WeatherPy

Analysis

- Temperature started rising as we moved from negative latitude towards 0 and peaked near 20 then started decreasing again
- Between -20 to 20 latitude there are very few cities with low humidity
- Between 40-60 latidude there is high concentration of cities with high humidity
- Cloudiness seems to work in groups i.e. either 0% or 40% or 75% or 100% irrespective of the Latitude
- There is no marked pattern with repect to Latitude vs Windspeed though there is relatively lesser windiness around 0 latitude

Import all the dependencies

In [1]:

```
# Dependencies and Setup
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import requests as rq
import json
import datetime

# Import API key
from api_keys import api_key

# Incorporated citipy to determine city based on latitude and longitude
from citipy import citipy

# Output File (CSV)
output_data_file = "output_data/cities.csv"

# Range of latitudes and longitudes
lat_range = (-90, 90)
lng_range = (-180, 180)
```

Generate the list of random latitudes and longitudes

In [2]:

```
# List for holding lat_lngs and cities
lat_lngs = []

# Create a set of random lat and lng combinations
lats = np.random.uniform(low=-90.000, high=90.000, size=2000)
lngs = np.random.uniform(low=-180.000, high=180.000, size=2000)
lat_lngs = zip(lats, lngs)
```

Get the city name and country of these random latitudes and longitudes using citipy library

In [3]:

```
cities = []                         # to store the citipy object
countryCode = []                    # store the country code
cityName = []                       # to sotre the city name

# Identify nearest city for each lat, lng combination
for lat_lng in lat_lngs:
    city = citipy.nearest_city(lat_lng[0], lat_lng[1])
    cityN = city.city_name
    countryN = city.country_code

    # If the city is unique, then add it to a our cities list
    if city not in cities:
        cities.append(city)
        cityName.append(cityN)
        countryCode.append(countryN)

# Identify the country Code for the cities
#for city in cities:
    #countryCode.append(city.country_code)

# Print the city count to confirm sufficient count
len(cityName)
```

Out[3]:

```
756
```

Use cityName and countryCode to create the dataframe

In [4]:

```
city_df = pd.DataFrame(data = cityName, columns = ['CityName'])
city_df['CountryCode'] = countryCode
city_df['CityCountry'] = city_df['CityName'] + ',' + city_df['CountryCode']
city_df.head()
```

Out[4]:

	CityName	CountryCode	CityCountry
0	puerto ayora	ec	puerto ayora,ec
1	sioux lookout	ca	sioux lookout,ca
2	crotone	it	crotone,it
3	patiya	bd	patiya,bd
4	provideniya	ru	provideniya,ru

Build the base url string for the API

In [5]:

```
# Save config information.
url = "http://api.openweathermap.org/data/2.5/weather?"
units = "Imperial"

# Build partial query URL
query_url = f"{url}appid={api_key}&units={units}&q="
```

Generate Cities List

Perform API Calls

- Perform a weather check on each city using a series of successive API calls.
- Include a print log of each city as it'sbeing processed (with the city number and city name).

In [6]:

```
# set up lists to hold reponse info
cityNumber = []
temp = []
humidity = []
wind = []
clouds = []
latitude = []
name = []
country = []
i=1
print('Beginning Data Retrieval')
print('--------------------------')

# Loop through the list of cities (using the dataframe) and perform a request fo
for city in city_df['CityCountry']:

    print('Processing record ' + str(i) + '.... ' + city)
    print(query_url + city)

    i=i+1

    response = rq.get(query_url + city).json()

    # Handle exceptions if the city is not found in the API
    try:
        cityNumber.append(response['id'])
        temp.append(response['main']['temp_max'])
        humidity.append(response['main']['humidity'])
        wind.append(response['wind']['speed'])
        clouds.append(response['clouds']['all'])
        latitude.append(response['coord']['lat'])
        name.append(response['name'])
        country.append(response['sys']['country'])

    except:
        print(city + ': not found....Skipping')
```

```
Processing record 148.... tupiza,bo
http://api.openweathermap.org/data/2.5/weather?appid=1b4ad24191edb55ca4ebee0c
7e4937a4&units=Imperial&q=tupiza,bo (http://api.openweathermap.org/data/2.5/w
eather?appid=1b4ad24191edb55ca4ebee0c7e4937a4&units=Imperial&q=tupiza,bo)
Processing record 149.... asau,tv
http://api.openweathermap.org/data/2.5/weather?appid=1b4ad24191edb55ca4ebee0c
7e4937a4&units=Imperial&q=asau,tv (http://api.openweathermap.org/data/2.5/wea
ther?appid=1b4ad24191edb55ca4ebee0c7e4937a4&units=Imperial&q=asau,tv)
asau,tv: not found....Skipping
Processing record 150.... vilaka,lv
http://api.openweathermap.org/data/2.5/weather?appid=1b4ad24191edb55ca4ebee0c
7e4937a4&units=Imperial&q=vilaka,lv (http://api.openweathermap.org/data/2.5/w
eather?appid=1b4ad24191edb55ca4ebee0c7e4937a4&units=Imperial&q=vilaka,lv)
vilaka,lv: not found....Skipping
Processing record 151.... santa cruz,cr
http://api.openweathermap.org/data/2.5/weather?appid=1b4ad24191edb55ca4ebee0c
7e4937a4&units=Imperial&q=santa (http://api.openweathermap.org/data/2.5/weath
er?appid=1b4ad24191edb55ca4ebee0c7e4937a4&units=Imperial&q=santa) cruz,cr
Processing record 152.... faanui,pf
http://api.openweathermap.org/data/2.5/weather?appid=1b4ad24191edb55ca4ebee0c
```

Convert Raw Data to DataFrame

- Export the city data into a .csv.
- Display the DataFrame

In [7]:
```
weather_df = pd.DataFrame(data = name, columns = ['CityName'])
weather_df['City_id'] = cityNumber
weather_df['Country'] = country
weather_df['Latitude'] = latitude
weather_df['Max_Temperature'] = temp
weather_df['Humidity'] = humidity
weather_df['Cloudiness'] = clouds
weather_df['WindSpeed'] = wind


weather_df.to_csv('output_data/WeatherData.csv')
weather_df.head()
```

Out[7]:

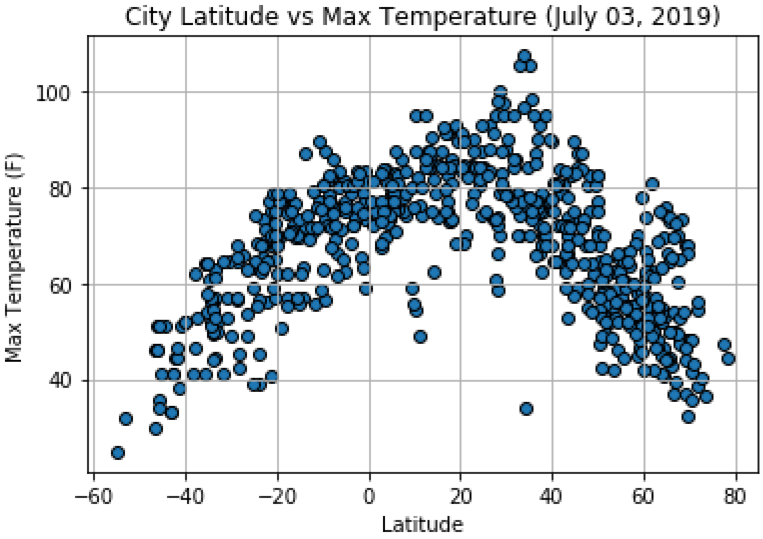
	CityName	City_id	Country	Latitude	Max_Temperature	Humidity	Cloudiness	WindSpeed
0	Puerto Ayora	3652764	EC	-0.74	77.00	61	0	19.46
1	Sioux Lookout	6148373	CA	50.10	77.00	47	75	8.05
2	Crotone	2524881	IT	39.09	77.12	50	0	4.23
3	Patiya	1185148	BD	22.30	84.50	79	100	10.29
4	Provideniya	4031574	RU	64.42	46.40	93	75	6.71

Plotting the Data

- Use proper labeling of the plots using plot titles (including date of analysis) and axes labels.
- Save the plotted figures as .pngs.

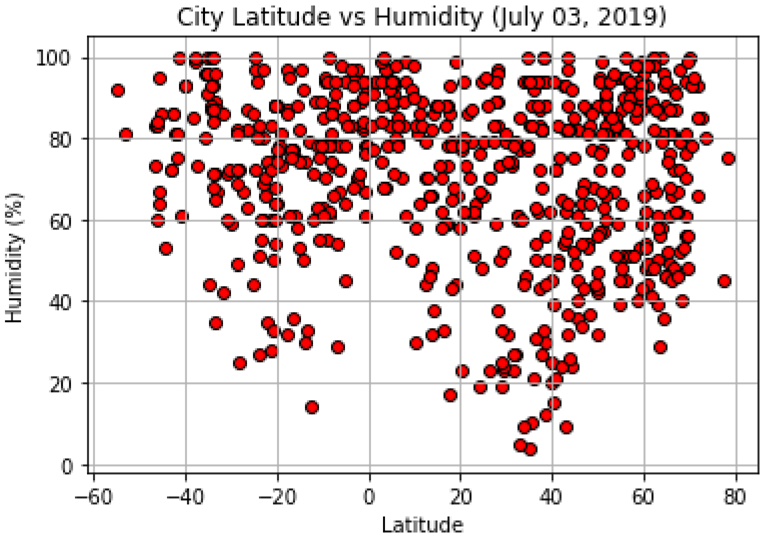
Latitude vs. Temperature Plot

In [8]:
```
plt.scatter(weather_df['Latitude'],weather_df['Max_Temperature'],edgecolor='blacl
plt.title('City Latitude vs Max Temperature' + " (" + datetime.date.today().strf
plt.xlabel('Latitude')
plt.ylabel('Max Temperature (F)')
plt.grid()
plt.savefig('output_data/Lat_vs_Temp.png')
plt.show()
```



Latitude vs. Humidity Plot

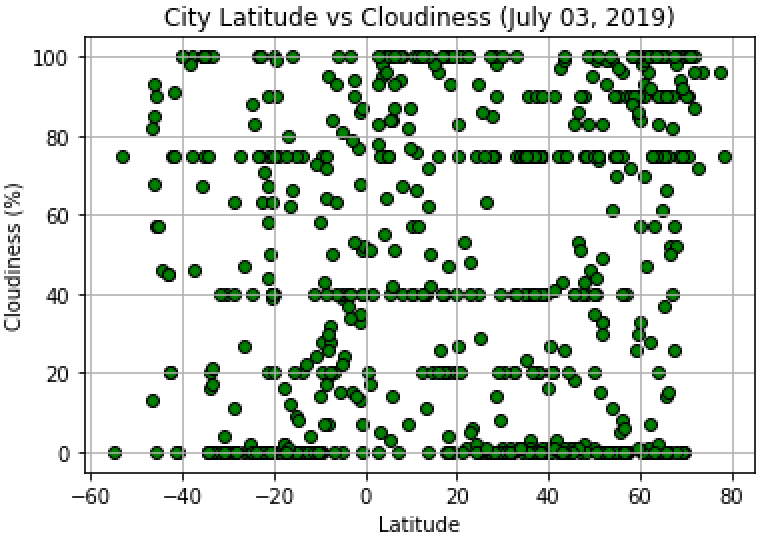
In [16]:
```
plt.scatter(weather_df['Latitude'],weather_df['Humidity'],edgecolors='black', co.
plt.title('City Latitude vs Humidity' + " (" + datetime.date.today().strftime("%l
plt.xlabel('Latitude')
plt.ylabel('Humidity (%)')
plt.ylim(-2,105)
plt.grid()
plt.savefig('output_data/Lat_vs_Humidity.png')
plt.show()
```



Latitude vs. Cloudiness Plot

In [10]:

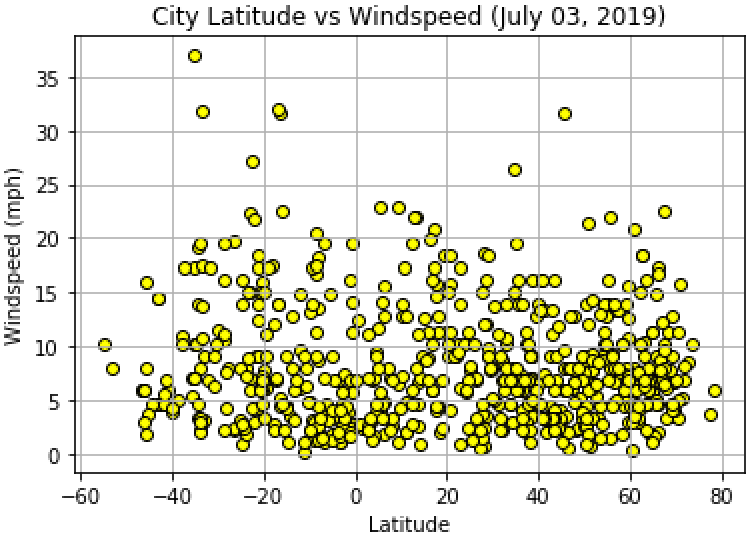
```
plt.scatter(weather_df['Latitude'],weather_df['Cloudiness'],edgecolors='black', 
plt.title('City Latitude vs Cloudiness' + " (" + datetime.date.today().strftime(
plt.xlabel('Latitude')
plt.ylabel('Cloudiness (%)')
plt.grid()
plt.savefig('output_data/Lat_vs_Cloud.png')
plt.show()
```



Latitude vs. Wind Speed Plot

In [11]:

```
plt.scatter(weather_df['Latitude'],weather_df['WindSpeed'],edgecolors='black',co
plt.title('City Latitude vs Windspeed' + " (" + datetime.date.today().strftime("
plt.xlabel('Latitude')
plt.ylabel('Windspeed (mph)')
plt.grid()
plt.savefig('output_data/Lat_vs_Wind.png')
plt.show()
```


City Latitude vs Windspeed (July 03, 2019)
Windspeed (mph)
Latitude