DATA ANALYTICS

A SUMMER INTERNSHIP REPORT

on

PERFORMING ANALYSIS OF METEOROLOGICAL DATA

Under the subject of

summer internship (317001)

Submitted by

Neel Prajapati [191080131024]

In partial fulfillment for the award of the degree of

Bachelor of engineering in Computer science & technology

Computer Science & Engineering Department

Amiraj College Of Engineering and Technology

Gujarat Technological University, Ahamdabad

[YEAR 2022-23

AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

GUJARAT TECHNOLOGICAL UNIVERSITY

AMIRAJ COLLEGE OF ENGINEERING & TECHNOLOGY

SH 17 ,KHORAJ ,NEAR TATA NANO PLANT ,GUJARAT 33170

Certificate

this is verify that the internship at " TOPS TECH PVT .LTD " has been satisfactorily completed by neel prajapati j (191080131024)under any guidance in the fulfillment for the course of summer internship (3170001)in information technology during the academic year 2022-23 .

NENSI KANSAGARA

(FACULTY GUIDE -INTERNAL)

NENSI KANSAGARA

(HEAD OF DEPARTMENT)

COMPANY PROFILE

TOPS TECHNOLOGY PVT. LTD

TOPS TECHPVT. LTD , is a leader in technology , training , soft skill training and recruitment solutions . as a proven partner focus on building tomorrow's partner focused on building outform the completion and stay of the innovative curve .

In 2008 , two enterprising people started TOPS consultants with the objective of providing right manpower for banks . financial services , real estate and allied sectors , manufacturing , automobile ,media and internet /dot com - domain .

We have placed candidates from junior level to senior levels from the times of our inception .

We are specialized in the development and training of web and mobile technologies .

the internship program is designed to provide student engraved in a field experience with an opportunity to share their insights, to explore the link between students academic preparation and their field work , and to assist participants in developing and carrying out the major project of developing a portfolio website which will serve to culminate their internship experience

During this , the knowledge , skills and attitudes learned in the program can be applied. The aim of an internship provides a direction to the activities , and helps to focus on a result .

1 .AIM AND OBJECT OF INTERNSHIP

1. basic concept of python & data analysis.

2. import file from jupyter

3. data fetching

4. live project

professional manner .

perform clerical duties , take manner , maintain files , organize doc .

be familiar with python , numpy , scipy and pandas .

excellent analytical and problem solving skills .

ability to work in a team .

2 . roles and responsibilities during internship

my roles and responsibilities

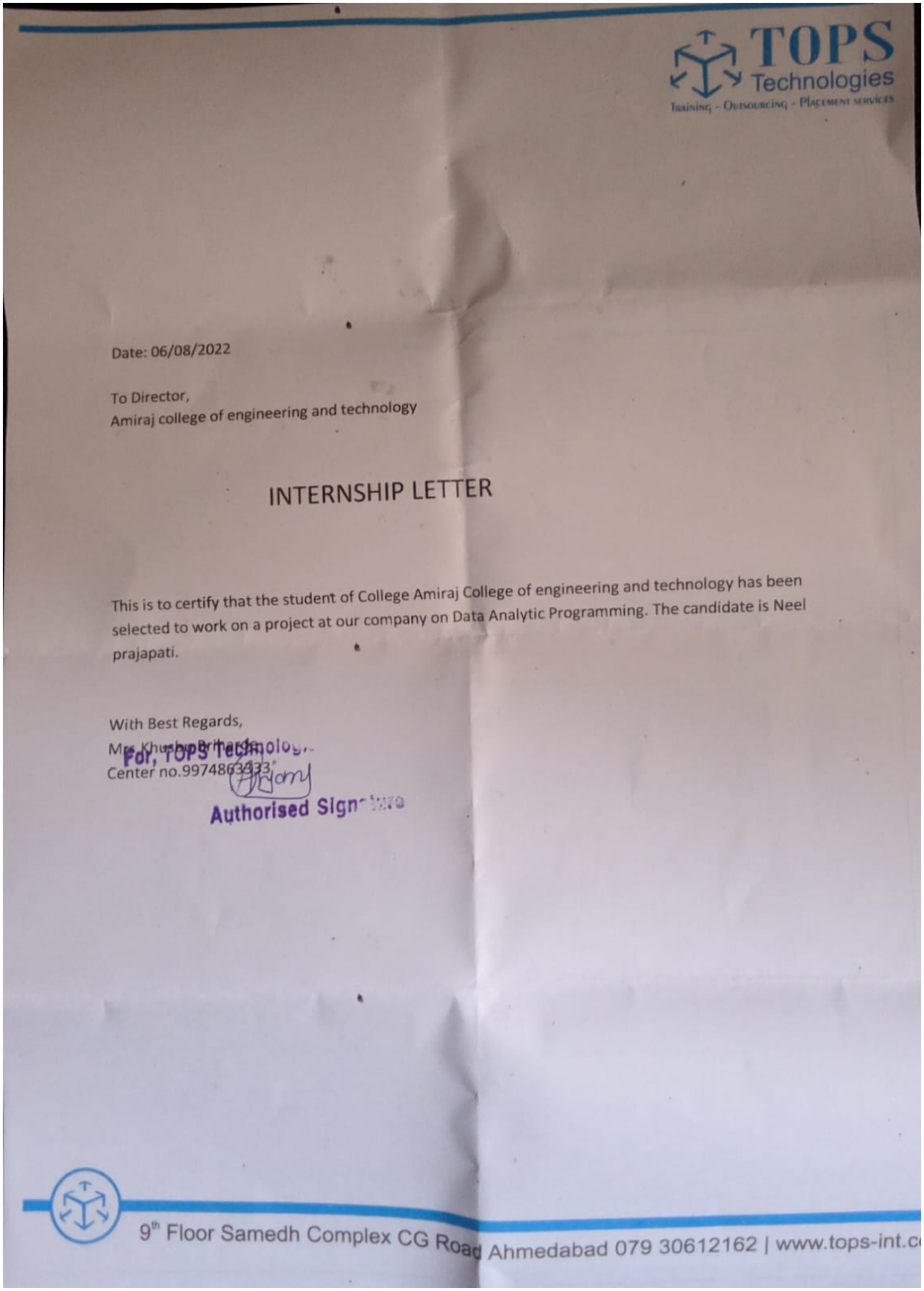
fulfill the task set out by supervisors from my department .

learn and gain experience .

conduct oneself in a responsible and

strong work ethic and attention .

website of company : https://www.tops-int.com



TOPS Technologies
Training - Outsourcing - Placement services

Date: 06/08/2022

To Director,
Amiraj college of engineering and technology

INTERNSHIP LETTER

This is to certify that the student of College Amiraj College of engineering and technology has been selected to work on a project at our company on Data Analytic Programming. The candidate is Neel prajapati.

With Best Regards,
Mrs. Khushbu Pritamdas
Center no.9974863333

For, TOPS Technologies
Authorised Signature

9th Floor Samedh Complex CG Road Ahmedabad 079 30612162 | www.tops-int.c

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to all those who provide me with the possibility of completing the internship . A special gratitude of thanks to the team of **TOPS TECH PVT .LTD** and external guide **Khushbu bhirade** ,whose contribution in stimulating suggestion and encouragement ,helped me to coordinate the internship especially in drafting his report .

Furthermore , I would also like to acknowledge with much appreciation the crucial role of the head of department prof . **nensi kansagara** , who have the permission to use all required equipment and the necessary material to fulfill the task . last but not least , many thanks to the teachers and my friends and families who have invested their full effort in guiding us in

archiving the goal.

Also, I appreciate the guidance given by developers at TOPS TECH PVT .LTD

NEEL PRAJAPATI J.

(191080131024)

ABSTRACT

A data analyst of meteorological data using python is all about playing with weather data related functions . There are various built in functions we must use in the python which enables the ability to easily perform the various data -related tasks . This meteorological data is very useful when we deal with analysis . doing analysis using python is more efficient then r language .

CODE

```
[ ]: #import libraries
import numpy as np # linear algebra
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

import warnings
warnings.filterwarnings("ignore")
```

```
[ ]: data = pd.read_csv('/kaggle/input/weather-dataset/weatherHist
ory.csv')
data.shape
```

O/P : (96453,12)

```
(96453, 12)
```

```
data.head()
```

Out[4]:

	Formatted Date	Summary	Precip Type	Temperature (C)	Apparent Temperature (C)	Humidity	Wind Speed (km/h)	Wi Beari (degree
0	2006-04-01 00:00:00.000 +0200	Partly Cloudy	rain	9.472222	7.388889	0.89	14.1197	251
1	2006-04-01 01:00:00.000 +0200	Partly Cloudy	rain	9.355556	7.227778	0.86	14.2646	259
2	2006-04-01 02:00:00.000 +0200	Mostly Cloudy	rain	9.377778	9.377778	0.89	3.9284	204
3	2006-04-01 03:00:00.000 +0200	Partly Cloudy	rain	8.288889	5.944444	0.83	14.1036	269
4	2006-04-01 04:00:00.000 +0200	Mostly Cloudy	rain	8.755556	6.977778	0.83	11.0446	259

```
df = data.copy()
df.dtypes
```

Out[5]:
```
Formatted Date               object
Summary                      object
Precip Type                  object
Temperature (C)             float64
Apparent Temperature (C)    float64
Humidity                    float64
Wind Speed (km/h)           float64
Wind Bearing (degrees)      float64
Visibility (km)             float64
Loud Cover                  float64
Pressure (millibars)        float64
Daily Summary                object
dtype: object
```

In [6]:
```
# let's correct the dtype of formatted date
df['Formatted Date'] = pd.to_datetime(df['Formatted Date'], u
tc=True)
```

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

Out[7]:
```
Formatted Date               datetime64[ns, UTC]
Summary                                   object
Precip Type                               object
Temperature (C)                          float64
Apparent Temperature (C)                 float64
Humidity                                 float64
Wind Speed (km/h)                        float64
Wind Bearing (degrees)                   float64
Visibility (km)                          float64
Loud Cover                               float64
Pressure (millibars)                     float64
Daily Summary                             object
dtype: object
```

```
In [8]: df.isnull().sum()
```

```
Out[8]: Formatted Date               0
        Summary                      0
        Precip Type                517
        Temperature (C)              0
        Apparent Temperature (C)     0
        Humidity                     0
        Wind Speed (km/h)            0
        Wind Bearing (degrees)       0
        Visibility (km)              0
        Loud Cover                   0
        Pressure (millibars)         0
        Daily Summary                0
        dtype: int64
```

EDA

CATEGORICAL FEATURES

```
In [11]: #drop the Daily summary column
         df.drop('Daily Summary', axis=1, inplace=True)
```

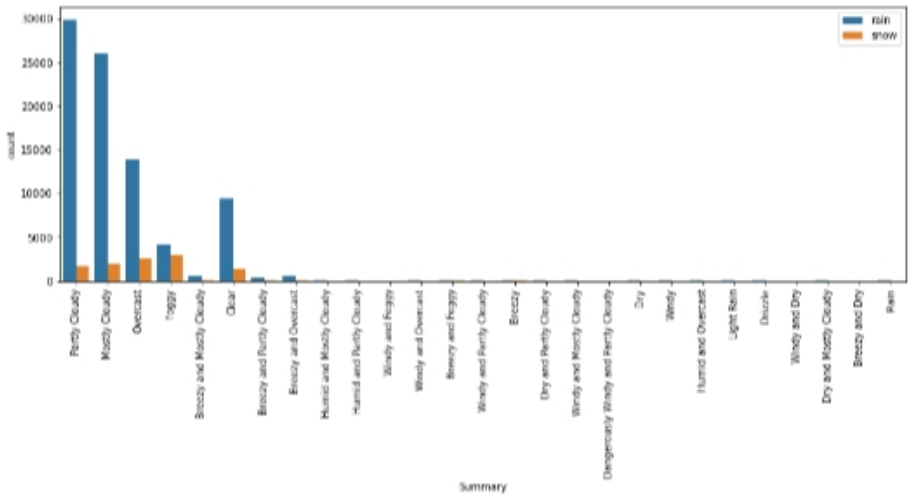
```
In [12]: print("%14s %15s %15s" %("column","Distinct","Null"))
         for col in ['Summary','Precip Type']:
             uniq = df[col].nunique()
             na = df[col].isnull().sum()
             print("%14s %15s %15s" %(col,uniq,na))
```

```
        column        Distinct            Null
       Summary              27               0
   Precip Type               2             517
```

```
In [13]: data.groupby('Precip Type')['Temperature (C)'].mean()
```

```
Out[13]: Precip Type
         rain    13.852989
         snow    -3.270885
         Name: Temperature (C), dtype: float64
```

In [14]:
```
plt.figure(figsize=(15,5))
sns.countplot(x='Summary',hue='Precip Type', data=df)
plt.legend(loc='upper right') #1
plt.xticks(rotation='vertical')
plt.show()
```



In [15]:
```
numerical_features = [feature for feature in df.columns if df
[feature].dtype != 'O']
print(f"len of categorical features {len(numerical_feature
s)}")
```

```
len of categorical features 9
```

In [16]:
```
df[numerical_features].head(3)
```

Out[16]:

	Formatted Date	Temperature (C)	Apparent Temperature (C)	Humidity	Wind Speed (km/h)	Wind Bearing (degrees)	Visibility (km)	Lo Co
0	2006-03-31 22:00:00+00:00	9.472222	7.388889	0.89	14.1197	251.0	15.8263	
1	2006-03-31 23:00:00+00:00	9.355556	7.227778	0.86	14.2646	259.0	15.8263	
2	2006-04-01 00:00:00+00:00	9.377778	9.377778	0.89	3.9284	204.0	14.9569	

After resampling

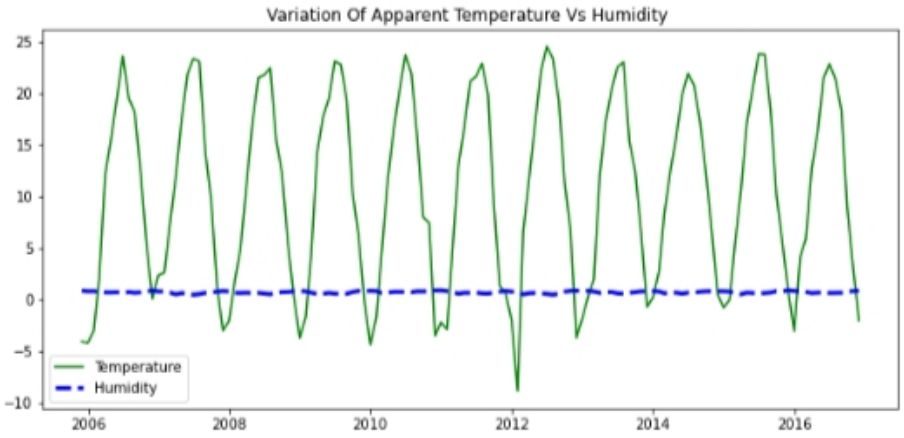
In [17]:
```
df1 = df.set_index('Formatted Date')
df1 = df1[['Apparent Temperature (C)','Humidity']].resample
('MS').mean()
df1
```

Out[17]:

Formatted Date	Apparent Temperature (C)	Humidity
2005-12-01 00:00:00+00:00	-4.050000	0.890000
2006-01-01 00:00:00+00:00	-4.173708	0.834610
2006-02-01 00:00:00+00:00	-2.990716	0.843467
2006-03-01 00:00:00+00:00	1.969780	0.778737
2006-04-01 00:00:00+00:00	12.098827	0.728625
...	...	...
2016-08-01 00:00:00+00:00	21.383094	0.674046
2016-09-01 00:00:00+00:00	18.355833	0.688833
2016-10-01 00:00:00+00:00	8.923947	0.799906
2016-11-01 00:00:00+00:00	3.048627	0.848472
2016-12-01 00:00:00+00:00	-2.017272	0.887981

133 rows × 2 columns

In [18]:
```
# relation between temperature and Humidity
plt.figure(figsize=(11,5))
plt.plot(df1['Apparent Temperature (C)'], label="Temperatur
e", color="green")
plt.plot(df1['Humidity'], label="Humidity", color="blue", lin
estyle="dashed", linewidth=3)
plt.title("Variation Of Apparent Temperature Vs Humidity ")
plt.legend(loc="best")
plt.show()
```



observation¶

from the above graph we can say that humidity is almost consult in these years , and temperature is also the same as peak lies on same line

the problem statement suggest that to check the variation of humidity of all the

10 years (2006 - 2016) so let's check the graph for some month separately

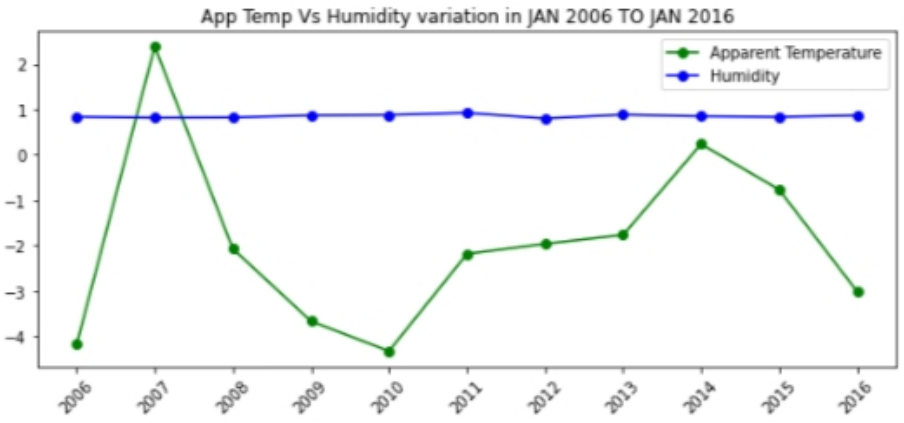
analysis of variation of temperature w.r.t humidity from 2006 to 2016 in different months

In [20]:
```
jan = df1[df1.index.month==1]
jan.head(2)
```

Out[20]:

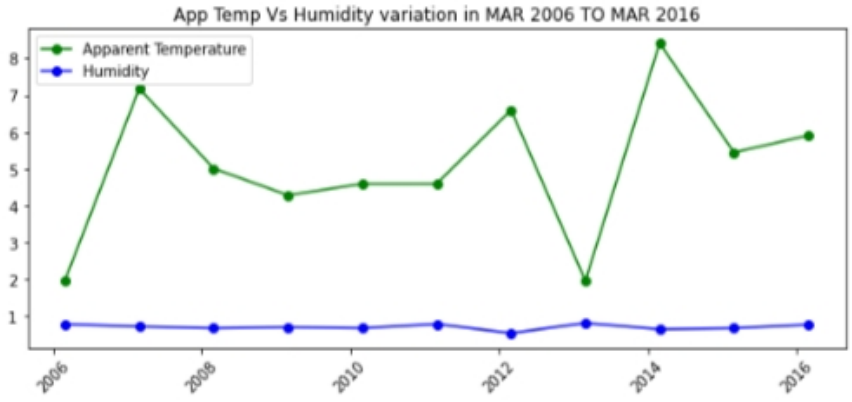
Formatted Date	Apparent Temperature (C)	Humidity
2006-01-01 00:00:00+00:00	-4.173708	0.834610
2007-01-01 00:00:00+00:00	2.387015	0.813495

In [21]:
```
plt.figure(figsize=(10,4))
plt.plot(jan.loc['2006-01-01':'2016-01-01','Apparent Temperature (C)'],
        marker='o',label="Apparent Temperature",linestyle='-',color="green")
plt.plot(jan.loc['2006-01-01':'2016-01-01','Humidity'],marker='o',
        linestyle='-',label="Humidity",color="blue")
plt.title("App Temp Vs Humidity variation in JAN 2006 TO JAN 2016")
plt.legend(loc="best")
plt.xticks(rotation=45)
plt.show()
```



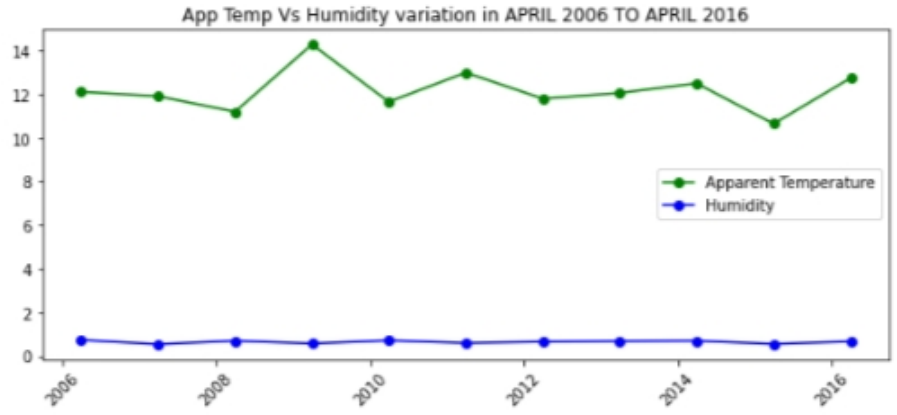
In [55]:
```
march = df1[df1.index.month==3]

plt.figure(figsize=(10,4))
plt.plot(march.loc['2006-03-01':'2016-03-01','Apparent Temperature (C)'],
        marker='o',linestyle='-',label="Apparent Temperature",color="green")
plt.plot(march.loc['2006-03-01':'2016-03-01','Humidity'],
        marker='o',linestyle='-',label="Humidity",color="blue")
plt.title("App Temp Vs Humidity variation in MAR 2006 TO MAR 2016")
plt.legend(loc="best")
plt.xticks(rotation=45)
plt.show()
```



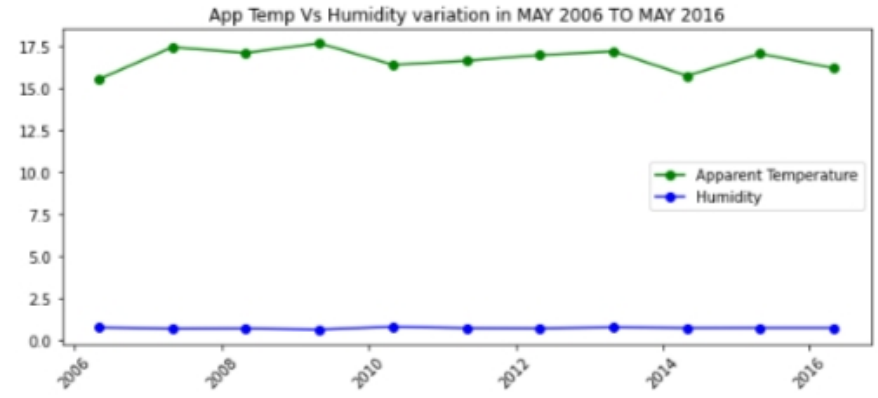
In [56]:
```
april = df1[df1.index.month==4]

plt.figure(figsize=(10,4))
plt.plot(april.loc['2006-04-01':'2016-04-01','Apparent Temperature (C)'],
        marker='o',linestyle='-',label="Apparent Temperature",color="green")
plt.plot(april.loc['2006-04-01':'2016-04-01','Humidity'],
        marker='o',linestyle='-',label="Humidity",color="blue")
plt.title("App Temp Vs Humidity variation in APRIL 2006 TO APRIL 2016")
plt.legend(loc="best")
plt.xticks(rotation=45)
plt.show()
```



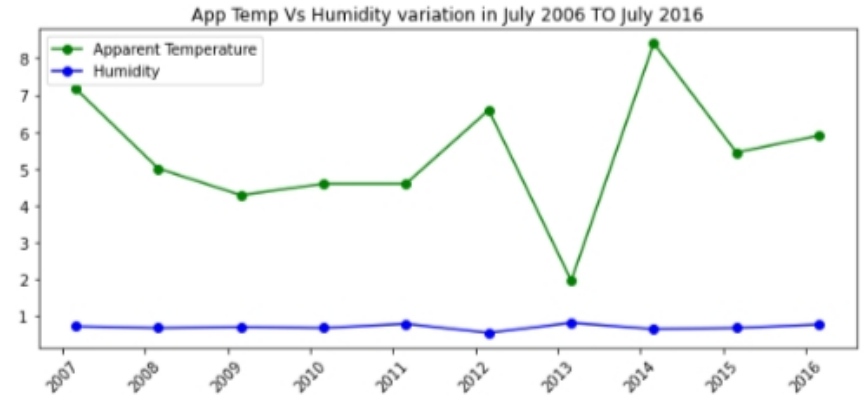
In [57]:
```
may = df1[df1.index.month==5]

plt.figure(figsize=(10,4))
plt.plot(may.loc['2006-05-01':'2016-05-01','Apparent Temperature (C)'],
        marker='o',linestyle='-',label="Apparent Temperature",color="green")
plt.plot(may.loc['2006-05-01':'2016-05-01','Humidity'],
        marker='o',linestyle='-',label="Humidity",color="blue")
plt.title("App Temp Vs Humidity variation in MAY 2006 TO MAY 2016")
plt.legend(loc="best")
plt.xticks(rotation=45)
plt.show()
```



In [59]:
```
july = df1[df1.index.month==6]

plt.figure(figsize=(10,4))
plt.plot(march.loc['2006-07-01':'2016-07-01','Apparent Temperature (C)'],
        marker='o',linestyle='-',label="Apparent Temperature",color="green")
plt.plot(march.loc['2006-07-01':'2016-07-01','Humidity'],
        marker='o',linestyle='-',label="Humidity",color="blue")
plt.title("App Temp Vs Humidity variation in July 2006 TO July 2016")
plt.legend(loc="best")
plt.xticks(rotation=45)
plt.show()
```



OBSERVATION

LET'S CHECK WIND SPEED W.R.T HUMIDITY

In [64]:
```
df2 = df.set_index('Formatted Date')
df2 = df2[['Temperature (C)','Wind Speed (km/h)','Humidit
y']].resample('MS').mean()
df2
```

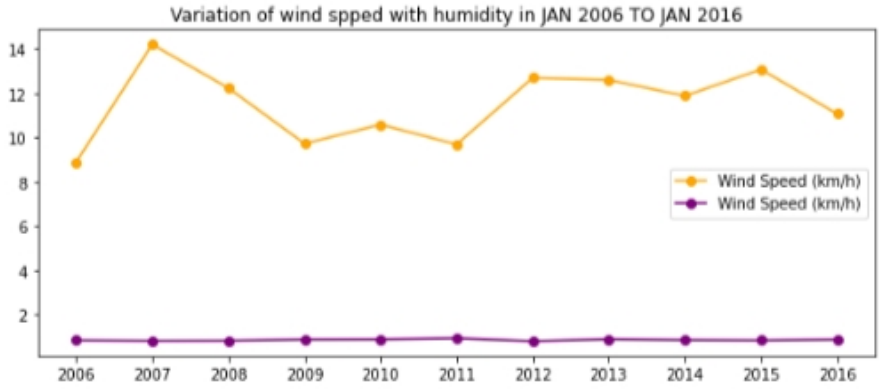
Out[64]:

Formatted Date	Temperature (C)	Wind Speed (km/h)	Humidity
2005-12-01 00:00:00+00:00	0.577778	17.114300	0.890000
2006-01-01 00:00:00+00:00	-1.677942	8.894211	0.834610
2006-02-01 00:00:00+00:00	-0.065394	10.957008	0.843467
2006-03-01 00:00:00+00:00	4.559274	14.421488	0.778737
2006-04-01 00:00:00+00:00	12.635031	10.930670	0.728625
...	...	...	...
2016-08-01 00:00:00+00:00	21.420296	9.151378	0.674046
2016-09-01 00:00:00+00:00	18.467924	6.849029	0.688833
2016-10-01 00:00:00+00:00	9.893242	11.566855	0.799906
2016-11-01 00:00:00+00:00	5.282662	10.302860	0.848472
2016-12-01 00:00:00+00:00	1.239158	11.024860	0.887981

133 rows × 3 columns

In [65]:
```
janw = df2[df2.index.month == 1]

plt.figure(figsize=(10,4))
plt.plot(janw.loc['2006-01-01':'2016-01-01','Wind Speed (km/
h)'],
        marker='o',linestyle='-',color="orange", label="Wind
Speed (km/h)")
plt.plot(janw.loc['2006-01-01':'2016-01-01','Humidity'],
        marker='o',linestyle='-',color="purple", label="Wind
Speed (km/h)")
plt.title("Variation of wind spped with humidity in JAN 2006
TO JAN 2016")
plt.legend(loc="best")
plt.show()
```



THANK YOU !

In [54]:
```
feb = df1[df1.index.month==2]

plt.figure(figsize=(10,4))
plt.plot(feb.loc['2006-02-01':'2016-02-01','Apparent Temperature (C)'],
        marker='o',linestyle='-',label="Apparent Temperature",color="green")
plt.plot(feb.loc['2006-02-01':'2016-02-01','Humidity'],
        marker='o',linestyle='-',label="Humidity",color="blue")
plt.title("App Temp Vs Humidity variation in FEB 2006 TO FEB 2016")
plt.legend(loc="best")
plt.xticks(rotation=45)
plt.show()
```