# Data Engineer Assignment

Love, Bonito

Completed By Cheryl Ting
https://www.linkedin.com/in/cheryl-ting-hung-niu-6269b886/

# Data Engineer Assessment

Goal: To implement a dashboard for the data team to query top 10 highest and lowest taxi population, hourly taxi availability across Singapore and implement alert mechanism for areas with no taxi availabilities

General Attributes of Taxi Availability Dataset:

- Queried from API endpoint https://data.gov.sg/dataset/taxi-availability
- Data represents locations (coordinates) of taxis that are currently available (not “hired” or “busy”) in Singapore
- Timestamp of data refers to the scrape time
- Data collection time range is from 7th June 8pm – 8th June 4pm

Programming Language: Python for extraction, processing and presentation

Extraction mode: Batch extraction every 15 minutes

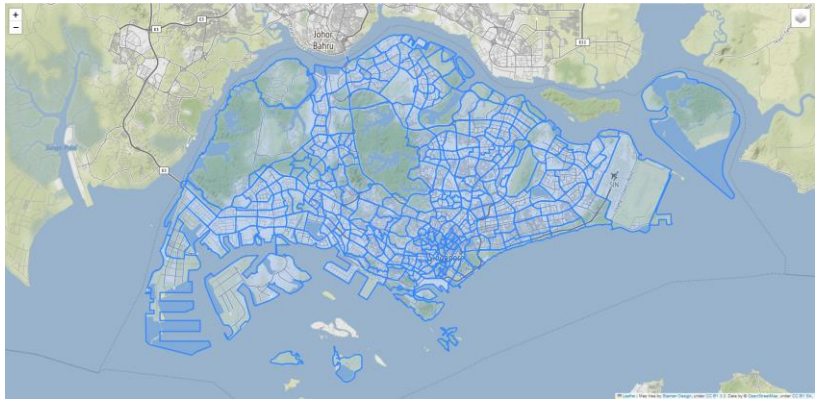
Github repo: https://github.com/IndieWorld82/TaxiAvailability

# Data Structures

- 2 tables were created in an MSSQL database (https://github.com/IndieWorld82/TaxiAvailability/blob/main/taxi_data_structures.sql)
- TaxiAvailability with following columns:
  - ID (int) → Primary Key
  - Timestamp (datetime) → Scrape time
  - Location (geometry) → Location coordinates of available taxi
  - TaxiCount (int) → Total count of taxis
  - RegionID (int) → Foreign key to ID of table RegionBoundaries
  - RegionDetectedFlag (tiny int) → Flag to indicate if taxi coordinates are found in RegionBoundaries reference table
- RegionBoundaries with following columns:
  - ID (int) → Primary Key
  - RegionName (varchar(100)) → Region name
  - RegionCode (varchar(10)) → Region code
  - Geometry (geometry) → Polygon data of region boundaries
  - SubZoneNo (varchar(20)) → Subzone number
  - SubZoneName (varchar(100)) → Subzone name
  - SubZoneCode (varchar(20)) → Subzone code
  - AreaName (varchar(100)) → Area name
  - AreaCode (varchar(20)) → Area code
  - INC_CRC (varchar(60)) → Reference data from KML file
  - FMEL_UPD_D (varchar(60)) → Reference data from KML file

# Data Extraction and Preparation

- In order to group taxi population by areas, first step was to group Singapore into distinct sub zones
- Subzone boundaries for Singapore was obtained from Data.gov.sg (https://data.gov.sg/dataset/master-plan-2019-subzone-boundary-no-sea?resource_id=84b62d90-c1b7-4ada-acfc-f5874b5fd945)
- Python used to load data from KML file to MSSQL table 'RegionBoundaries' (https://github.com/IndieWorld82/TaxiAvailability/blob/main/region_boundaries.py)
- To validate the regions data, Python code was created to visualize the boundaries in a map (https://github.com/IndieWorld82/TaxiAvailability/blob/main/check_region.py)



Link to map:
https://github.com/IndieWorld82/TaxiAvailability/blob/main/regional_map.html

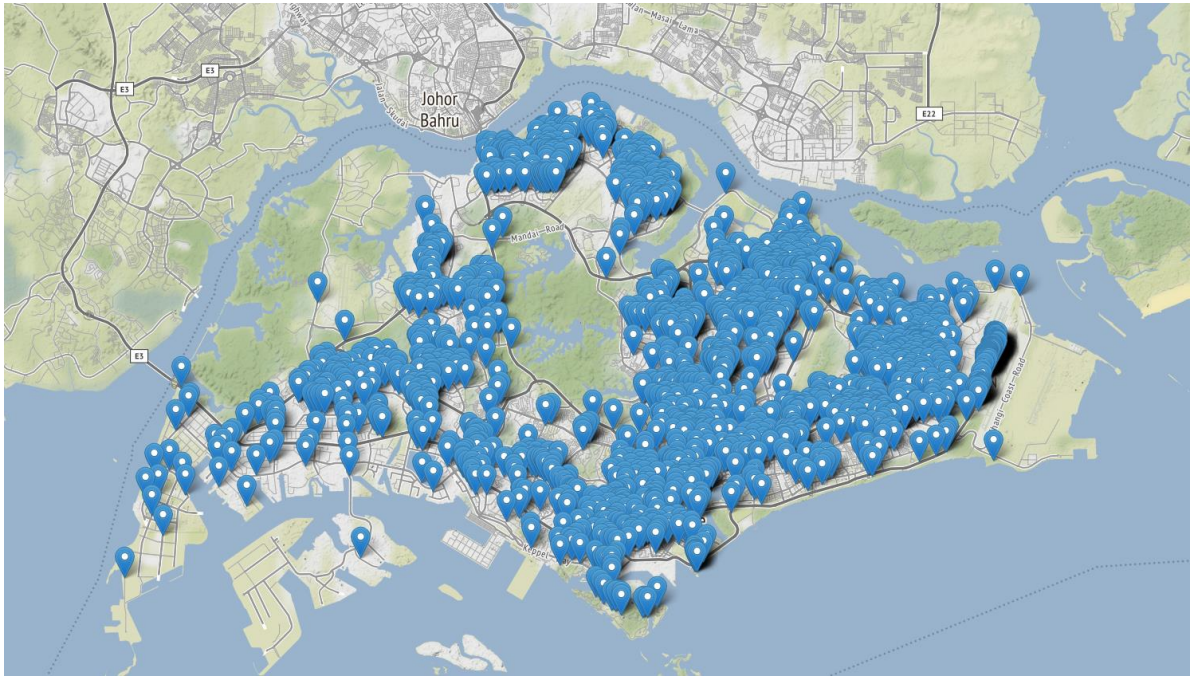
# Data Extraction and Preparation

- Python used to query taxi availability data from the API given
- Before inserting the information into the table ‘TaxiAvailability’, the location of the point coordinates was scanned from the table ‘RegionBoundaries’
- MSSQL spatial function STContains was used to detect the RegionId based on the coordinates of the taxi location data
- If the RegionID could not be found, the RegionDetectedFlag is set to 0 to indicate faulty data
- The Python code can be found here: https://github.com/IndieWorld82/TaxiAvailability/blob/main/taxi_availability_dataset.py

# Dashboard

## Real Time Taxi Availability Visualization in Singapore

- Python script: https://github.com/IndieWorld82/TaxiAvailability/blob/main/taxi_availability_map.py
- Using the ‘folium’ library to plot current locations of taxis on the map

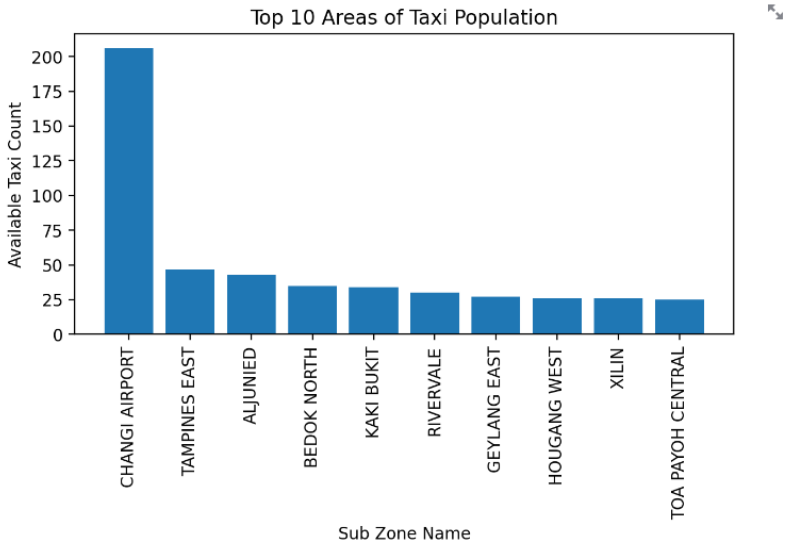



Link to map:
https://github.com/IndieWorld82/TaxiAvailability/blob/main/taxi_availability_map.html

# Dashboard

## Charts

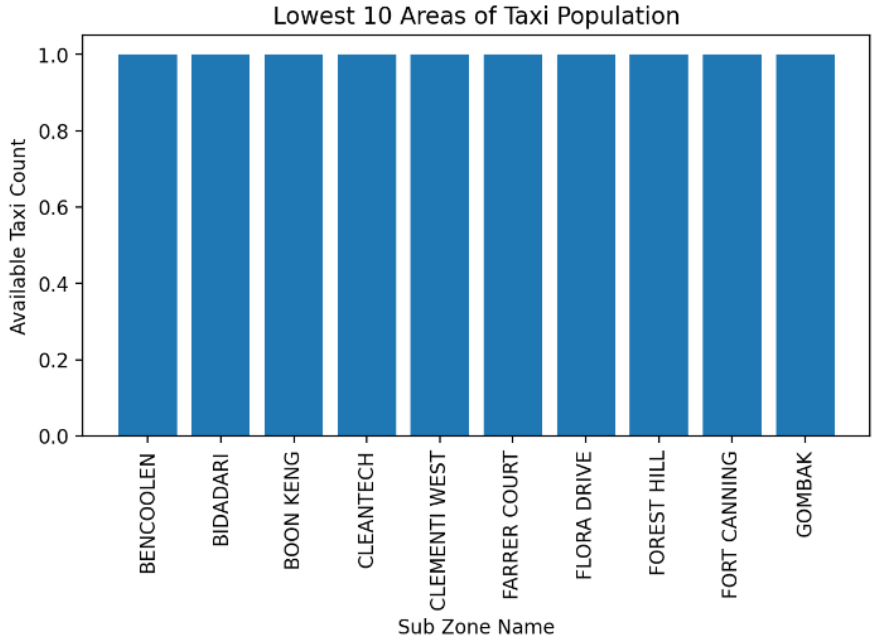
- Python script: https://github.com/IndieWorld82/TaxiAvailability/blob/main/taxi_availability_charts.py
- Using the ‘streamlit’ library to publish on local web server
- The chart of top 10 areas of taxi population is displaying the latest data only



# Dashboard

## Charts

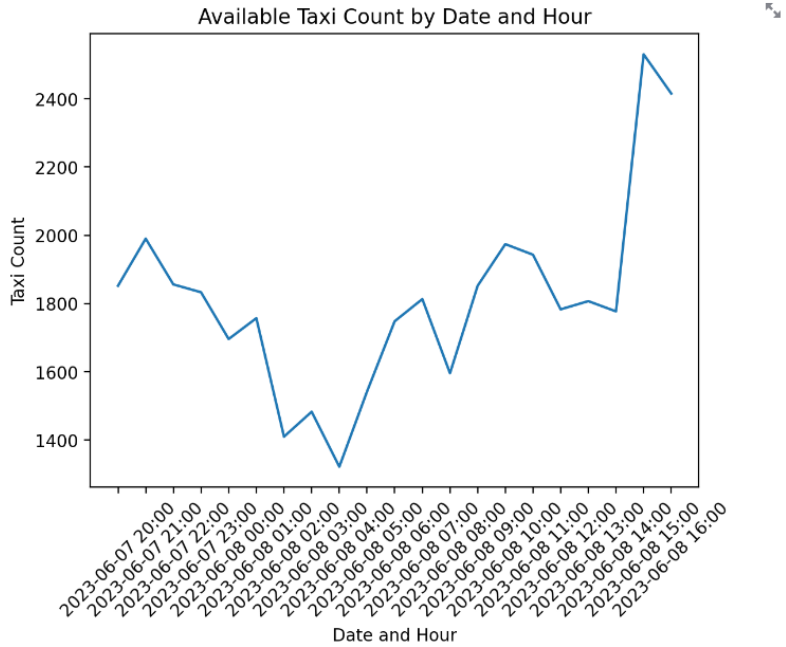
- The chart of lowest 10 areas of taxi population is displaying the latest data only



# Dashboard

## Charts

- The hourly taxi count displayed is taking the average of taxi counts per API query within an hour



# Alert Mechanism for Areas with no Taxis

- Using Python library ‘python-telegram-bot’ to send Telegram messages alerting team of areas with no taxi availabilities
- Connected to MSSQL table to query areas with no taxi availability based on latest data pulled from API
- Python code: https://github.com/IndieWorld82/TaxiAvailability/blob/main/alert_no_taxi.py

Note: Telegram bot was not actually configured here, only displayed in the code as the means of alert mechanism