MSDS 610 - Week 5 Assignment - Jeremy Beard

Section I of MSDS 610 Week 5 Assignment: Weather data and MongoDB

The first part of the MSDS 610 Week 5 assignment involves querying data from a weather-related api and placing it into a MongoDB database. This part of the assignment used to utilize a different weather API for the data (metaweather) but this has been said to not be functional anymore, hence the weatherapi weather API resource.

In [1]:
```
import requests as req
from time import sleep
import pandas as pd
```

In [2]:
```
api_key = 'eddd6b8fc6ed4e7db5c170905220306'
params = {'key':api_key, 'q':'London'}
response = req.get('http://api.weatherapi.com/v1/current.json', params=params)
response.json()
```

Out[2]:
```
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
```

```
 'cloud': 100,
 'feelslike_c': 13.8,
 'feelslike_f': 56.9,
 'vis_km': 3.5,
 'vis_miles': 2.0,
 'uv': 3.0,
 'gust_mph': 5.6,
 'gust_kph': 9.0}}
```

Next we will put the queried data into a Pandas dataframe. This will only be one moment in time so far. We will expand upon this later.

In [3]:
```
df = pd.io.json.json_normalize(response.json())
df.head(5)
```

```
/home/jeremy/.local/lib/python3.6/site-packages/ipykernel_launcher.py:1: FutureWarning: pandas.io.json.json_nor
malize is deprecated, use pandas.json_normalize instead
  """Entry point for launching an IPython kernel.
```

Out[3]:

	location.name	location.region	location.country	location.lat	location.lon	location.tz_id	location.localtime_epoch	location.localtime	curre
0	London	City of London, Greater London	United Kingdom	51.52	-0.11	Europe/London	1654459153	2022-06-05 20:59	

1 rows × 33 columns

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1 entries, 0 to 0
Data columns (total 33 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   location.name               1 non-null      object
 1   location.region             1 non-null      object
 2   location.country            1 non-null      object
 3   location.lat                1 non-null      float64
 4   location.lon                1 non-null      float64
 5   location.tz_id              1 non-null      object
 6   location.localtime_epoch    1 non-null      int64
 7   location.localtime          1 non-null      object
 8   current.last_updated_epoch  1 non-null      int64
 9   current.last_updated        1 non-null      object
 10  current.temp_c              1 non-null      float64
 11  current.temp_f              1 non-null      float64
 12  current.is_day              1 non-null      int64
```

```
 13  current.condition.text      1 non-null      object
 14  current.condition.icon      1 non-null      object
 15  current.condition.code      1 non-null      int64
 16  current.wind_mph            1 non-null      float64
 17  current.wind_kph            1 non-null      float64
 18  current.wind_degree         1 non-null      int64
 19  current.wind_dir            1 non-null      object
 20  current.pressure_mb         1 non-null      float64
 21  current.pressure_in         1 non-null      float64
 22  current.precip_mm           1 non-null      float64
 23  current.precip_in           1 non-null      float64
 24  current.humidity            1 non-null      int64
 25  current.cloud               1 non-null      int64
 26  current.feelslike_c         1 non-null      float64
 27  current.feelslike_f         1 non-null      float64
 28  current.vis_km              1 non-null      float64
 29  current.vis_miles           1 non-null      float64
 30  current.uv                  1 non-null      float64
 31  current.gust_mph            1 non-null      float64
 32  current.gust_kph            1 non-null      float64
dtypes: float64(17), int64(7), object(9)
memory usage: 392.0+ bytes
```

As one can see above, there is one query containing many weather-related fields, shown above. Now we will move on to time-based querying where we will query 20 weather datapoints and place them in an array that we will call 'timebased'.

In [5]:
```
# do time based querying
timebased = []
counter = 0
time_to_sleep = 1
while(counter < 20):
    response = req.get('http://api.weatherapi.com/v1/current.json', params=params)
    timebased.append(response.json())
    sleep(time_to_sleep)
    counter += 1
```

In [6]:
```
timebased
```

Out[6]:
```
[{'location': {'name': 'London',
   'region': 'City of London, Greater London',
   'country': 'United Kingdom',
   'lat': 51.52,
   'lon': -0.11,
   'tz_id': 'Europe/London',
   'localtime_epoch': 1654459153,
```

```
 'localtime': '2022-06-05 20:59'},
'current': {'last_updated_epoch': 1654458300,
 'last_updated': '2022-06-05 20:45',
 'temp_c': 14.0,
 'temp_f': 57.2,
 'is_day': 1,
 'condition': {'text': 'Light rain shower',
  'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
  'code': 1240},
 'wind_mph': 3.8,
 'wind_kph': 6.1,
 'wind_degree': 320,
 'wind_dir': 'NW',
 'pressure_mb': 1015.0,
 'pressure_in': 29.97,
 'precip_mm': 0.0,
 'precip_in': 0.0,
 'humidity': 94,
 'cloud': 100,
 'feelslike_c': 13.8,
 'feelslike_f': 56.9,
 'vis_km': 3.5,
 'vis_miles': 2.0,
 'uv': 3.0,
 'gust_mph': 5.6,
 'gust_kph': 9.0}},
{'location': {'name': 'London',
 'region': 'City of London, Greater London',
 'country': 'United Kingdom',
 'lat': 51.52,
 'lon': -0.11,
 'tz_id': 'Europe/London',
 'localtime_epoch': 1654459153,
 'localtime': '2022-06-05 20:59'},
'current': {'last_updated_epoch': 1654458300,
 'last_updated': '2022-06-05 20:45',
 'temp_c': 14.0,
 'temp_f': 57.2,
 'is_day': 1,
 'condition': {'text': 'Light rain shower',
  'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
  'code': 1240},
 'wind_mph': 3.8,
 'wind_kph': 6.1,
 'wind_degree': 320,
 'wind_dir': 'NW',
 'pressure_mb': 1015.0,
 'pressure_in': 29.97,
 'precip_mm': 0.0,
```

```
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
```

```
 'localtime_epoch': 1654459153,
 'localtime': '2022-06-05 20:59'},
'current': {'last_updated_epoch': 1654458300,
 'last_updated': '2022-06-05 20:45',
 'temp_c': 14.0,
 'temp_f': 57.2,
 'is_day': 1,
 'condition': {'text': 'Light rain shower',
  'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
  'code': 1240},
 'wind_mph': 3.8,
 'wind_kph': 6.1,
 'wind_degree': 320,
 'wind_dir': 'NW',
 'pressure_mb': 1015.0,
 'pressure_in': 29.97,
 'precip_mm': 0.0,
 'precip_in': 0.0,
 'humidity': 94,
 'cloud': 100,
 'feelslike_c': 13.8,
 'feelslike_f': 56.9,
 'vis_km': 3.5,
 'vis_miles': 2.0,
 'uv': 3.0,
 'gust_mph': 5.6,
 'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
```

```
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
```

```
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
```

```
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
```

```
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
```

```
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
```

```
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
```

```
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
```

```
 'country': 'United Kingdom',
 'lat': 51.52,
 'lon': -0.11,
 'tz_id': 'Europe/London',
 'localtime_epoch': 1654459153,
 'localtime': '2022-06-05 20:59'},
'current': {'last_updated_epoch': 1654458300,
 'last_updated': '2022-06-05 20:45',
 'temp_c': 14.0,
 'temp_f': 57.2,
 'is_day': 1,
 'condition': {'text': 'Light rain shower',
  'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
  'code': 1240},
 'wind_mph': 3.8,
 'wind_kph': 6.1,
 'wind_degree': 320,
 'wind_dir': 'NW',
 'pressure_mb': 1015.0,
 'pressure_in': 29.97,
 'precip_mm': 0.0,
 'precip_in': 0.0,
 'humidity': 94,
 'cloud': 100,
 'feelslike_c': 13.8,
 'feelslike_f': 56.9,
 'vis_km': 3.5,
 'vis_miles': 2.0,
 'uv': 3.0,
 'gust_mph': 5.6,
 'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
```

```
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
```

```
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
  'wind_kph': 6.1,
  'wind_degree': 320,
  'wind_dir': 'NW',
  'pressure_mb': 1015.0,
  'pressure_in': 29.97,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 94,
  'cloud': 100,
  'feelslike_c': 13.8,
  'feelslike_f': 56.9,
  'vis_km': 3.5,
  'vis_miles': 2.0,
  'uv': 3.0,
  'gust_mph': 5.6,
  'gust_kph': 9.0}},
{'location': {'name': 'London',
  'region': 'City of London, Greater London',
  'country': 'United Kingdom',
  'lat': 51.52,
  'lon': -0.11,
  'tz_id': 'Europe/London',
  'localtime_epoch': 1654459153,
  'localtime': '2022-06-05 20:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 20:45',
  'temp_c': 14.0,
  'temp_f': 57.2,
  'is_day': 1,
  'condition': {'text': 'Light rain shower',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
   'code': 1240},
  'wind_mph': 3.8,
```

```
'wind_kph': 6.1,
'wind_degree': 320,
'wind_dir': 'NW',
'pressure_mb': 1015.0,
'pressure_in': 29.97,
'precip_mm': 0.0,
'precip_in': 0.0,
'humidity': 94,
'cloud': 100,
'feelslike_c': 13.8,
'feelslike_f': 56.9,
'vis_km': 3.5,
'vis_miles': 2.0,
'uv': 3.0,
'gust_mph': 5.6,
'gust_kph': 9.0}}]
```

In [7]:

```
df = pd.json_normalize(timebased)
df.head(5)
```

Out[7]:

	location.name	location.region	location.country	location.lat	location.lon	location.tz_id	location.localtime_epoch	location.localtime	curre
0	London	City of London, Greater London	United Kingdom	51.52	-0.11	Europe/London	1654459153	2022-06-05 20:59	
1	London	City of London, Greater London	United Kingdom	51.52	-0.11	Europe/London	1654459153	2022-06-05 20:59	
2	London	City of London, Greater London	United Kingdom	51.52	-0.11	Europe/London	1654459153	2022-06-05 20:59	
3	London	City of London, Greater London	United Kingdom	51.52	-0.11	Europe/London	1654459153	2022-06-05 20:59	
4	London	City of London, Greater London	United Kingdom	51.52	-0.11	Europe/London	1654459153	2022-06-05 20:59	

5 rows × 33 columns

In [8]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 33 columns):
 #   Column                      Non-Null Count  Dtype
```

```
 ---  ------                       --------------  -----
  0   location.name               20 non-null     object
  1   location.region             20 non-null     object
  2   location.country            20 non-null     object
  3   location.lat                20 non-null     float64
  4   location.lon                20 non-null     float64
  5   location.tz_id              20 non-null     object
  6   location.localtime_epoch    20 non-null     int64
  7   location.localtime          20 non-null     object
  8   current.last_updated_epoch  20 non-null     int64
  9   current.last_updated        20 non-null     object
  10  current.temp_c              20 non-null     float64
  11  current.temp_f              20 non-null     float64
  12  current.is_day              20 non-null     int64
  13  current.condition.text      20 non-null     object
  14  current.condition.icon      20 non-null     object
  15  current.condition.code      20 non-null     int64
  16  current.wind_mph            20 non-null     float64
  17  current.wind_kph            20 non-null     float64
  18  current.wind_degree         20 non-null     int64
  19  current.wind_dir            20 non-null     object
  20  current.pressure_mb         20 non-null     float64
  21  current.pressure_in         20 non-null     float64
  22  current.precip_mm           20 non-null     float64
  23  current.precip_in           20 non-null     float64
  24  current.humidity            20 non-null     int64
  25  current.cloud               20 non-null     int64
  26  current.feelslike_c         20 non-null     float64
  27  current.feelslike_f         20 non-null     float64
  28  current.vis_km              20 non-null     float64
  29  current.vis_miles           20 non-null     float64
  30  current.uv                  20 non-null     float64
  31  current.gust_mph            20 non-null     float64
  32  current.gust_kph            20 non-null     float64
dtypes: float64(17), int64(7), object(9)
memory usage: 5.3+ KB
```

As one can see above, there are now 20 datapoints containing the same weather-related data as the first section of the assignment. We've now accumulated multiple datapoints related to multiple points in time.

Next we will query multiple locations and get one weather datapoint from each location. We will use London, Berlin, and Denver for our locations.

In [9]:

```
#location-based approach to querying weather
locations = ['London', 'Berlin', 'Denver']
location_weather = []
for location in locations:
```

```
    params = {'key':api_key, 'q':location}
    response = req.get('http://api.weatherapi.com/v1/current.json', params=params)
    location_weather.append(response.json())

location_weather
```

Out[9]:


```
[{'location': {'name': 'London',
   'region': 'City of London, Greater London',
   'country': 'United Kingdom',
   'lat': 51.52,
   'lon': -0.11,
   'tz_id': 'Europe/London',
   'localtime_epoch': 1654459153,
   'localtime': '2022-06-05 20:59'},
  'current': {'last_updated_epoch': 1654458300,
   'last_updated': '2022-06-05 20:45',
   'temp_c': 14.0,
   'temp_f': 57.2,
   'is_day': 1,
   'condition': {'text': 'Light rain shower',
    'icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
    'code': 1240},
   'wind_mph': 3.8,
   'wind_kph': 6.1,
   'wind_degree': 320,
   'wind_dir': 'NW',
   'pressure_mb': 1015.0,
   'pressure_in': 29.97,
   'precip_mm': 0.0,
   'precip_in': 0.0,
   'humidity': 94,
   'cloud': 100,
   'feelslike_c': 13.8,
   'feelslike_f': 56.9,
   'vis_km': 3.5,
   'vis_miles': 2.0,
   'uv': 3.0,
   'gust_mph': 5.6,
   'gust_kph': 9.0}},
 {'location': {'name': 'Berlin',
   'region': 'Berlin',
   'country': 'Germany',
   'lat': 52.52,
   'lon': 13.4,
   'tz_id': 'Europe/Berlin',
   'localtime_epoch': 1654459176,
   'localtime': '2022-06-05 21:59'},
  'current': {'last_updated_epoch': 1654458300,
```

```
  'last_updated': '2022-06-05 21:45',
  'temp_c': 24.0,
  'temp_f': 75.2,
  'is_day': 0,
  'condition': {'text': 'Clear',
   'icon': '//cdn.weatherapi.com/weather/64x64/night/113.png',
   'code': 1000},
  'wind_mph': 8.1,
  'wind_kph': 13.0,
  'wind_degree': 90,
  'wind_dir': 'E',
  'pressure_mb': 1013.0,
  'pressure_in': 29.91,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 41,
  'cloud': 0,
  'feelslike_c': 25.1,
  'feelslike_f': 77.2,
  'vis_km': 10.0,
  'vis_miles': 6.0,
  'uv': 6.0,
  'gust_mph': 17.4,
  'gust_kph': 28.1}},
{'location': {'name': 'Denver',
  'region': 'Colorado',
  'country': 'United States of America',
  'lat': 39.74,
  'lon': -104.98,
  'tz_id': 'America/Denver',
  'localtime_epoch': 1654459176,
  'localtime': '2022-06-05 13:59'},
 'current': {'last_updated_epoch': 1654458300,
  'last_updated': '2022-06-05 13:45',
  'temp_c': 26.1,
  'temp_f': 79.0,
  'is_day': 1,
  'condition': {'text': 'Partly cloudy',
   'icon': '//cdn.weatherapi.com/weather/64x64/day/116.png',
   'code': 1003},
  'wind_mph': 8.1,
  'wind_kph': 13.0,
  'wind_degree': 160,
  'wind_dir': 'SSE',
  'pressure_mb': 1011.0,
  'pressure_in': 29.85,
  'precip_mm': 0.0,
  'precip_in': 0.0,
  'humidity': 26,
```

```
 'cloud': 75,
 'feelslike_c': 24.5,
 'feelslike_f': 76.2,
 'vis_km': 16.0,
 'vis_miles': 9.0,
 'uv': 7.0,
 'gust_mph': 7.2,
 'gust_kph': 11.5}}]
```

In [10]:

```
df = pd.json_normalize(location_weather)
df.head(5)
```

Out[10]:

	location.name	location.region	location.country	location.lat	location.lon	location.tz_id	location.localtime_epoch	location.localtime	curre
0	London	City of London, Greater London	United Kingdom	51.52	-0.11	Europe/London	1654459153	2022-06-05 20:59	
1	Berlin	Berlin	Germany	52.52	13.40	Europe/Berlin	1654459176	2022-06-05 21:59	
2	Denver	Colorado	United States of America	39.74	-104.98	America/Denver	1654459176	2022-06-05 13:59	

3 rows × 33 columns

In [11]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 33 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   location.name                3 non-null      object
 1   location.region              3 non-null      object
 2   location.country             3 non-null      object
 3   location.lat                 3 non-null      float64
 4   location.lon                 3 non-null      float64
 5   location.tz_id               3 non-null      object
 6   location.localtime_epoch     3 non-null      int64
 7   location.localtime           3 non-null      object
 8   current.last_updated_epoch   3 non-null      int64
 9   current.last_updated         3 non-null      object
 10  current.temp_c               3 non-null      float64
 11  current.temp_f               3 non-null      float64
 12  current.is_day               3 non-null      int64
 13  current.condition.text       3 non-null      object
```

```
 14  current.condition.icon      3 non-null      object
 15  current.condition.code      3 non-null      int64
 16  current.wind_mph            3 non-null      float64
 17  current.wind_kph            3 non-null      float64
 18  current.wind_degree         3 non-null      int64
 19  current.wind_dir            3 non-null      object
 20  current.pressure_mb         3 non-null      float64
 21  current.pressure_in         3 non-null      float64
 22  current.precip_mm           3 non-null      float64
 23  current.precip_in           3 non-null      float64
 24  current.humidity            3 non-null      int64
 25  current.cloud               3 non-null      int64
 26  current.feelslike_c         3 non-null      float64
 27  current.feelslike_f         3 non-null      float64
 28  current.vis_km              3 non-null      float64
 29  current.vis_miles           3 non-null      float64
 30  current.uv                  3 non-null      float64
 31  current.gust_mph            3 non-null      float64
 32  current.gust_kph            3 non-null      float64
dtypes: float64(17), int64(7), object(9)
memory usage: 920.0+ bytes
```

Now that we've experimented with query the weather data multiple ways, we will open a MongoClient() and insert our weather data into a MongoDB collection.

In [12]:
```
from pymongo import MongoClient
```

In [13]:
```
client = MongoClient()
db = client['weather']
collection = db['world']
df.to_dict('records')
```

Out[13]:
```
[{'location.name': 'London',
  'location.region': 'City of London, Greater London',
  'location.country': 'United Kingdom',
  'location.lat': 51.52,
  'location.lon': -0.11,
  'location.tz_id': 'Europe/London',
  'location.localtime_epoch': 1654459153,
  'location.localtime': '2022-06-05 20:59',
  'current.last_updated_epoch': 1654458300,
  'current.last_updated': '2022-06-05 20:45',
  'current.temp_c': 14.0,
  'current.temp_f': 57.2,
  'current.is_day': 1,
```

```
 'current.condition.text': 'Light rain shower',
 'current.condition.icon': '//cdn.weatherapi.com/weather/64x64/day/353.png',
 'current.condition.code': 1240,
 'current.wind_mph': 3.8,
 'current.wind_kph': 6.1,
 'current.wind_degree': 320,
 'current.wind_dir': 'NW',
 'current.pressure_mb': 1015.0,
 'current.pressure_in': 29.97,
 'current.precip_mm': 0.0,
 'current.precip_in': 0.0,
 'current.humidity': 94,
 'current.cloud': 100,
 'current.feelslike_c': 13.8,
 'current.feelslike_f': 56.9,
 'current.vis_km': 3.5,
 'current.vis_miles': 2.0,
 'current.uv': 3.0,
 'current.gust_mph': 5.6,
 'current.gust_kph': 9.0},
{'location.name': 'Berlin',
 'location.region': 'Berlin',
 'location.country': 'Germany',
 'location.lat': 52.52,
 'location.lon': 13.4,
 'location.tz_id': 'Europe/Berlin',
 'location.localtime_epoch': 1654459176,
 'location.localtime': '2022-06-05 21:59',
 'current.last_updated_epoch': 1654458300,
 'current.last_updated': '2022-06-05 21:45',
 'current.temp_c': 24.0,
 'current.temp_f': 75.2,
 'current.is_day': 0,
 'current.condition.text': 'Clear',
 'current.condition.icon': '//cdn.weatherapi.com/weather/64x64/night/113.png',
 'current.condition.code': 1000,
 'current.wind_mph': 8.1,
 'current.wind_kph': 13.0,
 'current.wind_degree': 90,
 'current.wind_dir': 'E',
 'current.pressure_mb': 1013.0,
 'current.pressure_in': 29.91,
 'current.precip_mm': 0.0,
 'current.precip_in': 0.0,
 'current.humidity': 41,
 'current.cloud': 0,
 'current.feelslike_c': 25.1,
 'current.feelslike_f': 77.2,
 'current.vis_km': 10.0,
```

```
 'current.vis_miles': 6.0,
 'current.uv': 6.0,
 'current.gust_mph': 17.4,
 'current.gust_kph': 28.1},
{'location.name': 'Denver',
 'location.region': 'Colorado',
 'location.country': 'United States of America',
 'location.lat': 39.74,
 'location.lon': -104.98,
 'location.tz_id': 'America/Denver',
 'location.localtime_epoch': 1654459176,
 'location.localtime': '2022-06-05 13:59',
 'current.last_updated_epoch': 1654458300,
 'current.last_updated': '2022-06-05 13:45',
 'current.temp_c': 26.1,
 'current.temp_f': 79.0,
 'current.is_day': 1,
 'current.condition.text': 'Partly cloudy',
 'current.condition.icon': '//cdn.weatherapi.com/weather/64x64/day/116.png',
 'current.condition.code': 1003,
 'current.wind_mph': 8.1,
 'current.wind_kph': 13.0,
 'current.wind_degree': 160,
 'current.wind_dir': 'SSE',
 'current.pressure_mb': 1011.0,
 'current.pressure_in': 29.85,
 'current.precip_mm': 0.0,
 'current.precip_in': 0.0,
 'current.humidity': 26,
 'current.cloud': 75,
 'current.feelslike_c': 24.5,
 'current.feelslike_f': 76.2,
 'current.vis_km': 16.0,
 'current.vis_miles': 9.0,
 'current.uv': 7.0,
 'current.gust_mph': 7.2,
 'current.gust_kph': 11.5}]
```

In [14]:
```
collection.insert_many(df.to_dict('records'))
```

Out[14]:
```
<pymongo.results.InsertManyResult at 0x7fd918156d30>
```

In [15]:
```
collection.find_one()
```

Out[15]:
```
{'_id': ObjectId('629d06fe35732c403dc52aad'),
 'location.name': 'London',
```

```
'location.region': 'City of London, Greater London',
'location.country': 'United Kingdom',
'location.lat': 51.52,
'location.lon': -0.11,
'location.tz_id': 'Europe/London',
'location.localtime_epoch': 1654457314,
'location.localtime': '2022-06-05 20:28',
'current.last_updated_epoch': 1654456500,
'current.last_updated': '2022-06-05 20:15',
'current.temp_c': 14.0,
'current.temp_f': 57.2,
'current.is_day': 1,
'current.condition.text': 'Light rain',
'current.condition.icon': '//cdn.weatherapi.com/weather/64x64/day/296.png',
'current.condition.code': 1183,
'current.wind_mph': 3.8,
'current.wind_kph': 6.1,
'current.wind_degree': 320,
'current.wind_dir': 'NW',
'current.pressure_mb': 1015.0,
'current.pressure_in': 29.97,
'current.precip_mm': 0.0,
'current.precip_in': 0.0,
'current.humidity': 94,
'current.cloud': 75,
'current.feelslike_c': 13.8,
'current.feelslike_f': 56.9,
'current.vis_km': 7.0,
'current.vis_miles': 4.0,
'current.uv': 3.0,
'current.gust_mph': 5.6,
'current.gust_kph': 9.0}
```


Section II of MSDS 610 Week 5 Assignment: MISO Data and PostgreSQL

Now we can move onto the next part of the assignment, pulling data using an API and placing it in a PostgreSQL database. The example I was following from the MSDS 610 Week 5 Lab uses data concerning wind energy. I will be pulling from MISO but will instead be using data concerning solar energy.

In [25]:
```
#get data from miso energy and put in database using postgres
#response = req.get('https://api.misoenergy.org/MISORTWDDataBroker/DataBrokerServices.asmx?messageType=getWind
response = req.get('https://api.misoenergy.org/MISORTWDDataBroker/DataBrokerServices.asmx?messageType=getSolar
response.json()
```

Out[25]:

```
{'MktDay': '06-05-2022',
 'RefId': '05-Jun-2022 - Interval 15:00 EST',
```

```
 'instance': [{'DateTimeEST': '2022-06-05 12:00:00 AM',
   'HourEndingEST': '1',
   'Value': '-2.16'},
  {'DateTimeEST': '2022-06-05 1:00:00 AM',
   'HourEndingEST': '2',
   'Value': '-2.20'},
  {'DateTimeEST': '2022-06-05 2:00:00 AM',
   'HourEndingEST': '3',
   'Value': '-2.20'},
  {'DateTimeEST': '2022-06-05 3:00:00 AM',
   'HourEndingEST': '4',
   'Value': '-2.20'},
  {'DateTimeEST': '2022-06-05 4:00:00 AM',
   'HourEndingEST': '5',
   'Value': '-2.24'},
  {'DateTimeEST': '2022-06-05 5:00:00 AM',
   'HourEndingEST': '6',
   'Value': '2.15'},
  {'DateTimeEST': '2022-06-05 6:00:00 AM',
   'HourEndingEST': '7',
   'Value': '158.38'},
  {'DateTimeEST': '2022-06-05 7:00:00 AM',
   'HourEndingEST': '8',
   'Value': '504.37'},
  {'DateTimeEST': '2022-06-05 8:00:00 AM',
   'HourEndingEST': '9',
   'Value': '738.92'},
  {'DateTimeEST': '2022-06-05 9:00:00 AM',
   'HourEndingEST': '10',
   'Value': '1053.08'},
  {'DateTimeEST': '2022-06-05 10:00:00 AM',
   'HourEndingEST': '11',
   'Value': '1335.39'},
  {'DateTimeEST': '2022-06-05 11:00:00 AM',
   'HourEndingEST': '12',
   'Value': '1441.66'},
  {'DateTimeEST': '2022-06-05 12:00:00 PM',
   'HourEndingEST': '13',
   'Value': '1472.70'},
  {'DateTimeEST': '2022-06-05 1:00:00 PM',
   'HourEndingEST': '14',
   'Value': '1343.87'},
  {'DateTimeEST': '2022-06-05 2:00:00 PM',
   'HourEndingEST': '15',
   'Value': '1221.13'}]}
```


After using requests to get the MISO data, we can query a few things about it to learn about the metadata.

In [18]:

```
# According to professor, supposed to authenticate for API, but doesn't seem to be the case
# https://www.misoenergy.org/markets-and-operations/notifications-overview/it-and-system-notifications/api/
response.headers
```

Out[18]:
```
{'Content-Length': '328', 'Content-Type': 'application/json', 'Date': 'Sun, 05 Jun 2022 20:07:27 GMT', 'Access-
Control-Allow-Headers': 'Content-Type, Origin, X-Requested-With, Content-Type, Accept', 'Access-Control-Allow-M
ethods': 'PUT', 'Access-Control-Allow-Origin': '*', 'Cache-Control': 'no-cache, no-store, must-revalidate', 'Co
ntent-Encoding': 'gzip', 'Expires': '0', 'Pragma': 'no-cache', 'Set-Cookie': 'ASP.NET_SessionId=tr1fxlkhedkuxhx
5sbgpjz4e; path=/; HttpOnly; SameSite=Lax'}
```

In [26]:
```
response.json().keys()
```

Out[26]:
```
dict_keys(['MktDay', 'RefId', 'instance'])
```

Next, we will then put the data that we queried using requests, into a Pandas dataframe using a for loop.

In [27]:
```
json_data = response.json()['instance']
df = pd.io.json.json_normalize(json_data[0])
for data in json_data[1:]:
    df = df.append(pd.io.json.json_normalize(data))
df
```

```
/home/jeremy/.local/lib/python3.6/site-packages/ipykernel_launcher.py:2: FutureWarning: pandas.io.json.json_nor
malize is deprecated, use pandas.json_normalize instead

/home/jeremy/.local/lib/python3.6/site-packages/ipykernel_launcher.py:4: FutureWarning: pandas.io.json.json_nor
malize is deprecated, use pandas.json_normalize instead
  after removing the cwd from sys.path.
```

Out[27]:

	DateTimeEST	HourEndingEST	Value
0	2022-06-05 12:00:00 AM	1	-2.16
0	2022-06-05 1:00:00 AM	2	-2.20
0	2022-06-05 2:00:00 AM	3	-2.20
0	2022-06-05 3:00:00 AM	4	-2.20
0	2022-06-05 4:00:00 AM	5	-2.24
0	2022-06-05 5:00:00 AM	6	2.15
0	2022-06-05 6:00:00 AM	7	158.38
0	2022-06-05 7:00:00 AM	8	504.37

	DateTimeEST	HourEndingEST	Value
0	2022-06-05 8:00:00 AM	9	738.92
0	2022-06-05 9:00:00 AM	10	1053.08
0	2022-06-05 10:00:00 AM	11	1335.39
0	2022-06-05 11:00:00 AM	12	1441.66
0	2022-06-05 12:00:00 PM	13	1472.70
0	2022-06-05 1:00:00 PM	14	1343.87
0	2022-06-05 2:00:00 PM	15	1221.13

In [20]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 15 entries, 0 to 0
Data columns (total 3 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   DateTimeEST    15 non-null     object
 1   HourEndingEST  15 non-null     object
 2   Value          15 non-null     object
dtypes: object(3)
memory usage: 480.0+ bytes
```

Since the DateTimeEST column is only a string right now, we will convert it to a datetime object using the pd.to_datetime function.

In [21]:
```
df['DateTimeEST'] = pd.to_datetime(df['DateTimeEST']).dt.tz_localize('US/Eastern').astype('object')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 15 entries, 0 to 0
Data columns (total 3 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   DateTimeEST    15 non-null     object
 1   HourEndingEST  15 non-null     object
 2   Value          15 non-null     object
dtypes: object(3)
memory usage: 480.0+ bytes
```

In [22]:
```
df
```

Out[22]:

	DateTimeEST	HourEndingEST	Value
0	2022-06-05 00:00:00-04:00	1	6367.35
0	2022-06-05 01:00:00-04:00	2	5692.94
0	2022-06-05 02:00:00-04:00	3	5215.71
0	2022-06-05 03:00:00-04:00	4	5354.81
0	2022-06-05 04:00:00-04:00	5	4809.61
0	2022-06-05 05:00:00-04:00	6	4369.86
0	2022-06-05 06:00:00-04:00	7	4148.82
0	2022-06-05 07:00:00-04:00	8	3102.38
0	2022-06-05 08:00:00-04:00	9	2166.20
0	2022-06-05 09:00:00-04:00	10	2282.38
0	2022-06-05 10:00:00-04:00	11	3034.73
0	2022-06-05 11:00:00-04:00	12	3769.74
0	2022-06-05 12:00:00-04:00	13	4246.14
0	2022-06-05 13:00:00-04:00	14	4685.81
0	2022-06-05 14:00:00-04:00	15	5399.81

Before doing the next step, we need to create a MISO database in our PostgreSQL server. Screenshots have been provided for this step in the MSDS610-JeremyBeard-Week5 docx or pdf assignment submission. The "CREATE DATABASE miso;" command was used. Then the next step in this Jupyter notebook was executed, connecting to the database and then creating a table called "rt_solar" which will eventually contain the solar energy data.

In [30]:

```
# password needs to be set for the postgres user, or another user added for the db
# https://stackoverflow.com/a/12721095/4549682
import psycopg2
conn = psycopg2.connect("dbname=miso host=localhost user=postgres password=postgres")
cursor = conn.cursor() # create cursor, interface into database

# need to specify schema on table creation
# https://www.postgresql.org/docs/10/datatype-numeric.html
# https://www.postgresql.org/docs/10/datatype-datetime.html
cursor.execute('CREATE TABLE IF NOT EXISTS rt_solar (datetime TIMESTAMP WITH TIME ZONE, value NUMERIC);')
conn.commit()  # required to actually execute statements
```

```
conn.close()
# need to use sqlalchemy for easy interface with pandas
# https://stackoverflow.com/a/42587012/4549682
```

In [32]:
```
from sqlalchemy import create_engine
from sqlalchemy.types import TIMESTAMP, NUMERIC
engine = create_engine('postgresql://postgres:postgres@localhost:5432/miso')
```

In [34]:
```
df.columns = [d.lower() for d in df.columns]
df['datetime'] = df['datetimeest']
```

In [35]:
```
df
```

Out[35]:

	datetimeest	hourendingest	value	datetime
0	2022-06-05 12:00:00 AM	1	-2.16	2022-06-05 12:00:00 AM
0	2022-06-05 1:00:00 AM	2	-2.20	2022-06-05 1:00:00 AM
0	2022-06-05 2:00:00 AM	3	-2.20	2022-06-05 2:00:00 AM
0	2022-06-05 3:00:00 AM	4	-2.20	2022-06-05 3:00:00 AM
0	2022-06-05 4:00:00 AM	5	-2.24	2022-06-05 4:00:00 AM
0	2022-06-05 5:00:00 AM	6	2.15	2022-06-05 5:00:00 AM
0	2022-06-05 6:00:00 AM	7	158.38	2022-06-05 6:00:00 AM
0	2022-06-05 7:00:00 AM	8	504.37	2022-06-05 7:00:00 AM
0	2022-06-05 8:00:00 AM	9	738.92	2022-06-05 8:00:00 AM
0	2022-06-05 9:00:00 AM	10	1053.08	2022-06-05 9:00:00 AM
0	2022-06-05 10:00:00 AM	11	1335.39	2022-06-05 10:00:00 AM
0	2022-06-05 11:00:00 AM	12	1441.66	2022-06-05 11:00:00 AM
0	2022-06-05 12:00:00 PM	13	1472.70	2022-06-05 12:00:00 PM
0	2022-06-05 1:00:00 PM	14	1343.87	2022-06-05 1:00:00 PM
0	2022-06-05 2:00:00 PM	15	1221.13	2022-06-05 2:00:00 PM

We then use the to_sql function to fill the rt_solar table with the MISO data.

In [36]:
```
df[['datetime', 'value']].to_sql(name='rt_solar',
                                 con=engine,
                                 if_exists='append',
                                 index=False,
                                 dtype={'datetime':TIMESTAMP(timezone=True),
                                        'value': NUMERIC})
```

This assignment was interesting! I enjoyed revisiting both MongoDB and PostgreSQL and I hope that I will continue to use both of these tools as we progress throughout this course. Both seem like very useful tools that can be utilized effectively in certain circumstances, especially in unstructured vs. structure data circumstances. NoSQL (MongoDB, in this case) is a much more appropriate choice for unstructured data.

Thank you! Jeremy Beard

In [ ]:

In [ ]:

In [ ]:

In [ ]: