



IBM DATA SCIENCE PROFESSIONAL CERTIFICATION

CAPSTONE PROJECT ON

**COMMUNITY MAPPING WITH GEOGRAPHICAL
COORDINATES USING FOURSQUARE RESOURCES**

BY

IDOKO JOB JOHN

April 2020

OUTLINE

- Introduction/Business Problem
- Method of Study
 - Importing Required Python Libraries
 - Data Acquisition and Processing
 - Data Presentation
 - Generating a Community Map of Staten Island
- Result Presentation, Analysis and Discussion

INTRODUCTION/BUSINESS PROBLEM

- Community mapping is the creation of pictures of the geographical entities.
- Maps are Needed for:
 - Navigation
 - City Planning
 - Surveying
 - Roads, Railways, Airports planning

METHOD OF STUDY

- Importing Required Python Libraries
- Data Acquisition and Processing
- Generating a Community Map of Staten Island
 - Defining an instance of the geocoder
 - Creating the Community Map

REQUIRED PYTHON LIBRARIES

```
import numpy as np # library to handle data in a vectorized manner
import pandas as pd # library for data analysis
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
import json # library to handle JSON files
!conda install -c conda-forge geopy --yes
from geopy.geocoders import Nominatim # converts an address into latitude and longitude values
import requests # library to handle geolocator requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas dataframe

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors

!conda install -c conda-forge folium=0.5.0 --yes
import folium # map rendering library
```

Figure 1: Python Syntaxes for Importing the Relevant Libraries

DATA ACQUISITION AND PROCESSING

```
!wget -q -O 'newyork_data.json' https://cocl.us/new_york_dataset
with open('newyork_data.json') as json_data:
    newyork_data = json.load(json_data)
neighborhoods_data = newyork_data['features']
column_names = ['Borough', 'Neighborhood', 'Latitude', 'Longitude']
neighborhoods = pd.DataFrame(columns=column_names)
for data in neighborhoods_data:
    borough = neighborhood_name = data['properties']['borough']
    neighborhood_name = data['properties']['name']
    neighborhood_latlon = data['geometry']['coordinates']
    neighborhood_lat = neighborhood_latlon[1]
    neighborhood_lon = neighborhood_latlon[0]
    neighborhoods = neighborhoods.append({'Borough': borough,
                                         'Neighborhood': neighborhood_name,
                                         'Latitude': neighborhood_lat,
                                         'Longitude': neighborhood_lon}, ignore_index=True)
staten_island = neighborhoods[neighborhoods['Borough'] == 'Staten Island'].reset_index(drop=True)
staten_island.head()
```

Figure 2: Python Syntax for obtaining and processing the needed data

DATA PRESENTATION

	Borough	Neighborhood	Latitude	Longitude
0	Staten Island	St. George	40.644982	-74.079353
1	Staten Island	New Brighton	40.640615	-74.087017
2	Staten Island	Stapleton	40.626928	-74.077902
3	Staten Island	Rosebank	40.615305	-74.069805
4	Staten Island	West Brighton	40.631879	-74.107182
5	Staten Island	Grymes Hill	40.624185	-74.087248
6	Staten Island	Todt Hill	40.597069	-74.111329
7	Staten Island	South Beach	40.580247	-74.079553
8	Staten Island	Port Richmond	40.633669	-74.129434
9	Staten Island	Mariner's Harbor	40.632546	-74.150085

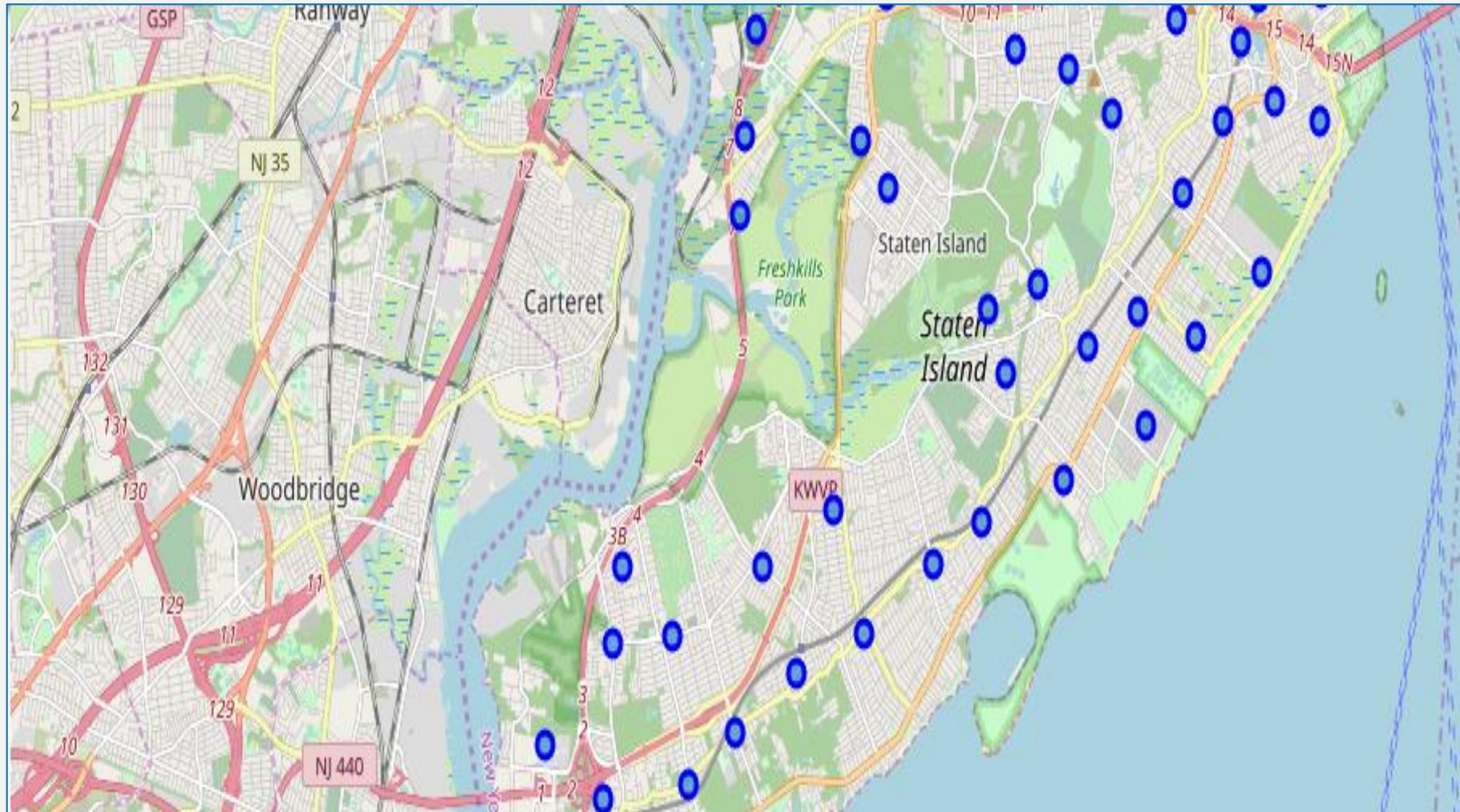
Figure 3: Snapshot of Pandas data frame showing the first 10 rows of the data (Neighbourhoods of Staten Island with their Latitude and Longitude)

GENERATING A COMMUNITY MAP OF STATEN ISLAND

In Three Steps:

1. Defining an instance of the geocoder with a user agent
2. Generate the longitude and latitude of the Staten Island Borough
3. Create map with python folium library

RESULTS PRESENTATION



○ Neighbourhoods

Figure 4: Map of Staten Island Showing Neighbourhoods



THANK YOU