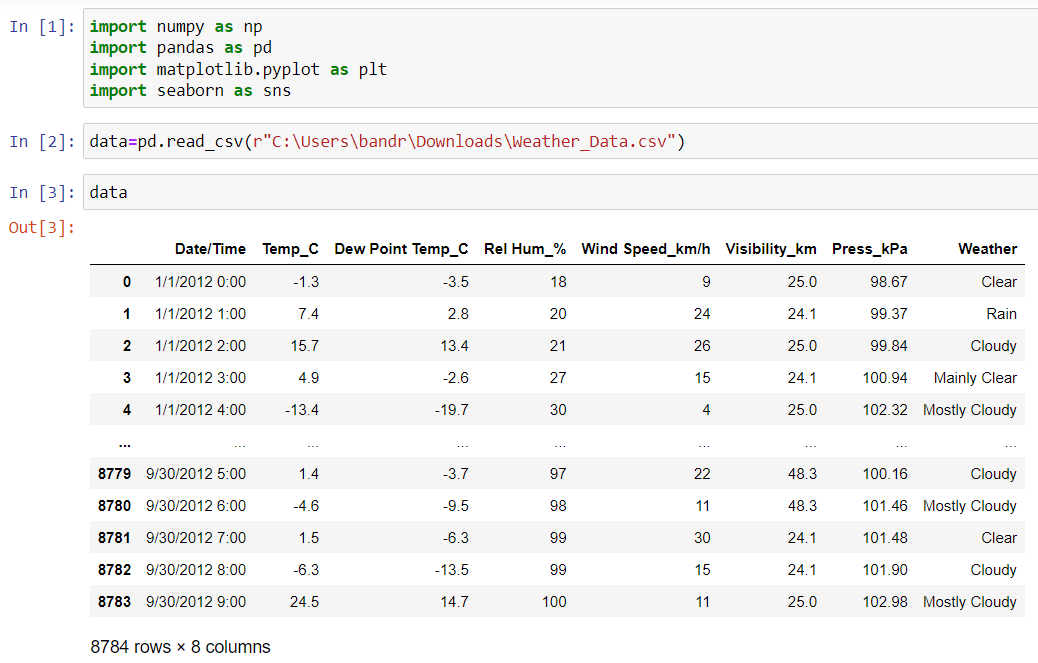
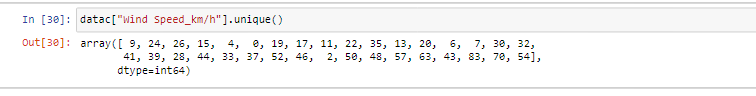
\*\*\* Report on weather \*\*\*

First the data we collected into the excel data should be loaded to the jupyter note book by using the code  
data=pd.read\_csv(r”copy path”) # will load the excel data into dataframe in pandas



Q. 1) Find all the unique 'Wind Speed' values in the data.

* To know the unique wind speed values in the data we use the code  
  datac[“Wind speed\_km/h”].unique()
* The above code will provide the information of what are the unique wind speed values in the data which are



* So here the output we obtain is an array of unique values of the wind speed as

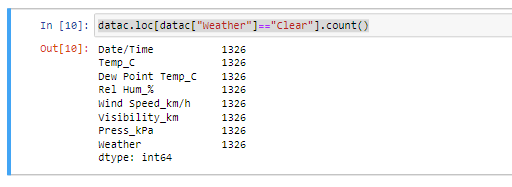
array([ 9, 24, 26, 15, 4, 0, 19, 17, 11, 22, 35, 13, 20, 6, 7, 30, 32,

41, 39, 28, 44, 33, 37, 52, 46, 2, 50, 48, 57, 63, 43, 83, 70, 54]

Q. 2) Find the number of times when the 'Weather is exactly Clear'.

* To know the number of times that weather is exactly clear we use the code

datac.loc[datac["Weather"]=="Clear"].count()  
will gives the output of the total no of count of the weather where the weather is exactly clear



* Here in the total count of 8784 different weather conditions we have

1326 data points whose weather is clear

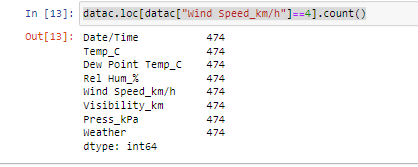
Q. 3) Find the number of times when the 'Wind Speed was exactly 4 km/h'.

* To know the number of times that wind speed was exactly 4km/h is

We use the code as

datac.loc[datac["Wind Speed\_km/h"]==4].count()

this code gives the output as below



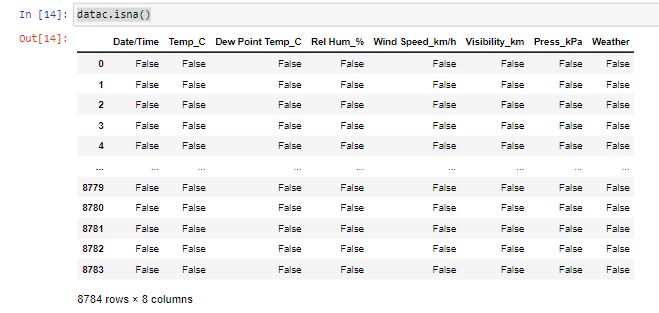
* Here we got the total number of times that wind speed is exactly 4 is 474

Q. 4) Find out all the Null Values in the data.

* To find the null values in the data we have to check the data using the code

datac.isna()

* will provides us a Boolean masking type table where in the table gives false if we do not have any null value and gives True if it consist of null values  
  🡪here is the output for our data



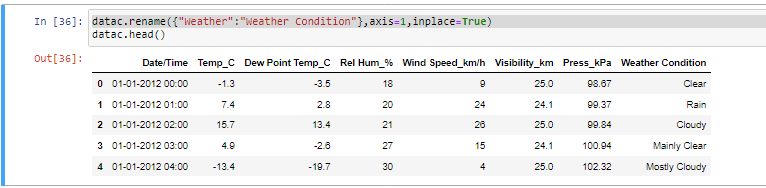
* By looking at the output we understand that the data do not have any null values

Q. 5) Rename the column name 'Weather' of the dataframe to 'Weather Condition'.

* To change the column name in a dataframe we use the code as

datac.rename({"Weather":"WeatherCondition"},axis=1,inplace=True)

* By the above code we can change the name of any column in the data



So data represents to be like the column name weather was changed to weather condition

Q. 6) What is the mean 'Visibility' ?

* To find the mean for the column “visibility” we simply use the code as

datac["Visibility\_km"].mean()

* It will gives the average (or) mean value of the required column as as central information of that column



For the data the variable (or) column name “Visibility\_km” the mean is 27.6644

Q. 7) What is the Standard Deviation of 'Pressure' in this data?

* To find the standard deviation for the column name “Pressure” we simply use the code as

datac["Press\_kPa"].std()

* It will provide the information of standard deviation for the required column as



* From the data the standard deviation for the column name “Press\_kPa” is 0.84400

Q. 8) What is the Variance of 'Relative Humidity' in this data ?

* To find the variance for the data in the column name “'Relative Humidity” is simly by usiging the code as

datac["Rel Hum\_%"].var()

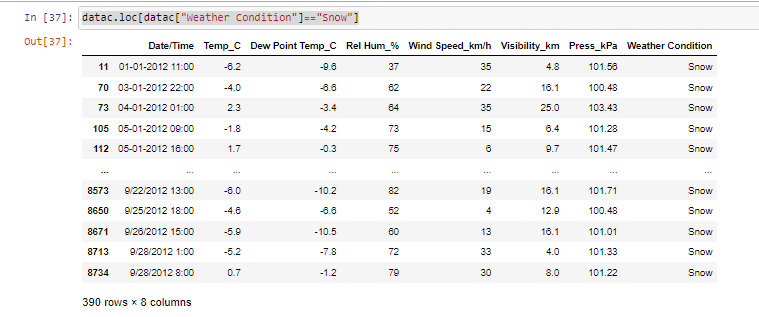
* It will gives the variance for the column name “'Relative Humidity” for the data



* Here the variance for the column name “Relative Humidity” is 286.24855

Q. 9) Find all instances when 'Snow' was recorded.

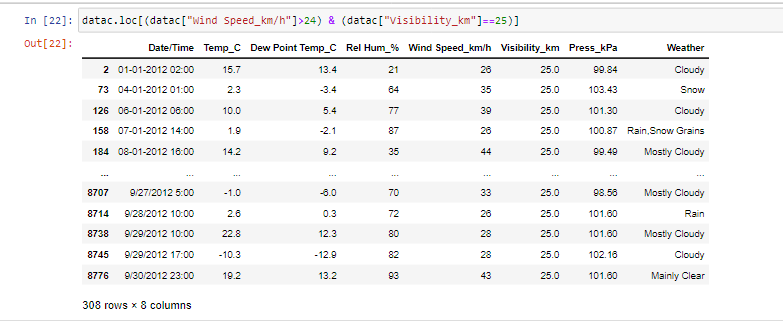
* To know the data whose weather condition is “snow” is by using the code  
  datac.loc[datac["Weather Condition"]=="Snow"]
* Here the data gives the dataframe table where the data consist of weather condition as snow



* The above output shows the information about the dataframe of the data whose weather condition is snow
* We have a count of 390 data poins whose weather condition is snow

Q. 10) Find all instances when 'Wind Speed is above 24' and 'Visibility is 25'.

* To know the data whose wind speed is greater than 24 and visibility is equal to 25 we use the code as   
  datac.loc[(datac["Wind Speed\_km/h"]>24) & (datac["Visibility\_km"]==25)]
* It gives the data of column name “wind speed”greater than 24 and also should satisfy the condition for the column name “visibility” is equal to 25



* Here is the data above where whose wind speed is greate than 24 and whose visibility is equal to 25
* We got 308 data point which satisfies both the conditions

Q. 11) What is the Mean value of each column against each 'Weather Condition ?

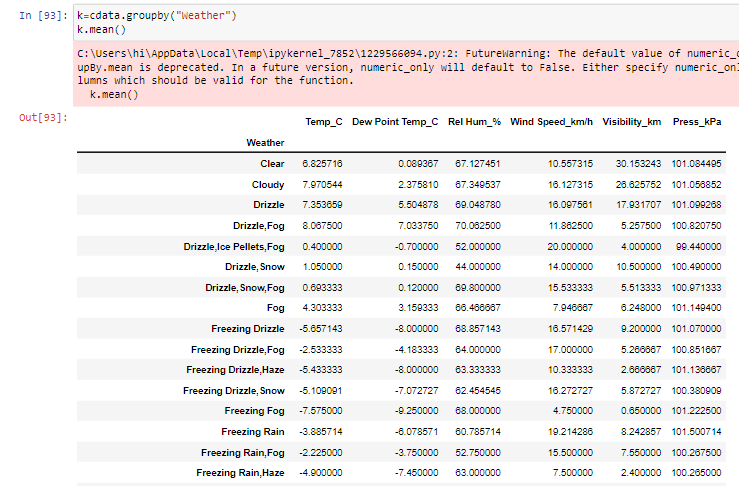
* To find the mean for column on the specified weather condition is

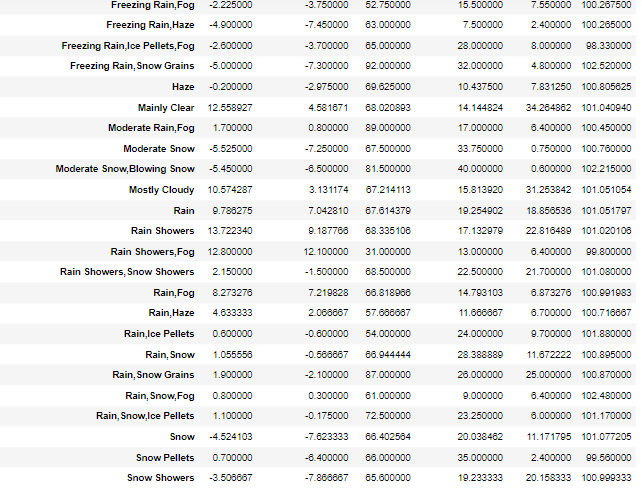
By using the code as

k=cdata.groupby("Weather")

k.mean()

* where the groupby will specify the data into different categories in the particulat column and also we can provide function we require by using aggregate function with it







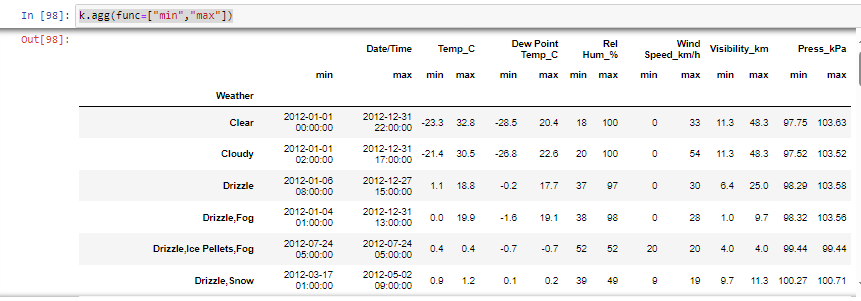
* Hence here is the mean all the data with the specified column weather condition
* It will represents the data as the mean for the temperature column data whose weather condition is cloudy is 0.825716
* Also the mean for all the other columns with the specified weather condition

Q. 12) What is the Minimum & Maximum value of each column against each 'Weather Condition ?

* To know the minimum and maximum data for all the columns on the specified column we use the code as

k.agg(func=["min","max"])

* where the k means the data which is already grouped by the condition for the particular required column name weather condition
* Also we use the aggregate function for the data wher we can use any type of function inside it



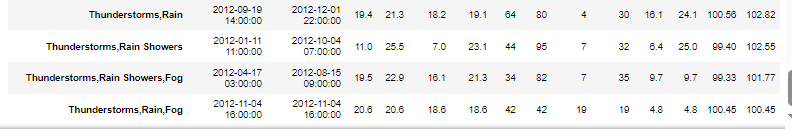












Here above is the data for the different weather conditions minimum and maximum values of the other parameters (or) columns in the data

* Here the minimum value of the (

| **Date/Time** | **Temp\_C** | **Dew Point Temp\_C** | **Rel Hum\_%** | **Wind Speed\_km/h** | **Visibility\_km** | **Press\_kPa)** |
| --- | --- | --- | --- | --- | --- | --- |

for the weather condition clear is (2012-01-01 00:00:00, -23.3, -28.5, 18,0, 11.3, 97.75) respectively

* Here the maximum value of the (

| **Date/Time** | **Temp\_C** | **Dew Point Temp\_C** | **Rel Hum\_%** | **Wind Speed\_km/h** | **Visibility\_km** | **Press\_kPa)** |
| --- | --- | --- | --- | --- | --- | --- |

for the weather condition clear is (2012-12-31 22:00:00, 32.8, 20.4, 100,33,48.3, 103.63) respectively

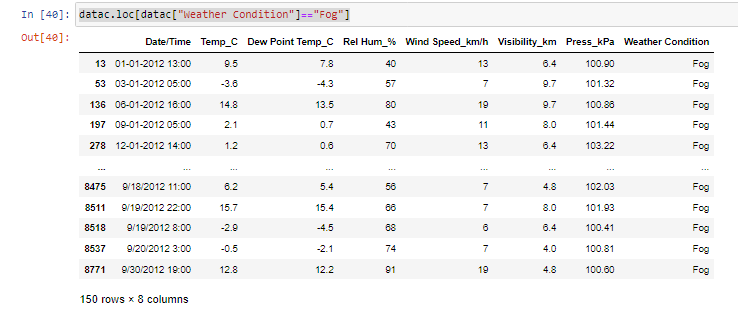
likely the data is in the above is given the maximum and minimum values of all the data on the specified weather condition

Q. 13) Show all the Records where Weather Condition is Fog.

* To know the data records whose weather condition is Fog we use the code as

datac.loc[datac["Weather Condition"]=="Fog"]

* It gives the entire data for the weather condition is equal to fog



* Here is the data whose weather condition is fog
* We have 150 different data points in the data whose weather condition is fog

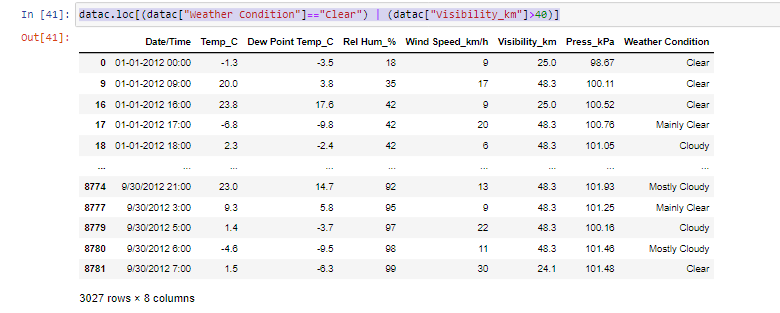
Q. 14) Find all instances when 'Weather is Clear' or 'Visibility is above 40'.

* To get the data for the conditions either weather condition is clear or visibility is 40

We use the code as

datac.loc[(datac["Weather Condition"]=="Clear") | (datac["Visibility\_km"]>40)]

* We get the data which satisfies the condition that weather condition is clear or visibility is greater than 40



* In the condition we should use the bitwise or (|)operator to satisfy the condition
* And the above data is the which satisfied the conditions
* We have 3027 number of data points in the data which satisfying the condition for weather condition is clear or visibility is greater than 40

Q. 15) Find all instances when :

A. 'Weather is Clear' and 'Relative Humidity is greater than 50'

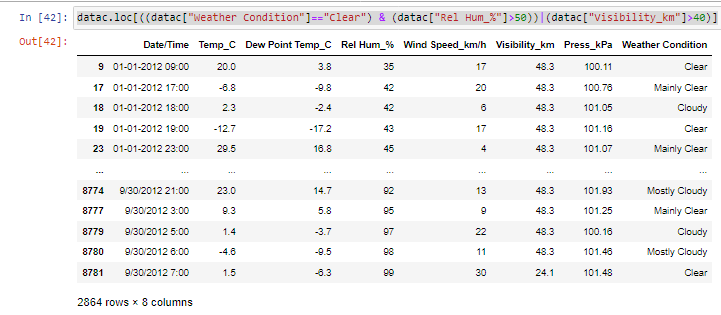
or

B. 'Visibility is above 40'

* To get the data for the conditions either weather condition is clear and Relative humidity greater than 50 or visibility is greater than 40
* We use the code as

datac.loc[((datac["Weather Condition"]=="Clear") & (datac["Rel Hum\_%"]>50))|(datac["Visibility\_km"]>40)]

* This condition satisfies the requirement for the weather condition is clear and relative humidity is greater than 50 or else the visibility is greater than 40



* In the condition we should use the bitwise and (&) and bitwise or (|)operators to satisfy the conditions
* And the above data is the which satisfied the conditions
* We have the total count of 2864 datapoints from the data which satisfies the condition for the weather condition should be clear and relative humidity greater than 50 orelse the visibility is greater than 40