In [10]:
```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import seaborn as sns
import requests
import time
import urllib
from citipy import citipy
output_data_file = "cities.csv"
lat_range = (-90, 90)
lng_range = (-180, 180)
```

In [11]:
```
lat_lngs = []
cities = []


lats = np.random.uniform(low=-90.000, high=90.000, size=1500)
lngs = np.random.uniform(low=-180.000, high=180.000, size=1500)
lat_lngs = zip(lats, lngs)


for lat_lng in lat_lngs:
    city = citipy.nearest_city(lat_lng[0], lat_lng[1]).city_name


    if city not in cities:
        cities.append(city)


len(cities)
```

Out[11]: 593

In [12]:

```
api_key = "924783bda048569443e49dd6a03e5591"


url = "http://api.openweathermap.org/data/2.5/weather?units=Imperial&APPID=


city_data = []


print("Beginning Data Retrieval     ")
print("-----------------------------")


record_count = 1
set_count = 1


for i, city in enumerate(cities):


    if (i % 50 == 0 and i >= 50):
        set_count += 1
        record_count = 0


    city_url = url + "&q=" + urllib.request.pathname2url(city)


    print("Processing Record %s of Set %s | %s" % (record_count, set_count,
    print(city_url)


    record_count += 1


    try:

        city_weather = requests.get(city_url).json()


        city_lat = city_weather["coord"]["lat"]
        city_lng = city_weather["coord"]["lon"]
        city_max_temp = city_weather["main"]["temp_max"]
        city_humidity = city_weather["main"]["humidity"]
        city_clouds = city_weather["clouds"]["all"]
        city_wind = city_weather["wind"]["speed"]
        city_country = city_weather["sys"]["country"]
        city_date = city_weather["dt"]


        city_data.append({"City": city,
                          "Lat": city_lat,
                          "Lng": city_lng,
                          "Max Temp": city_max_temp,
                          "Humidity": city_humidity,
```

```
                            "Cloudiness": city_clouds,
                            "Wind Speed": city_wind,
                            "Country": city_country,
                            "Date": city_date})


    except:
        print("City not found. Skipping...")
        pass


print("-----------------------------")
print("Data Retrieval Complete      ")
print("-----------------------------")
```

```
Beginning Data Retrieval
-----------------------------
Processing Record 1 of Set 1 | leningradskiy
http://api.openweathermap.org/data/2.5/weather?units=Imperial&APPID=924
783bda048569443e49dd6a03e5591&q=leningradskiy (http://api.openweatherma
p.org/data/2.5/weather?units=Imperial&APPID=924783bda048569443e49dd6a03
e5591&q=leningradskiy)
Processing Record 2 of Set 1 | rikitea
http://api.openweathermap.org/data/2.5/weather?units=Imperial&APPID=924
783bda048569443e49dd6a03e5591&q=rikitea (http://api.openweathermap.org/
data/2.5/weather?units=Imperial&APPID=924783bda048569443e49dd6a03e5591&
q=rikitea)
Processing Record 3 of Set 1 | egvekinot
http://api.openweathermap.org/data/2.5/weather?units=Imperial&APPID=924
783bda048569443e49dd6a03e5591&q=egvekinot (http://api.openweathermap.or
g/data/2.5/weather?units=Imperial&APPID=924783bda048569443e49dd6a03e559
1&q=egvekinot)
Processing Record 4 of Set 1 | cape town
http://api.openweathermap.org/data/2.5/weather?units=Imperial&APPID=924
```

In [13]:
```
# Convert array of JSONs into Pandas DataFrame
city_data_pd = pd.DataFrame(city_data)

# Extract relevant fields from the data frame
lats = city_data_pd["Lat"]
max_temps = city_data_pd["Max Temp"]
humidity = city_data_pd["Humidity"]
cloudiness = city_data_pd["Cloudiness"]
wind_speed = city_data_pd["Wind Speed"]


city_data_pd.to_csv(output_data_file, index_label="City_ID")

# Show Record Count
city_data_pd.count()
```

Out[13]:
```
City          534
Cloudiness    534
Country       534
Date          534
Humidity      534
Lat           534
Lng           534
Max Temp      534
Wind Speed    534
dtype: int64
```

In [14]:
```
city_data_pd.head()
```

Out[14]:

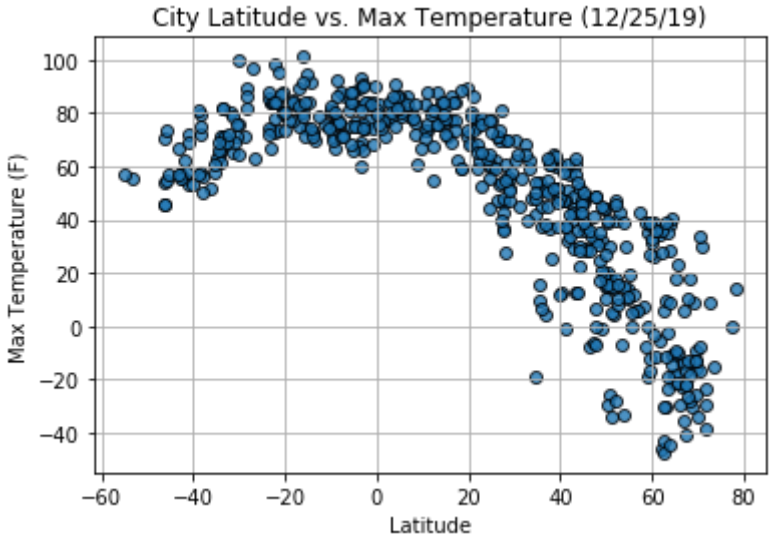
	City	Cloudiness	Country	Date	Humidity	Lat	Lng	Max Temp	Wind Speed
0	leningradskiy	100	RU	1577296858	92	69.38	178.42	-9.60	17.60
1	rikitea	100	PF	1577296780	76	-23.12	-134.97	73.83	5.50
2	egvekinot	23	RU	1577296832	93	66.32	-179.17	-22.77	6.80
3	cape town	40	ZA	1577296688	63	-33.93	18.42	66.00	13.87
4	cockburn town	89	TC	1577296844	74	21.46	-71.14	78.10	18.90

In [16]:
```
plt.scatter(lats,
            max_temps,
            edgecolor="black", linewidths=1, marker="o",
            alpha=0.8, label="Cities")


plt.title("City Latitude vs. Max Temperature (%s)" % time.strftime("%x"))
plt.ylabel("Max Temperature (F)")
plt.xlabel("Latitude")
plt.grid(True)


plt.savefig("Fig1.png")


plt.show()
```



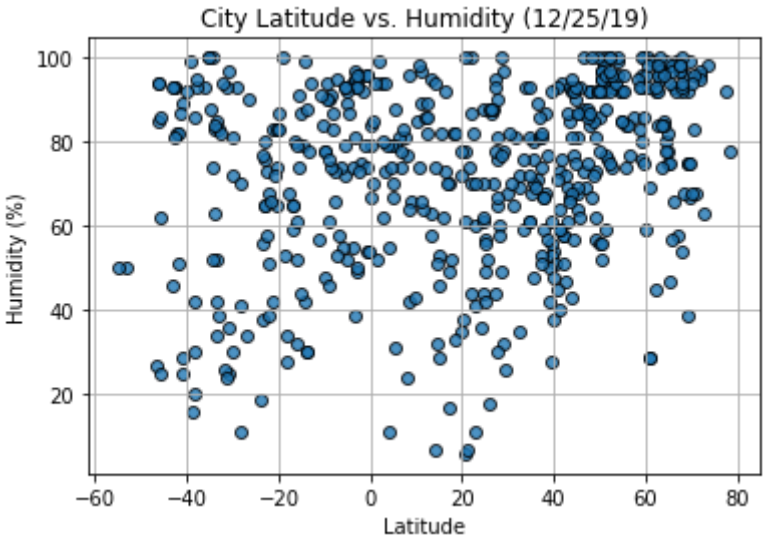
In [17]:

```
plt.scatter(lats,
            humidity,
            edgecolor="black", linewidths=1, marker="o",
            alpha=0.8, label="Cities")

plt.title("City Latitude vs. Humidity (%s)" % time.strftime("%x"))
plt.ylabel("Humidity (%)")
plt.xlabel("Latitude")
plt.grid(True)

plt.savefig("Fig2.png")


plt.show()
```

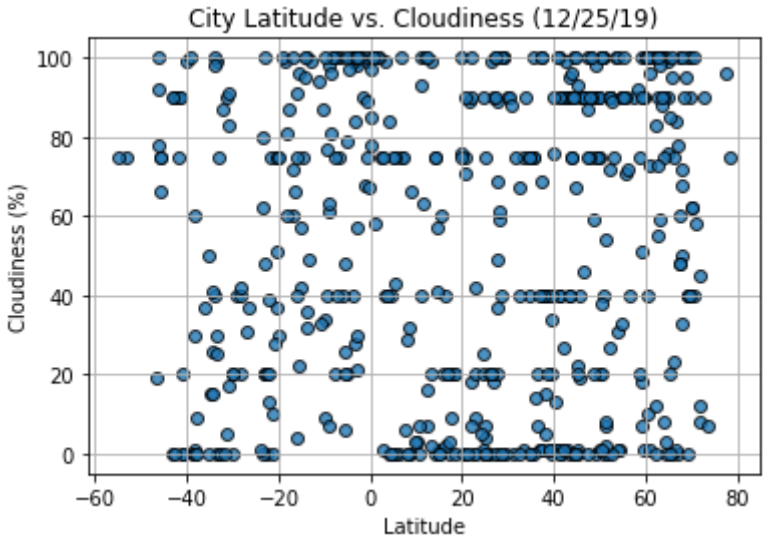


In [18]:
```
plt.scatter(lats,
            cloudiness,
            edgecolor="black", linewidths=1, marker="o",
            alpha=0.8, label="Cities")


plt.title("City Latitude vs. Cloudiness (%s)" % time.strftime("%x"))
plt.ylabel("Cloudiness (%)")
plt.xlabel("Latitude")
plt.grid(True)


plt.savefig("Fig3.png")


plt.show()
```



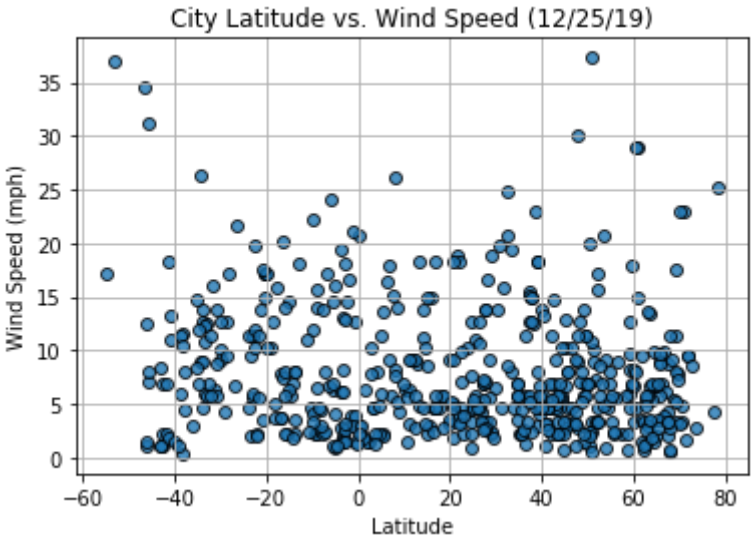
In [19]:

```
plt.scatter(lats,
            wind_speed,
            edgecolor="black", linewidths=1, marker="o",
            alpha=0.8, label="Cities")


plt.title("City Latitude vs. Wind Speed (%s)" % time.strftime("%x"))
plt.ylabel("Wind Speed (mph)")
plt.xlabel("Latitude")
plt.grid(True)


plt.savefig("Fig4.png")


plt.show()
```



In [ ]: