In [13]: `data['tmax'].unique()`

```
Out[13]: array([2.91273885, 2.35859873, 0.82038217, ..., 3.69205607, 6.13564815,  
              4.3766055 ])
```

In [14]: `data.nunique()`

```
Out[14]: country      3  
date      1826  
precipitation  5405  
snow_depth  3485  
tavg      4645  
tmax      5460  
tmin      5452  
dtype: int64
```

In [15]: `data.count`

```
Out[15]: <bound method DataFrame.count of          country      date  precipitation  snow_depth
tavg      tmax  \
0    Finland  1/1/2015      1.714141  284.545455    1.428571    2.912739
1    Finland  1/2/2015     10.016667  195.000000    0.553571    2.358599
2    Finland  1/3/2015      3.956061  284.294118   -1.739286    0.820382
3    Finland  1/4/2015      0.246193  260.772727   -7.035714   -3.110828
4    Finland  1/5/2015      0.036364  236.900000  -17.164286   -8.727564
...      ...      ...      ...      ...      ...      ...
5473  Sweden  12/27/2019      0.105079  141.220930   -4.277778   -2.391204
5474  Sweden  12/28/2019      1.117688  142.955224   -3.866667   -3.006542
5475  Sweden  12/29/2019      1.758669  136.725146    1.755556    3.692056
5476  Sweden  12/30/2019      0.561996  120.740741    4.900000    6.135648
5477  Sweden  12/31/2019      0.848161  131.583333    1.722222    4.376606

      tmin
0    -1.015287
1    -0.998718
2    -3.463871
3    -9.502581
4   -19.004487
...      ...
5473  -8.993458
5474 -11.593056
5475  -4.516204
5476   1.859070
5477  -2.290278

[5478 rows x 7 columns]>
```

In [19]: `data['tmax'].value_counts()`

```
Out[19]: 15.017532    2
         4.812500    2
         8.463636    2
         7.434234    2
         7.394828    2
         ..
         3.344156    1
         1.264052    1
        -4.703947    1
        -3.161039    1
         4.376606    1
Name: tmax, Length: 5460, dtype: int64
```

In [24]: `data.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5478 entries, 0 to 5477
Data columns (total 7 columns):
#   Column          Non-Null Count  Dtype
---  -
0   country         5478 non-null   object
1   date            5478 non-null   object
2   precipitation    5478 non-null   float64
3   snow_depth      5478 non-null   float64
4   tavg            5478 non-null   float64
5   tmax            5478 non-null   float64
6   tmin            5478 non-null   float64
dtypes: float64(5), object(2)
memory usage: 299.7+ KB
```

Find all unique 'Precipitation' values in the data

In [25]: `data.head(5)`

```
Out[25]:
```

	country	date	precipitation	snow_depth	tavg	tmax	tmin
0	Finland	1/1/2015	1.714141	284.545455	1.428571	2.912739	-1.015287
1	Finland	1/2/2015	10.016667	195.000000	0.553571	2.358599	-0.998718
2	Finland	1/3/2015	3.956061	284.294118	-1.739286	0.820382	-3.463871
3	Finland	1/4/2015	0.246193	260.772727	-7.035714	-3.110828	-9.502581
4	Finland	1/5/2015	0.036364	236.900000	-17.164286	-8.727564	-19.004487

In [31]: `data.nunique()`

```
Out[31]: country         3
date           1826
precipitation   5405
snow_depth     3485
tavg           4645
tmax           5460
tmin           5452
dtype: int64
```

In [32]: `data['tavg'].nunique()`

```
Out[32]: 4645
```

Find maximum snow_depth in the data

In [35]: `max_snow_depth=data['snow_depth'].max()`

In [36]: `max_snow_depth`

```
Out[36]: 682.9090909
```

Find out all the null values in the data

In [40]: `data.isnull().sum()`

```
Out[40]: country      0
         date         0
         precipitation 0
         snow_depth   0
         tavg         0
         tmax         0
         tmin         0
         dtype: int64
```

Find all the not null values in the data

In [44]: `data.notnull().sum()`

```
Out[44]: country      5478
         date         5478
         precipitation 5478
         snow_depth   5478
         tavg         5478
         tmax         5478
         tmin         5478
         dtype: int64
```

Find all Precipitation above 10

In [43]: `data[data['precipitation'] > 10]`

```
Out[43]:
```

	country	date	precipitation	snow_depth	tavg	tmax	tmin
1	Finland	1/2/2015	10.016667	195.000000	0.553571	2.358599	-0.998718
239	Finland	8/28/2015	15.034500	0.000000	14.053571	16.379870	12.421569
260	Finland	9/18/2015	18.083838	0.000000	13.700000	15.293464	12.005882
532	Finland	6/16/2016	11.087437	0.000000	15.707143	20.872368	10.272185
533	Finland	6/17/2016	11.154000	0.000000	15.014286	19.074342	12.613158
...
3962	Sweden	11/7/2015	10.082825	1.256831	8.100000	7.930952	3.687251
4634	Sweden	9/9/2017	11.066167	0.000000	11.900000	12.896833	8.829412
4657	Sweden	10/2/2017	12.330885	0.000000	10.555556	10.864220	7.830137
4708	Sweden	11/22/2017	11.598145	81.489362	-2.688889	-0.407407	-8.379817
5050	Sweden	10/30/2018	10.750769	12.270115	3.166667	4.146544	-3.484545

154 rows × 7 columns

Rename tavg, tmin and tmax to temp_avg, temp_min

```
In [45]: data.rename(columns = {'tavg' : 'temp_avg', 'tmin': 'temp_min', 'tmax': 'temp_max'}, inplace=True)
```

```
In [46]: data
```

```
Out[46]:
```

	country	date	precipitation	snow_depth	temp_avg	temp_max	temp_min
0	Finland	1/1/2015	1.714141	284.545455	1.428571	2.912739	-1.015287
1	Finland	1/2/2015	10.016667	195.000000	0.553571	2.358599	-0.998718
2	Finland	1/3/2015	3.956061	284.294118	-1.739286	0.820382	-3.463871
3	Finland	1/4/2015	0.246193	260.772727	-7.035714	-3.110828	-9.502581
4	Finland	1/5/2015	0.036364	236.900000	-17.164286	-8.727564	-19.004487
...
5473	Sweden	12/27/2019	0.105079	141.220930	-4.277778	-2.391204	-8.993458
5474	Sweden	12/28/2019	1.117688	142.955224	-3.866667	-3.006542	-11.593056
5475	Sweden	12/29/2019	1.758669	136.725146	1.755556	3.692056	-4.516204
5476	Sweden	12/30/2019	0.561996	120.740741	4.900000	6.135648	1.859070
5477	Sweden	12/31/2019	0.848161	131.583333	1.722222	4.376606	-2.290278

5478 rows × 7 columns

Find mean temp_max

```
In [47]: data.temp_max.mean()
```

```
Out[47]: 8.415625396744618
```

All data where precipitation is above 15 and snow_depth is above 200

```
In [48]: data[(data['precipitation'] > 15) & (data['snow_depth'] > 200)]
```

```
Out[48]:
```

	country	date	precipitation	snow_depth	temp_avg	temp_max	temp_min
1891	Norway	3/7/2015	15.306884	335.669767	3.838889	6.774107	1.541964
3630	Norway	12/10/2019	16.371429	201.222222	-5.111111	0.377451	-8.225962

Mean value of each country

```
In [49]: data.groupby('country').mean()
```

```
Out[49]:
```

	precipitation	snow_depth	temp_avg	temp_max	temp_min
country					
Finland	1.710121	182.610660	3.900884	7.453320	0.090045
Norway	3.625628	113.083423	4.442090	8.700597	1.914550
Sweden	1.784801	86.062509	6.189291	9.092960	0.927675

Minimum and Maximum value of each country

Minimum

```
In [50]: data.groupby('country').min()
```

```
Out[50]:
```

	date	precipitation	snow_depth	temp_avg	temp_max	temp_min
country						
Finland	1/1/2015	0.000000	0.000000	-27.951852	-24.197403	-29.63961
Norway	1/1/2015	0.010317	-0.168539	-12.168421	-9.824107	-14.49469
Sweden	1/1/2015	0.000000	-0.026178	-15.644444	-15.636546	-22.42120

Maximum

```
In [51]: data.groupby('country').max()
```

```
Out[51]:
```

	date	precipitation	snow_depth	temp_avg	temp_max	temp_min
country						
Finland	9/9/2019	18.083838	682.909091	24.407143	30.561438	17.930667
Norway	9/9/2019	21.947368	583.681818	20.258824	27.869903	16.573267
Sweden	9/9/2019	12.829487	534.726225	23.400000	29.789671	17.992891

All data where precipitation is above 15 and snow_depth is below 300 or temp_max is above 15

```
In [55]: data[(data['precipitation'] > 15) & (data['snow_depth'] < 300)|(data['temp_max'] > 15)
```

Out[55]:

	country	date	precipitation	snow_depth	temp_avg	temp_max	temp_min
141	Finland	5/22/2015	2.167005	112.500000	10.710714	15.458065	3.333548
145	Finland	5/26/2015	2.297462	69.750000	11.267857	15.975974	3.861290
146	Finland	5/27/2015	2.361929	67.250000	11.092857	15.142581	7.659091
148	Finland	5/29/2015	1.483249	24.750000	11.046429	15.940645	5.080000
149	Finland	5/30/2015	0.595431	5.000000	11.725000	16.250000	5.581169
...
5364	Sweden	9/9/2019	4.264561	0.000000	12.611111	15.276744	7.266977
5365	Sweden	9/10/2019	2.764737	0.000000	15.133333	17.635047	10.776279
5366	Sweden	9/11/2019	3.697544	0.000000	14.222222	17.266512	11.396729
5367	Sweden	9/12/2019	1.867018	0.000000	13.744444	16.728638	10.645794
5376	Sweden	9/21/2019	0.816140	0.247678	11.000000	15.026168	5.381221

1583 rows × 7 columns

```
In [ ]:
```