# README

# Street Tree List API

## About

This repository contains code for accessing and analysing tree plantation data in the City of Francisco from US government API. The API offers a comprehensive look at the number of trees planted and maintained by the Department of Public Works in the city of Francisco.

To use this code, you will need to have access to the API provided by the US government. Once you have access to the API, you can use the code in this repository to access and analyse the data. The code is written in Python and is designed to be easily modifiable to suit your needs.

To get started, you will need to clone this repository to your local machine. Once you have done that, you can open the code in your favourite Python editor and begin exploring the pollution data. The code is well-commented and should be easy to follow.

API Used:

## <https://data.sfgov.org/api/views/tkzw-k3nq/rows.csv?accessType=DOWNLOAD>

## Pre-requisites

Visual Studio- 1.77.3

Python version- 3.11

MySql server in openstack- 8.0.32

PostgresSql server in openstack- 11.19

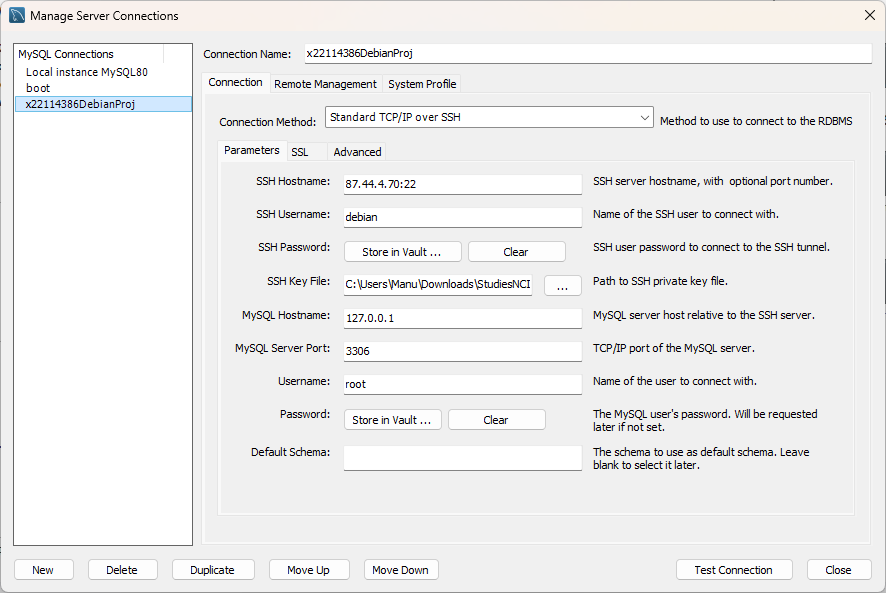
PEM file in the link – new.pem

PPK file in the link – new.ppk

Debian server in openstack- 10.0

Debian Instance- [x22114386Project](https://openstack.cloudenci.ie/project/instances/b22b1dab-19d0-494f-b5c0-9616dfb7ea1b/) - 87.44.4.70

* Mysql server details and database details:



* Postgres Server and database details:

user = "dap",

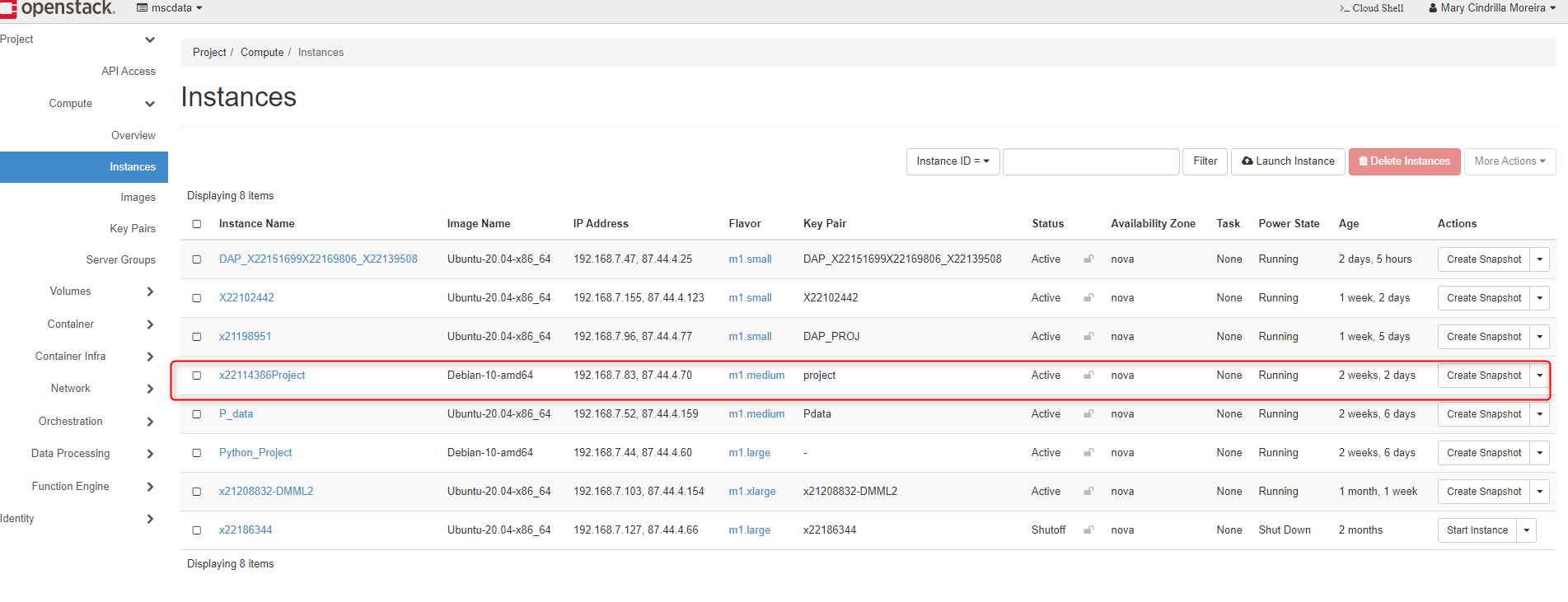
password = "dap",

host = "87.44.4.70",

port = "5432",

database = "postgres"

* Openstack cloud



More insights on the DB connections are attached in file “MySQL Server DATA BASE CONNECTION DETAILS.DOCX” in the file structure.

## Sequence of execution of Program

1. **x22153926ProjectApi. ipynb**

This is the first file to execute this file consist of

a. Triggering the open API.

b. Extracting the Json Output to a CSV file in the local path. The output file is outttput.csv

1. **x22153926ProjectDB. ipynb**

The code is related to Data base connection.

a. Connection string to create a connection pipeline to Database.

b. Creation of Database Schema

c. Dump the entire TreeplantCleaned to the table.

d. Normalization of the data

e. Insertion of data with respect to table.

1. **x22153926ProjectPsql.ipynb**

This code gives the connectivity to PostgreSQL pipeline in case we are unable to create a pipeline in dagster just an intermediate file for reference.

a. Connection string to create a connection pipeline to Database.

b. Creation of Database Schema

c. Dump the entire TreeplantCleaned to the table.

d. Normalization of the data

e. Insertion of data with respect to table.

1. **x22153926ProjectCleaningAnalysis.ipynb**

The CSV file is taken, and analysis of the file has been done.

Post this all the 4 dataset is merged, analysed on a common ground.

## Other files in the folder structure:

1. [shortclean.csv](https://github.com/x22114386/Project_DataBase_DataAnalytics/blob/akshay/shortclean.csv) – Cleaned CSV data.
2. [RAWTREE.csv](https://github.com/x22114386/Project_DataBase_DataAnalytics/blob/akshay/RAWTREE.csv) – Raw CSV data
3. [DATA\_CLEANING.ipynb](https://github.com/x22114386/Project_DataBase_DataAnalytics/blob/akshay/DATA_CLEANING.ipynb)- Data Cleaning
4. [DATA VISUALISATION.ipynb](https://github.com/x22114386/Project_DataBase_DataAnalytics/blob/akshay/DATA%20VISUALISATION.ipynb) – Data visualisation
5. [x22152563DAPAPI.ipynb](https://github.com/x22114386/Project_DataBase_DataAnalytics/blob/akshay/x22152563DAPAPI.ipynb) – API fetching server.
6. ShortDescriptiononDataSets.txt- about the dataset
7. MySQL Server DATA BASE CONNECTION DETAILS- for the data base connectivity using ssh tunnel.
8. new.pem- private key for ssh tunnelling
9. new.ppk- for putty connection
10. [MAIN PYTHON FILE.ipynb](https://github.com/x22114386/Project_DataBase_DataAnalytics/blob/akshay/MAIN%20PYTHON%20FILE.ipynb) – File which contain all the code.