**VacationPy**

**Note**

* Instructions have been included for each segment. You do not have to follow them exactly, but they are included to help you think through the steps.

# Dependencies and Setup

import matplotlib.pyplot as plt

import pandas as pd

import numpy as np

import requests

import gmaps

import os

# Import API key

from api\_keys import g\_key

**Store Part I results into DataFrame**

* Load the csv exported in Part I to a DataFrame

**Humidity Heatmap**

* Configure gmaps.
* Use the Lat and Lng as locations and Humidity as the weight.
* Add Heatmap layer to map.

**Create new DataFrame fitting weather criteria**

* Narrow down the cities to fit weather conditions.
* Drop any rows will null values.

**Hotel Map**

* Store into variable named hotel\_df.
* Add a "Hotel Name" column to the DataFrame.
* Set parameters to search for hotels with 5000 meters.
* Hit the Google Places API for each city's coordinates.
* Store the first Hotel result into the DataFrame.
* Plot markers on top of the heatmap.

# NOTE: Do not change any of the code in this cell

# Using the template add the hotel marks to the heatmap

info\_box\_template = """

<dl>

<dt>Name</dt><dd>{Hotel Name}</dd>

<dt>City</dt><dd>{City}</dd>

<dt>Country</dt><dd>{Country}</dd>

</dl>

"""

# Store the DataFrame Row

# NOTE: be sure to update with your DataFrame name

hotel\_info = [info\_box\_template.format(\*\*row) for index, row in hotel\_df.iterrows()]

locations = hotel\_df[["Lat", "Lng"]]

# Add marker layer ontop of heat map

# Display figure

WeatherPy

Note

Instructions have been included for each segment. You do not have to follow them exactly, but they are included to help you think through the steps.

In [1]: # Dependencies and Setup

import matplotlib.pyplot as plt

import pandas as pd

import numpy as np

import requests

import time

from scipy.stats import linregress

# Import API key

from api\_keys import weather\_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 Cities List

In [2]: # List for holding lat\_lngs and cities

lat\_lngs = []

cities = []

# Create a set of random lat and lng combinations

lats = np.random.uniform(lat\_range[0], lat\_range[1], size=1500)

lngs = np.random.uniform(lng\_range[0], lng\_range[1], size=1500)

lat\_lngs = zip(lats, lngs)

# Identify nearest city for each lat, lng combination

for lat\_lng in lat\_lngs:

city = citipy.nearest\_city(lat\_lng[0], lat\_lng[1]).city\_name

# If the city is unique, then add it to a our cities list

if city not in cities:

cities.append(city)

# Print the city count to confirm sufficient count

len(cities) Out [2]: 603 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 [3]:

# OpenWeatherMap API Key

weather\_api\_key = "YOUR KEY HERE!"

# Google API Key

g\_key = "YOUR KEY HERE!"