ETL Project Report

**Objective:**

The goal of this project is to perform what is commonly referred to as ETL. ETL stands for extract, transform, and load. This is a practice that is crucial in creating data that is clean and usable for the purpose of analyzing, storing, etc. The process can be broken down into its individual parts: extracting or acquiring the data from a source or sources, transforming or cleaning it into a form that is more readable and usable, and finally loading the data into a database or table.

**Extract:**

Data is available in a variety of formats from a variety of sources. For this project, data was collected from Kaggle, a free-to-use website that allows the open sharing of databases that cover diverse topics. The first dataset utilized outlines recent NYC property sales around the five boroughs. This dataset was downloaded from Kaggle, in the format of a CSV file. The second dataset utilized is a NYC tree census. This dataset contains in formation about the tree located throughout the five boroughs of NYC. This was also downloaded from Kaggle, in the format of a CSV file.

**Transform:**

To transform this data into a usable form, Pandas, a popular Python library, was utilized. Pandas proved to be very effective here due to its ability to easily load data from a CSV (or other) file into a pandas dataframe, where data can be manipulated into a form the user finds relevant to their analysis. The CSV files were read into a pandas dataframe. Not all the columns are relevant to us, therefore the first step in our cleaning was dropping any columns that will not be used later. This left us with [insert what we kept]. In order to perform be placed in SQL tables (in the hopes of performing a join later), the columns needed to be renamed to match the column names of tables that will eventually be made using Postgres. Due to SQL being a relational database, the tables will need to possess something that matches to be able to specifically join on.

**Load:**

After we were satisfied with the format of our dataframes, they were loaded into tables in Postgres (pgAdmin4). This involved creating tables that we will eventually load our data into. Each table is assigned a primary key (typically id) so that we could relate one table to another. We created tables with the number and names of columns present in the dataframes we previously created.