Real Estate Investments, LLC.

MIS 740
Final Report

Team 3:
Jason Jasper
Shanthan Venkat Pochampally
Kumar Rishav
Luke Stevens
Curtis Wa

Company Background

Real Estate Investments, LLC is a real estate property investment and property management firm. The company buys and sells many homes every month across the United States. Some of these homes are short term purchases only staying in their portfolio for 6-18 months at a time before being sold again. Even though some of the homes are short term purchases, the company is still responsible for paying property taxes for all of the properties. Due to the national scale of their portfolio, it is difficult for the company to manage the different tax rates in each state, much less each county. Currently, Real Estate Investments doesn't utilize information systems very much. They have a master Excel spreadsheet that holds all the data relevant to the property and their tax payments. Moving forward, they are looking to automate the process of data entry, find out the property taxes, tax payment date, as well as quickly calculating their profit loss.

Business Problem

The organization currently manually enters the data regarding their home purchases. This is important information such as property tax rates, property tax due dates, purchase date, purchase price and more. Due to the limitations of Excel, employees must search for each property from their respective treasurer's office website and then input the data into the master Excel sheet. The manual entry of this data can lead to human errors such as incorrect property tax rates or wrong due dates being inputted. These mistakes can lead to missed payment fees or money lost due to over payments ranging from hundreds of dollars per home. In addition, this data entry takes a large amount of time to complete. The company prioritizes saving money and time. Creating an in-house solution would reduce the capital in comparison to needing to outsource this project to a third party. Another function the program would serve is to aggregate data. This combined data could be used to easily create visualizations and help the company make more informed decisions about where to direct the company.

Functional Requirements

The required solution will ingest or search through large datasets to extract information based on addresses, parcel numbers, or both. This is a high value proof of concept for the business using open data sets and sources. The company will later convert to commercial (paid) data sources and APIs.

The program should have two initial inputs. The ability to input addresses or parcel numbers manually, or in bulk by a data file such as CSV format. These addresses/parcel numbers are to be queried for property value and tax against the target data set. The program should then validate the inputted addresses/parcel

numbers are in the correct format for querying. If not, the program should output that there are issues with the formatting, and provide a sample format.

The program should be able to determine whether to parse, or iterate through the target dataset based on the file size of the dataset. Large datasets should not be parsed into memory. Using the imputed dataset and target dataset, the program will search for the properties in question and store property values and taxes in memory.

Once the property search has completed, the program should output summary data, including:

- Number of properties found in the dataset, and number of properties **not** found in the dataset.
- The total value of properties.
- The total tax liability of the properties.

The program should then present a number of reports that can be outputted, then prompt the user whether they want specific reports, all, or no reports to be generated. The reports included (for now) are:

- None. (Summary data only).
- All reports.
- Property values only.
- Property tax owed only.
- Property values with property tax.
- Aggregate reports.
    - Total taxes per city.
    - Total taxes per zip code.
    - Total property value per city.
    - Total property value per zip code.

Reports will include the summary data for the number of properties found and number of properties not found, and the total value and total tax liability of the properties. Choosing to output a property value and/or property tax owed report will list found properties with the chosen information. Selecting aggregate reports will prompt the user to choose aggregation by city or zip code and if the report should include property value and/or property tax owed. The aggregated report will show the summation of the selected value for all properties found in the dataset for each unique zip code or city. Selecting all reports will result in three reports: one listing the property values and property tax owed for each property, the second listing the aggregated information by city, and the final listing the aggregated information by zip code.

The reports selected will then need to be saved to the disk with a predetermined filename format such as: (Report_Name)-(Run_date)-(Run_time).csv As of right now the reports will be saved as CSV files until the need for other formats become necessary. Finally, the program will prompt the user if they would like to run the program again, or quit.

Business Value

Since the company wants to automate its process, the errors caused by human indulgence and the errors, in general, can be eliminated which leads to a more consistent and efficient way of data storage.

Last year, The mistakes made by employees when entering data into the spreadsheet cost the firm a total of US $10000 in lost revenue. Additionally, reputation is also lost among our partners and clients. With this project, the company hopes to bring it down to the lowest amount possible. It will be very convenient for the firm to manage the tax rates of a particular state, or even a county due to automation.

Entering data into excel manually is very time consuming, automating it will save on input time which in turn results in freeing up time of the data entry employee so that they can be used on more important tasks. Automation and programming also have a significantly higher success rate and higher time efficiency while calculating the property taxes, tax payment due date or to compute the profit/loss of a particular estate.

The time taken to search for information about a certain property can also be significantly reduced. Based on an analysis done by the firm, the average time taken for an employee to search for a property from the respective treasurer’s office website and then to input the data into the master Excel sheet was calculated to be 2.4 minutes. Whereas the entire process can be achieved in a matter of seconds with automation.

Automation also helps the firm by generating automatic summary reports based on the input. With the various kinds of reports, the company can benefit from a customer satisfaction point of view as the customers can now filter the reports according to their needs. This also helps the company in making informed decisions moving forward.

Introduction

Companies participating in the real estate market must pay taxes for their held properties. By automating the process of finding property tax values, this program reduces the man hours required to find data for individual properties and reduces errors caused by manual entry, saving the company money in personnel costs and avoiding fees for unpaid taxes. This program accomplishes the task by selecting data from a file containing property values and property taxes based on the zip code or city of interest to the company. Reports are generated either by individual property or by agglomerating the data based on zip code and city.

Interface

Since this is an earlier prototype of a more extended system, the interface will be designed as a CLI type interface with textual menus and commands.

Legend:
Underline text is for paths
Bold text is processing
Italicized text is input and output

MAIN PROGRAM LOOP UI

Program starts
The main menu is displayed :
“ Welcome to the program. “
“Main Menu - Please input the number you like for the specific function : “

1) *Ad Hoc Property Search*
2) *Import Property List*
3) *Import Data Set*
4) *Run Property Search*
5) *Generate Report(s)*
6) *Show Summary Report*
7) *Quit*

The program reads the input and checks the condition that checks if the input is valid
If user inputs invalid value, then the program displays an error message:

“Invalid entry, please enter again.”
User inputs a valid function number

AD HOC PROPERTY SEARCH UI

If user enters “1”, the program says:
“Would you like to search for properties with the parcel number or the address? (parcel/address)”
User inputs parcel number or the address
The program reads the input and checks the condition that checks if the input is valid
If user inputs invalid value, then the program displays an error message:
“Invalid entry, please enter again :”

Path 1:
If the user has inputted “parcel number”
“Please enter the parcel number related to the property :”
The program sets a function name called “useParcelNumber to “true”
The program then flows into the “Main loop flow diagram”

Path 2:
If the user entered “address” instead:
“Please enter the street Number :”
“Please enter street :”
“Please enter city :”
“Please enter state :”
“Please enter zipcode :”
The program checks whether the zipcode is numeric
If the program is not numeric, program displays error message :
“Output Zip Code is invalid, please enter again :”
If the data is validated, the program sets a boolean flag called “useAddress” to “true”
The program flows into the “Main loop flow diagram”

PROPERTY LIST IMPORT UI

If user enters “2”:

The program checks if the adHoc search has already been done or not

If adHoc has been done :
“AdHoc address has already been entered, are you sure? “
User inputs “Y” or “y”
Program flows to path 1

If adHoc has not been done :
Program flows to path 1

Path 1:
“ Please input the File name : “
Program reads the file name
Program checks if the file name exists

Path 1(a):

If path does not exist :
“File not found, please enter input Filename again”

Path 1(b):

If path exists:
Program sets propertyListFileName, inputFileName, and useDataFile to “True”
“Data file loaded”
Flows into the main loop

Path 2:

If user enters anything except “Y” or “y” :
Program flows into the main loop

DATA SET IMPORT UI

If User enters “3”:

Path 1:
“Please input the filename :”
Filename.csv
“This file does not exist. Please input a valid file name :”
Dataset.csv
“Data file format is not compatible.”

Path 2:
“Please input the filename :”
Filename.csv
“This file does not exist. Please input a valid file name :”
Dataset.csv
“Data file format is compatible.”

PROPERTY SEARCH UI

If User enters “4”:
Program checks if the adHoc search has already been done or not

If adHoc has been done:

Path 1:
The program checks if either “useAddress” or “useParcelNum” are done or not
Program equates either parcelNumber or address to searchParameter
“Property search starting. Please wait.”
Program begins calculations
“Property search completed.”

If adHoc has not been done :

Path 2:
Program checks if property list has been imported or not

Path 2(a):

If "no" :
"Property search type not specified, please choose one from the main menu first"
Flows to the main loop

Path 2(b):
If "Yes" :
"Property search starting. Please wait."
The program starts calculating
"Property search completed."

Program flows back to the main loop

GENERATE REPORTS UI

If User enters "5":

Program checks if property search has been done or not

If "No" :
"Property search not performed, perform this first. "
Program flows to the main loop

If "Yes" :

"Reports Main Menu :"
"Please select and input the type of report you want to generate :"

A. *All reports*
B. *Property values only*
C. *Property tax owed only*
D. *Property values with property tax*

“Aggregate Reports Menu :”

E. Total taxes per city
F. Total taxes per zip code
G. Total property value per city
H. Total property value per zip code

User inputs the number for the report type
Program checks if the number is valid

If “No” :
“Invalid report type, please input again.”

If “Yes” :
Program generates the specified report type

“Report completed. File saved to (Report type).csv”

SUMMARY REPORTS UI

Program outputs :
“Total Properties Found: propertyCount”
“Total Property Value: totalValue”
“Total Property Tax: totalTax”

“Summary report data generated to ‘SummaryReport.txt’ “

Design

The main design is centered around a main loop that displays a main menu and asks for menu item entry. Each menu has its own functionality, which calls the appropriate functions. Once any part of the program completes its function, the program then goes back into the main menu loop until the user explicitly uses the “Quit Program” menu item to end the program. The function inputs addresses or parcel numbers or imports a property data file in CSV format. The user then instructs the program to perform the property search. The user would then ask the program to generate 1 out of 7 reports, all reports, or just a summary report.

The design is illustrated in the following 14 figures below:

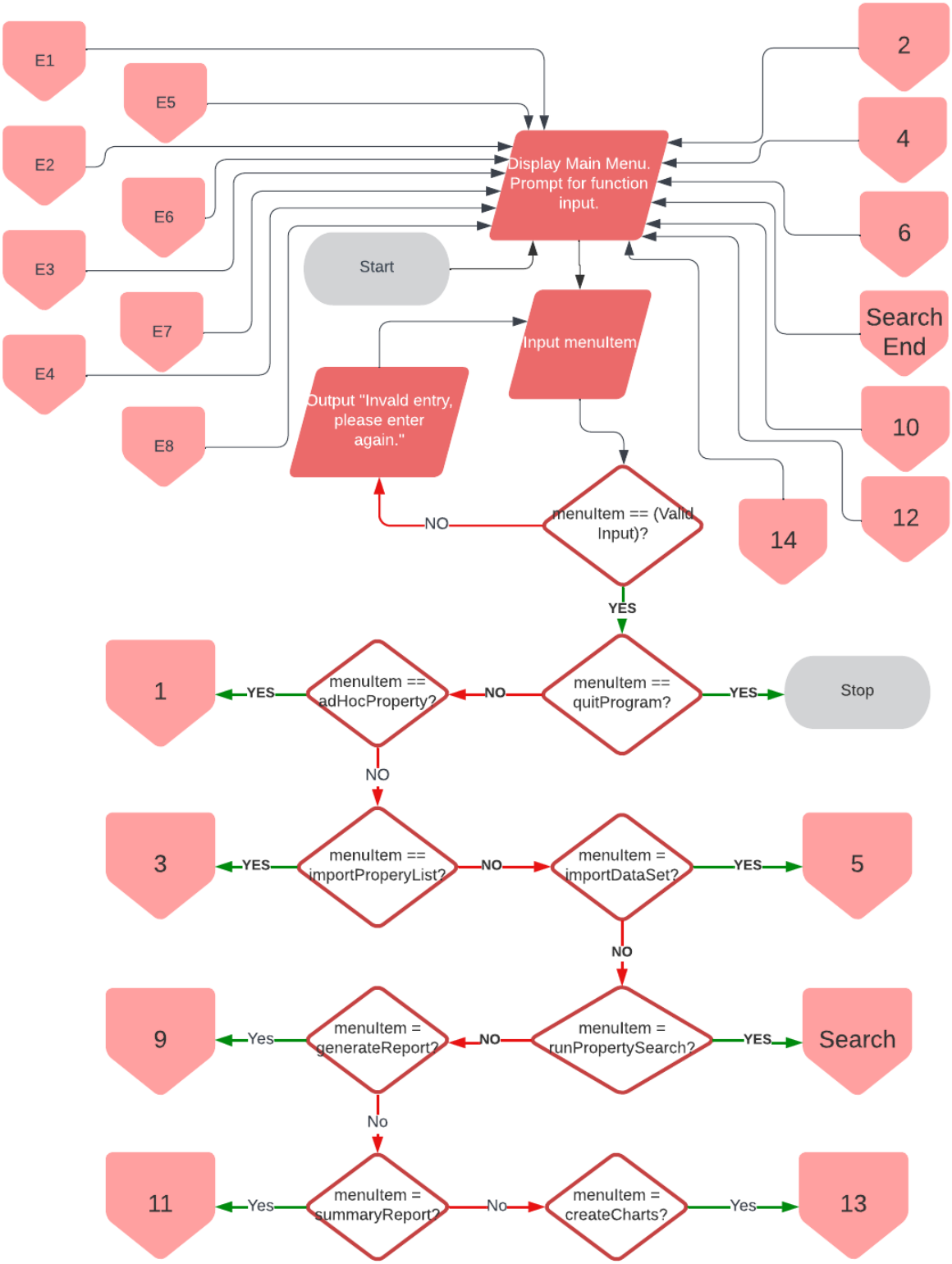


Figure 1: Main loop flow diagram

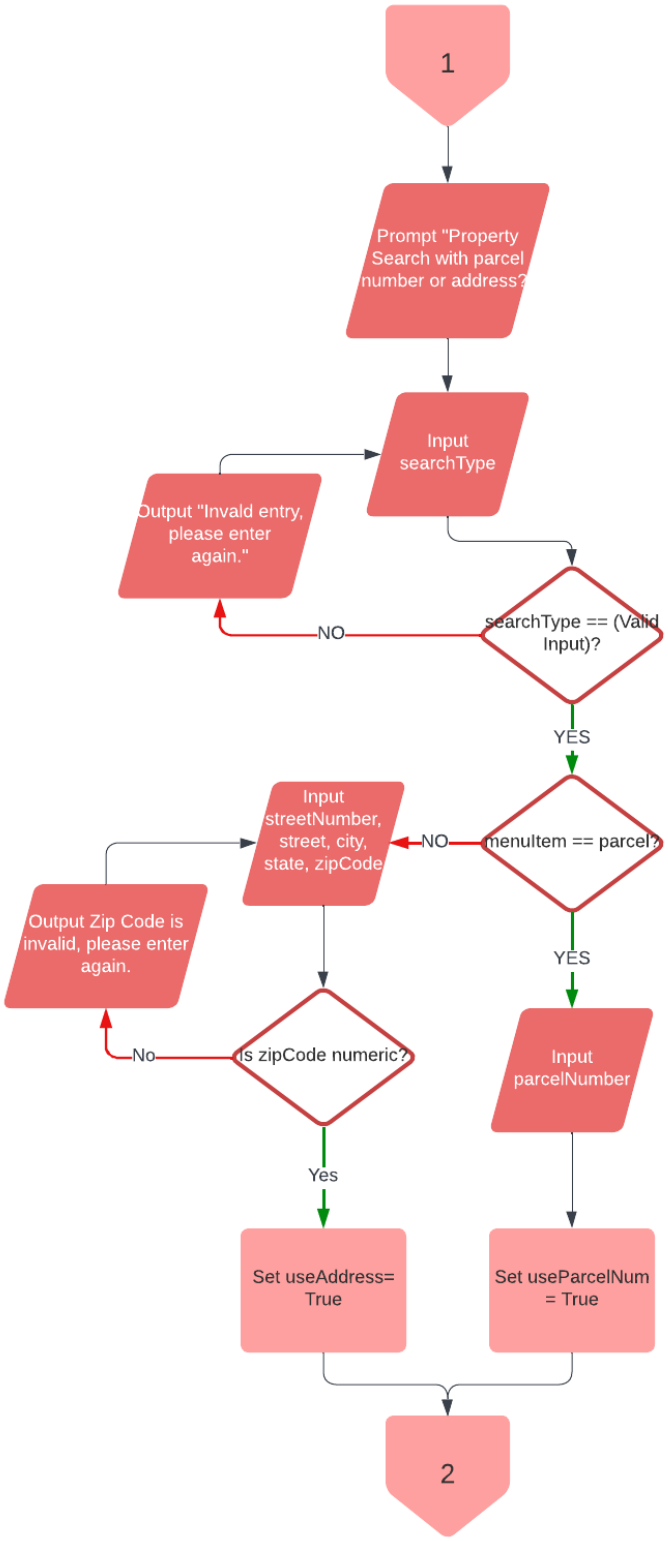


Figure 2: Ad Hoc Property Search flow diagram

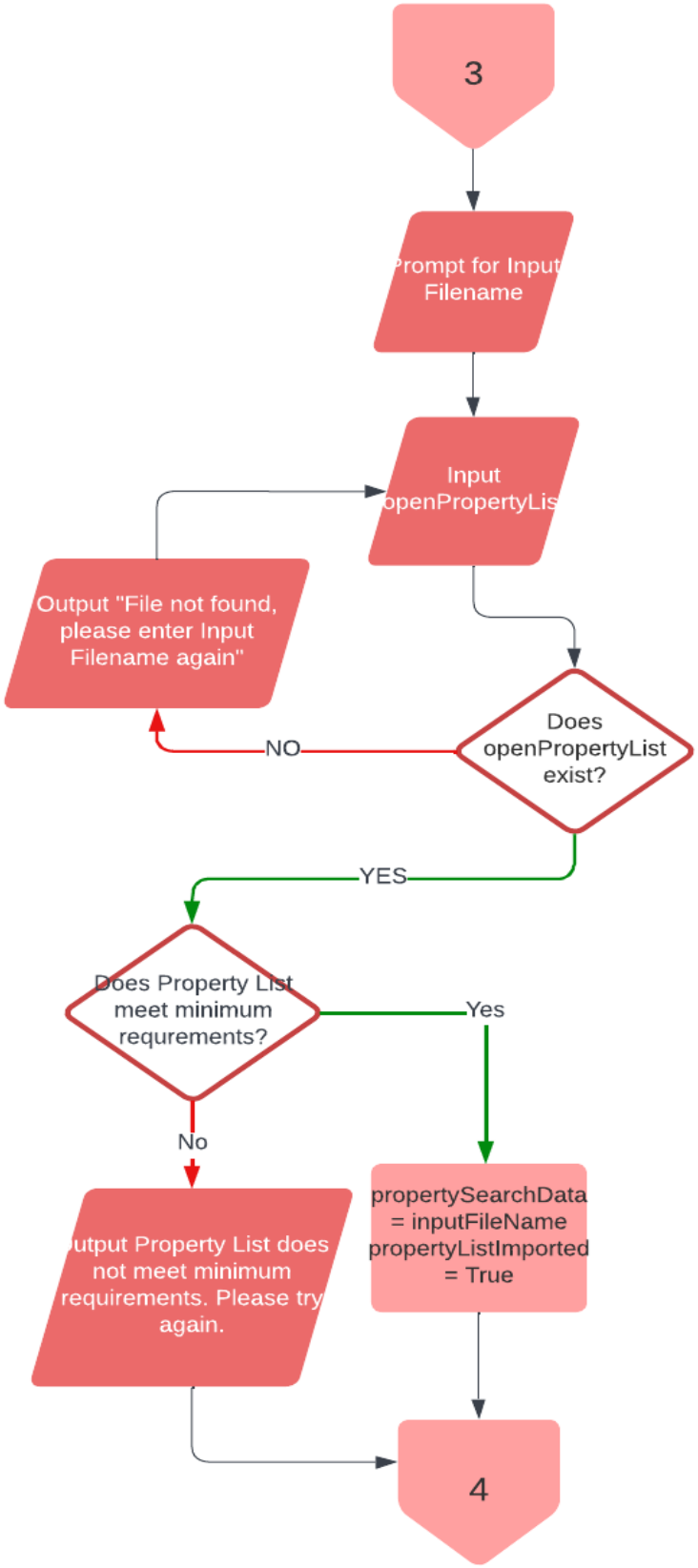


Figure 4: Property list import flow diagram

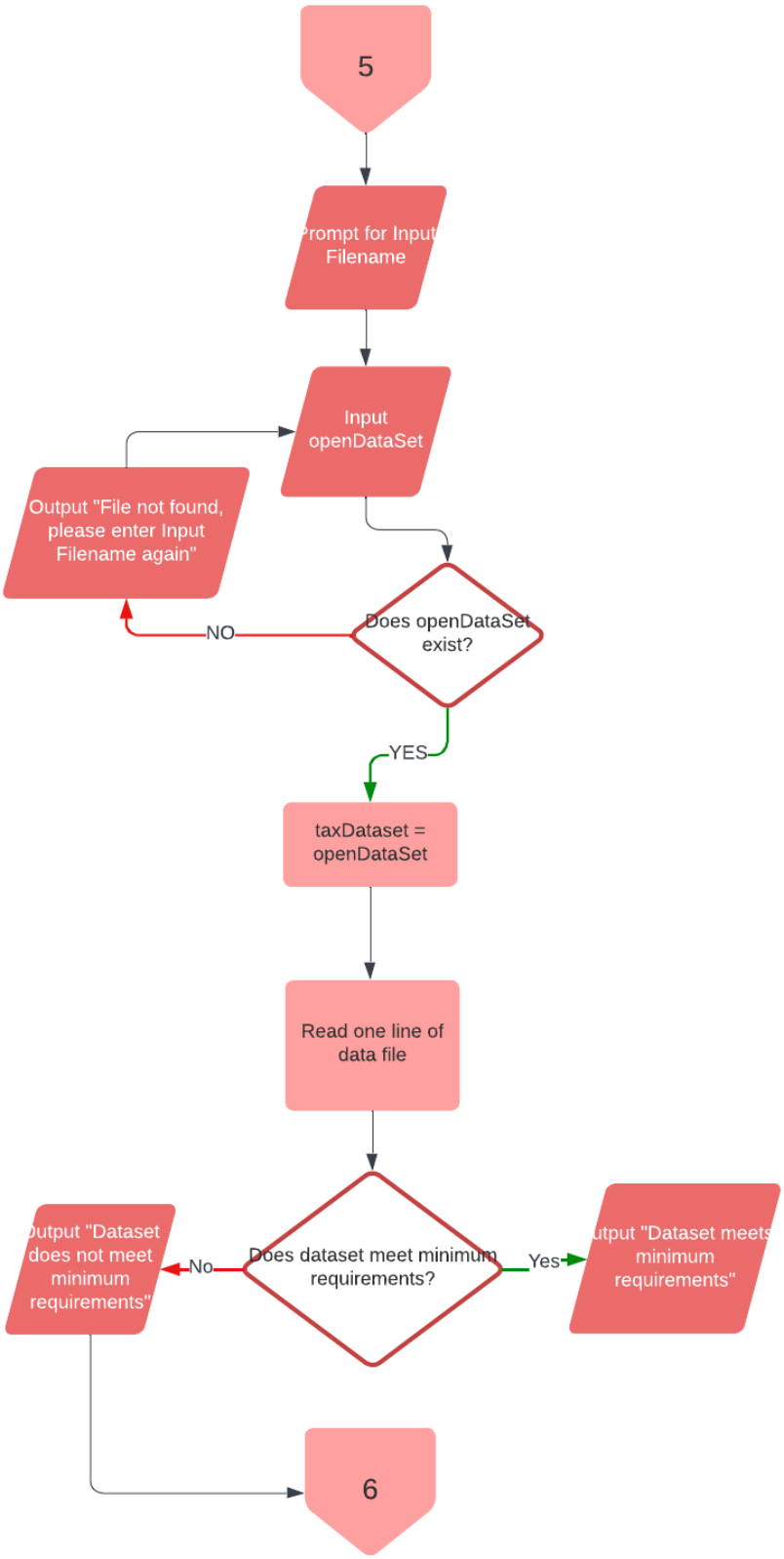


Figure 5: Data set import flow diagram

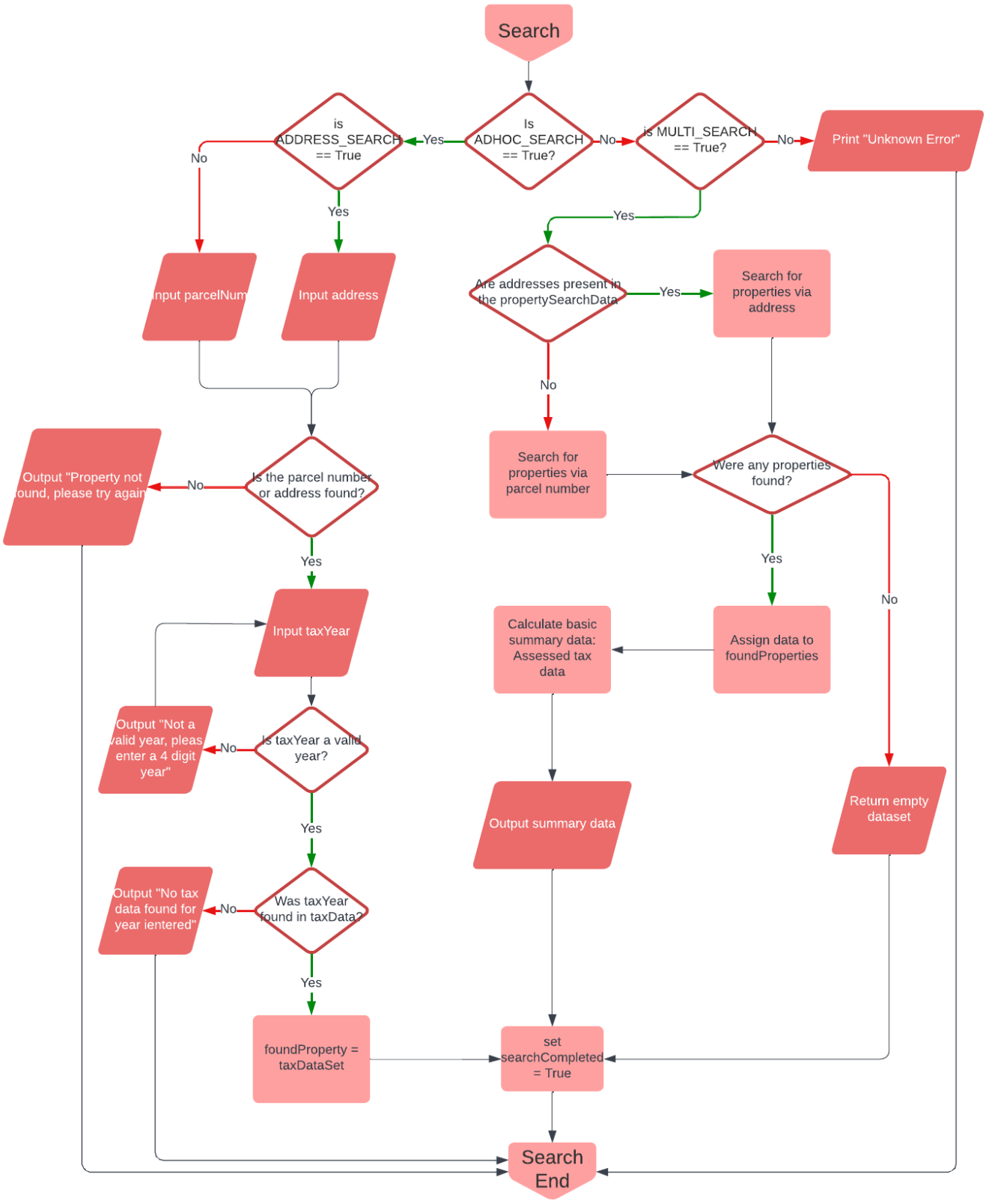


Figure 6: Property search flow diagram

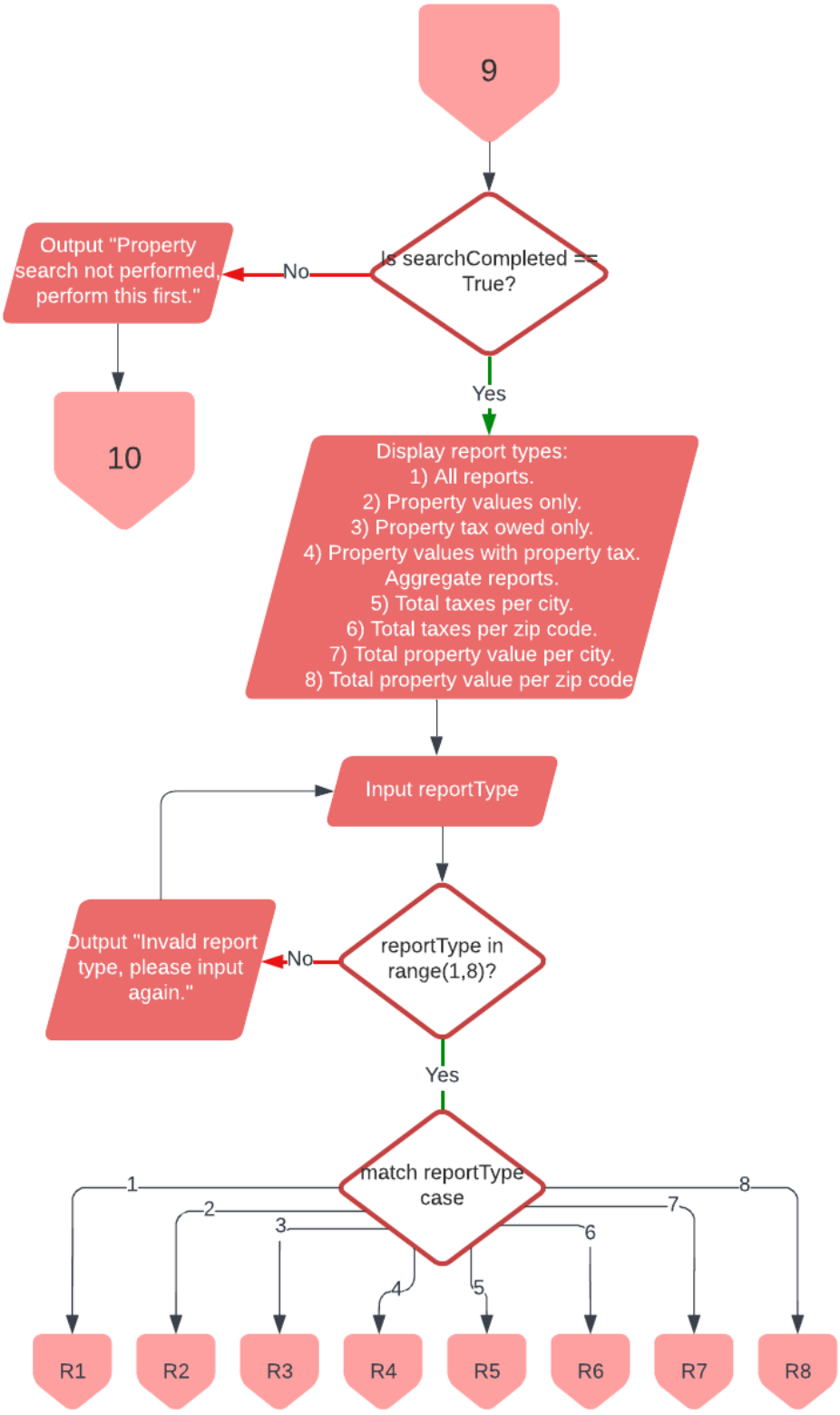


Figure 7: Generate reports flow diagram (main section)

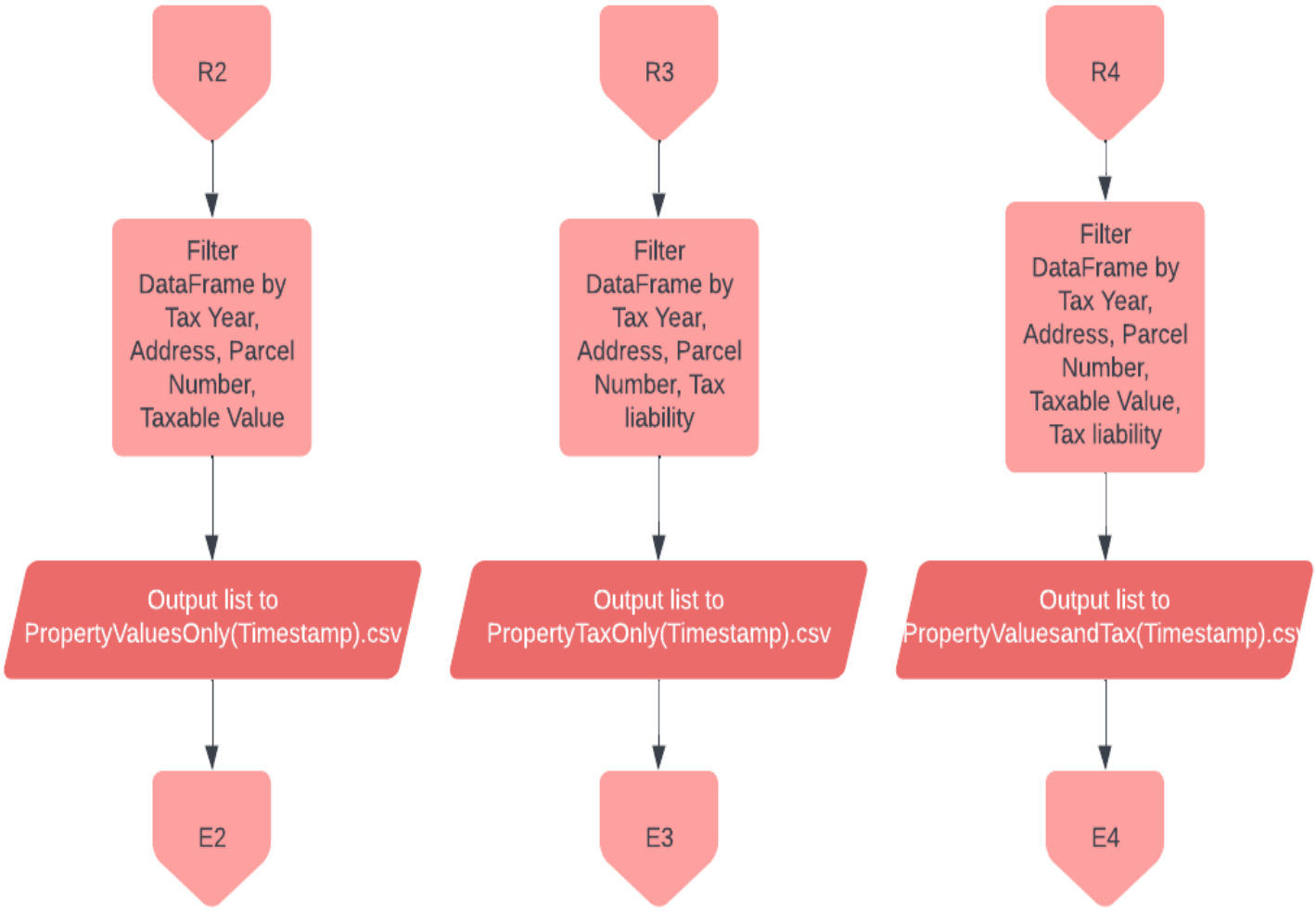


Figure 8: Reports 2, 3, and 4 (Property Values Only, Property Tax Only, Property Values and Taxes)

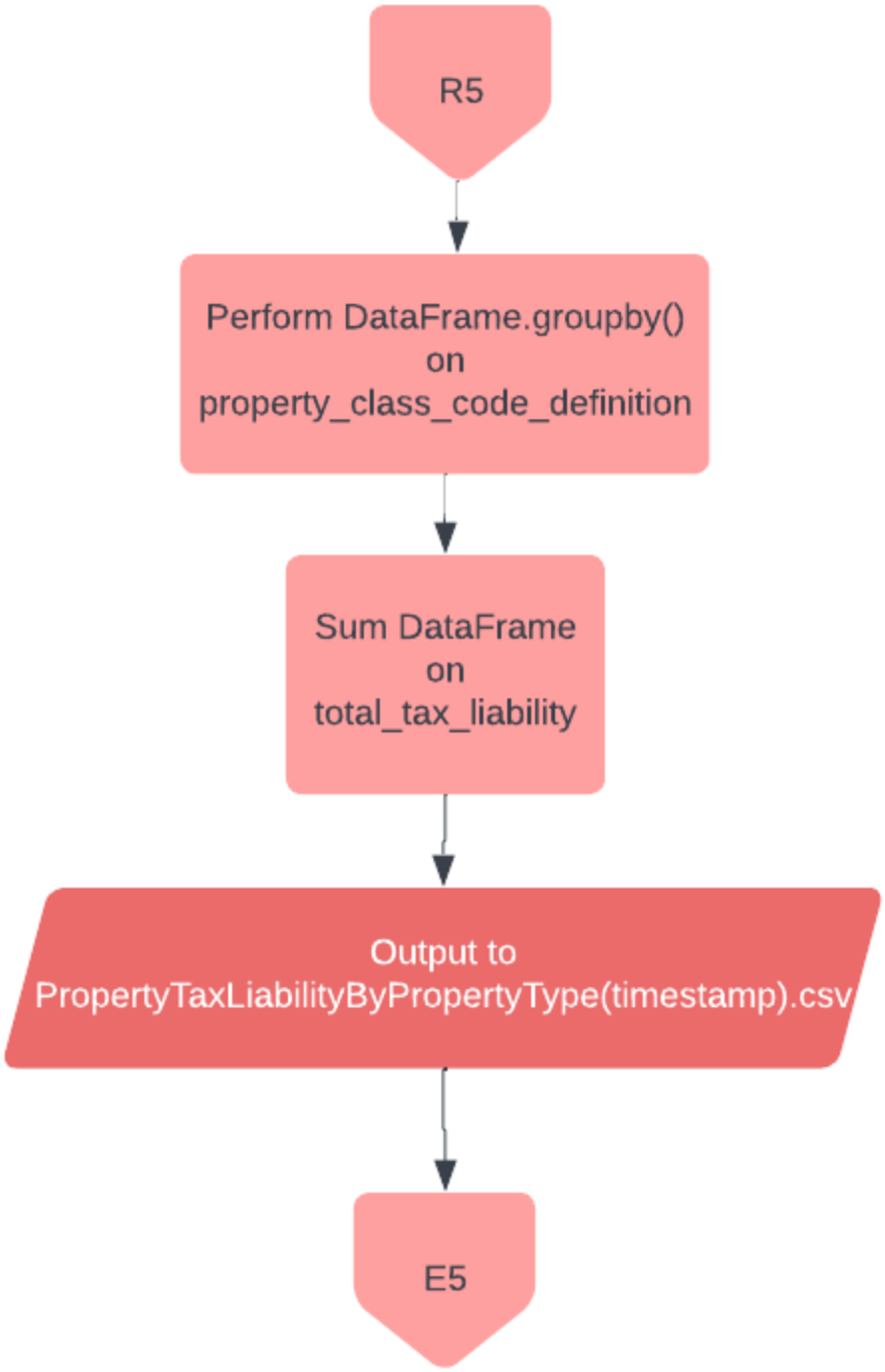


Figure 9: Report 5 (Aggregate report: Property tax per city)

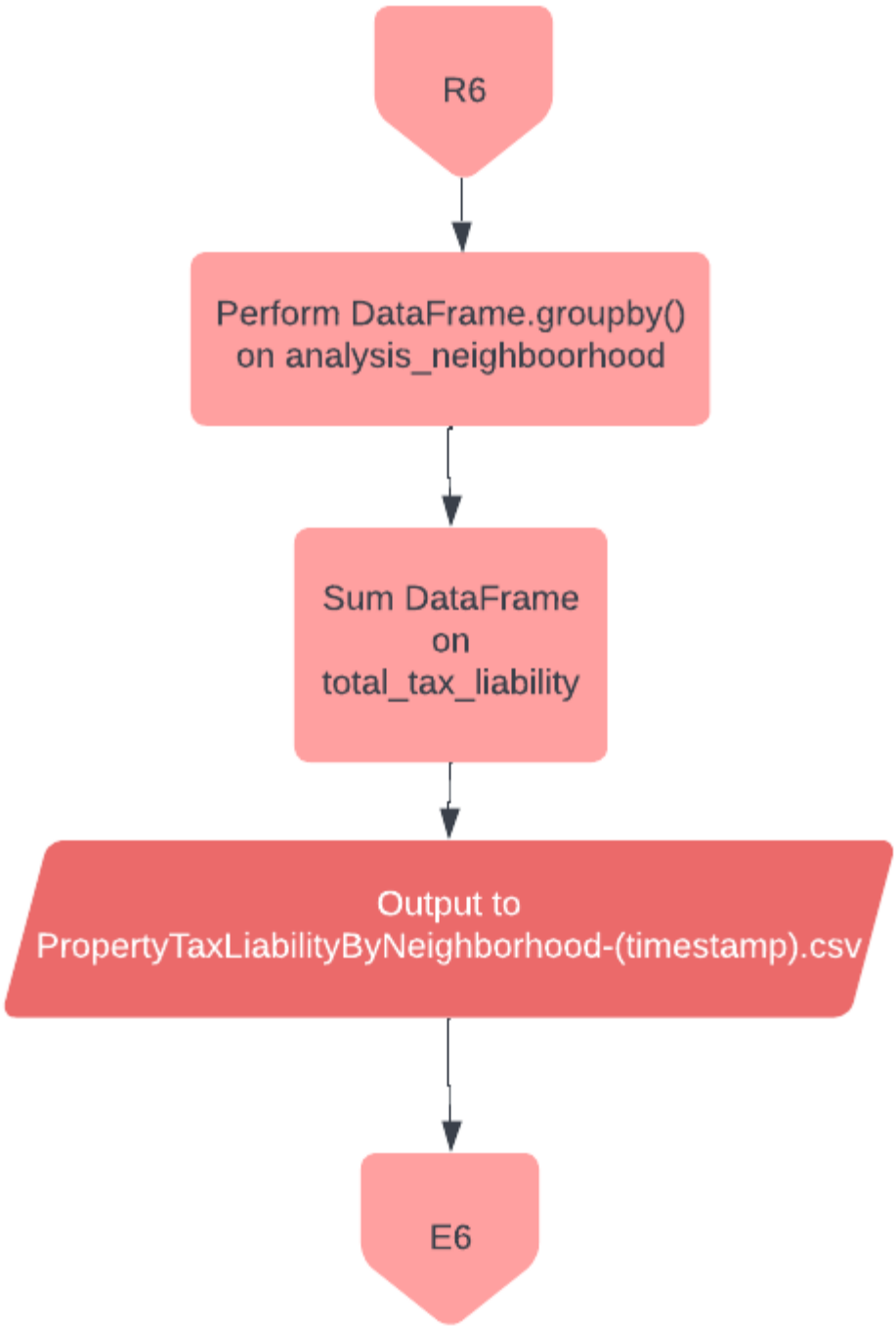


Figure 10: Report 6 (Aggregate report: Property tax per zip code)

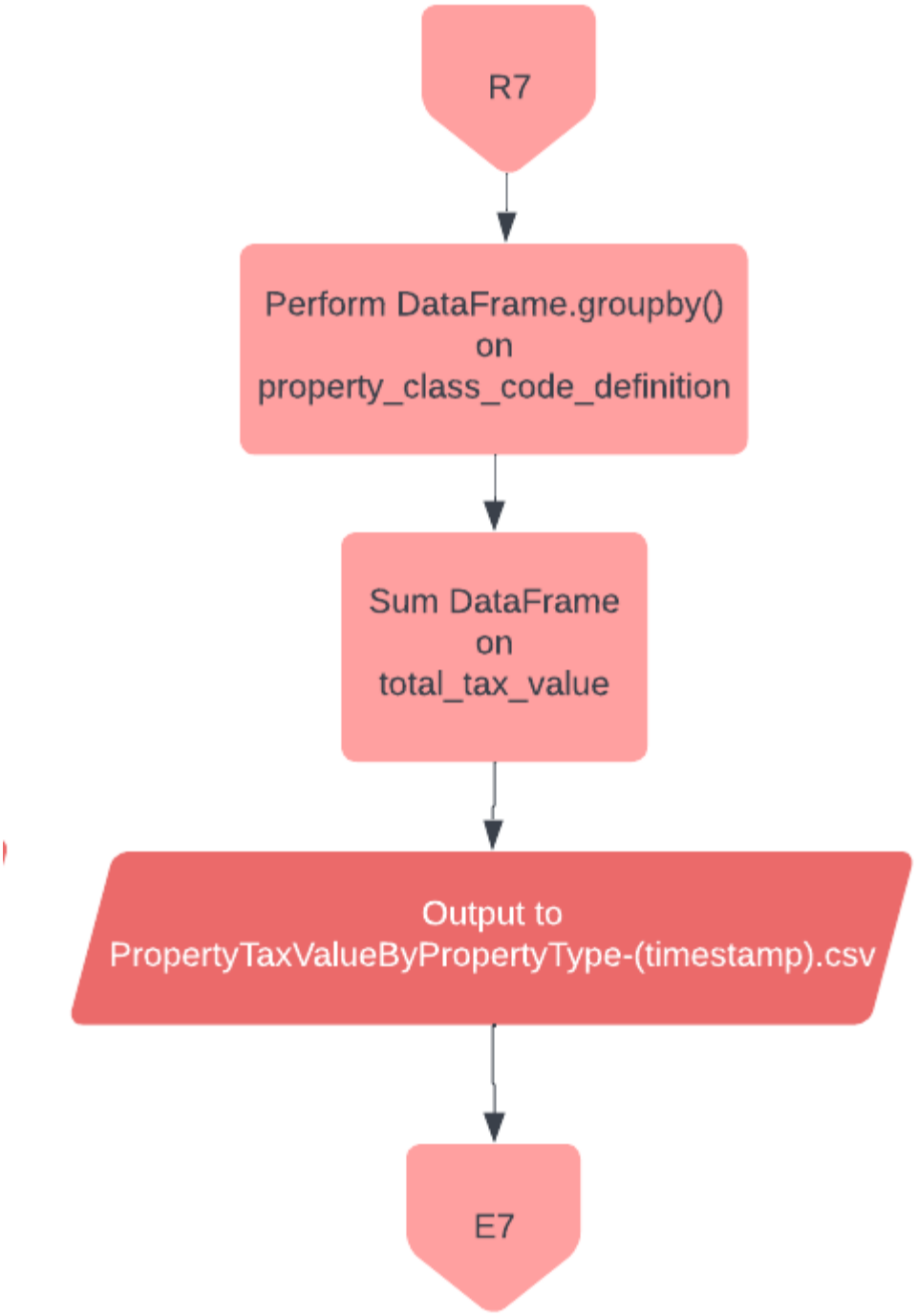


Figure 11: Report 7 (Aggregate report: Property value per city)

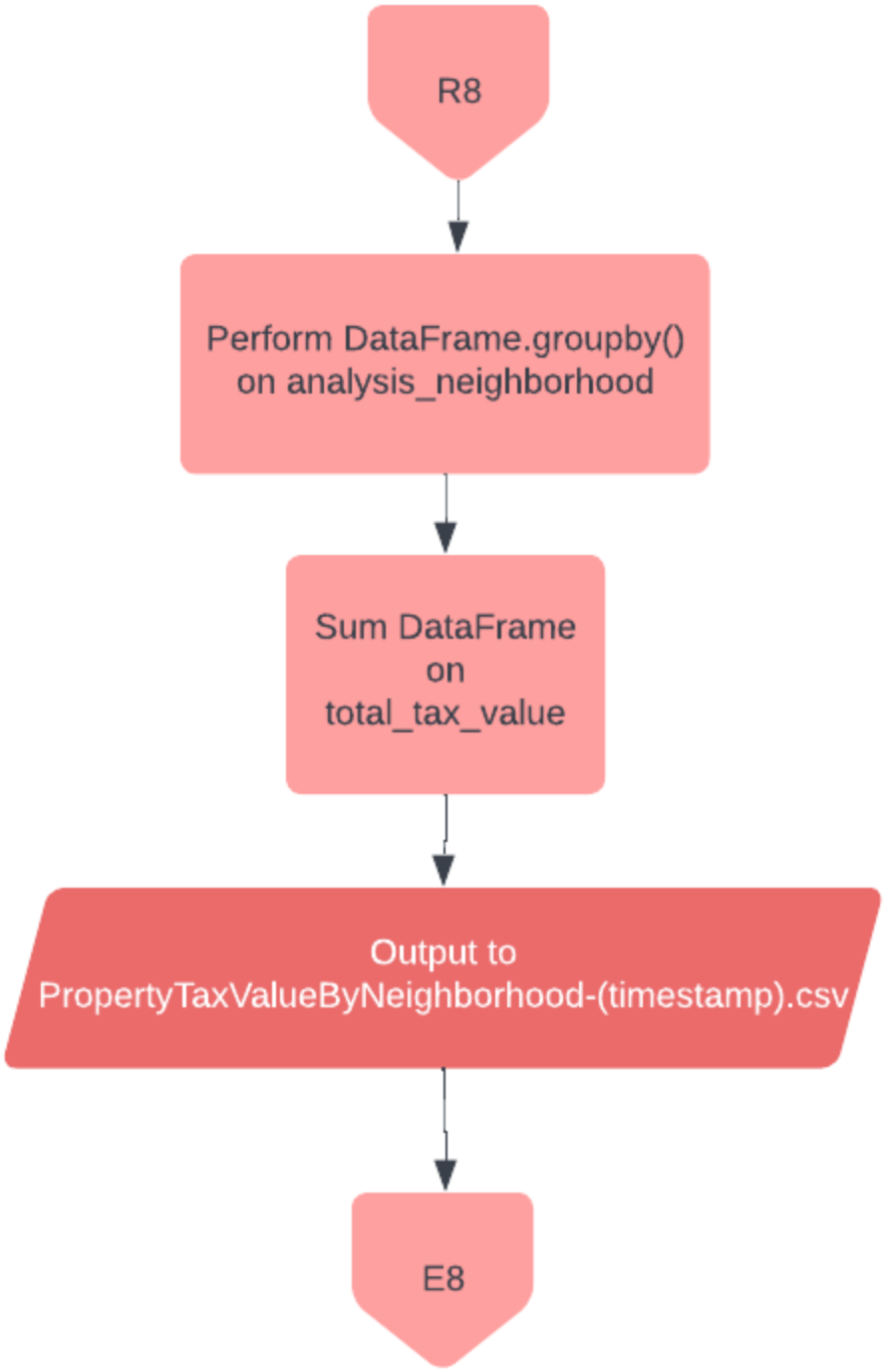


Figure 12: Report 8 (Aggregate report: Property value per zipcode)

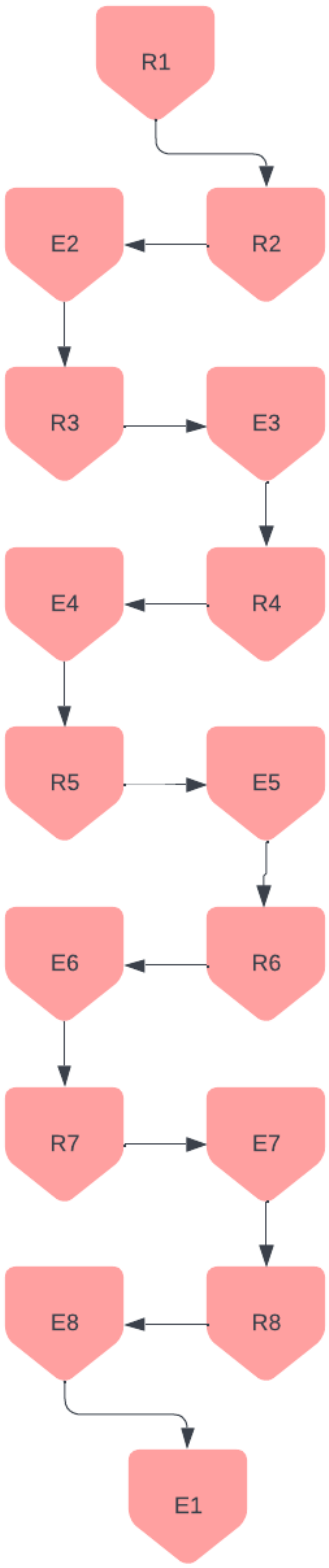


*Figure 13: Report 1 (**All reports**)*

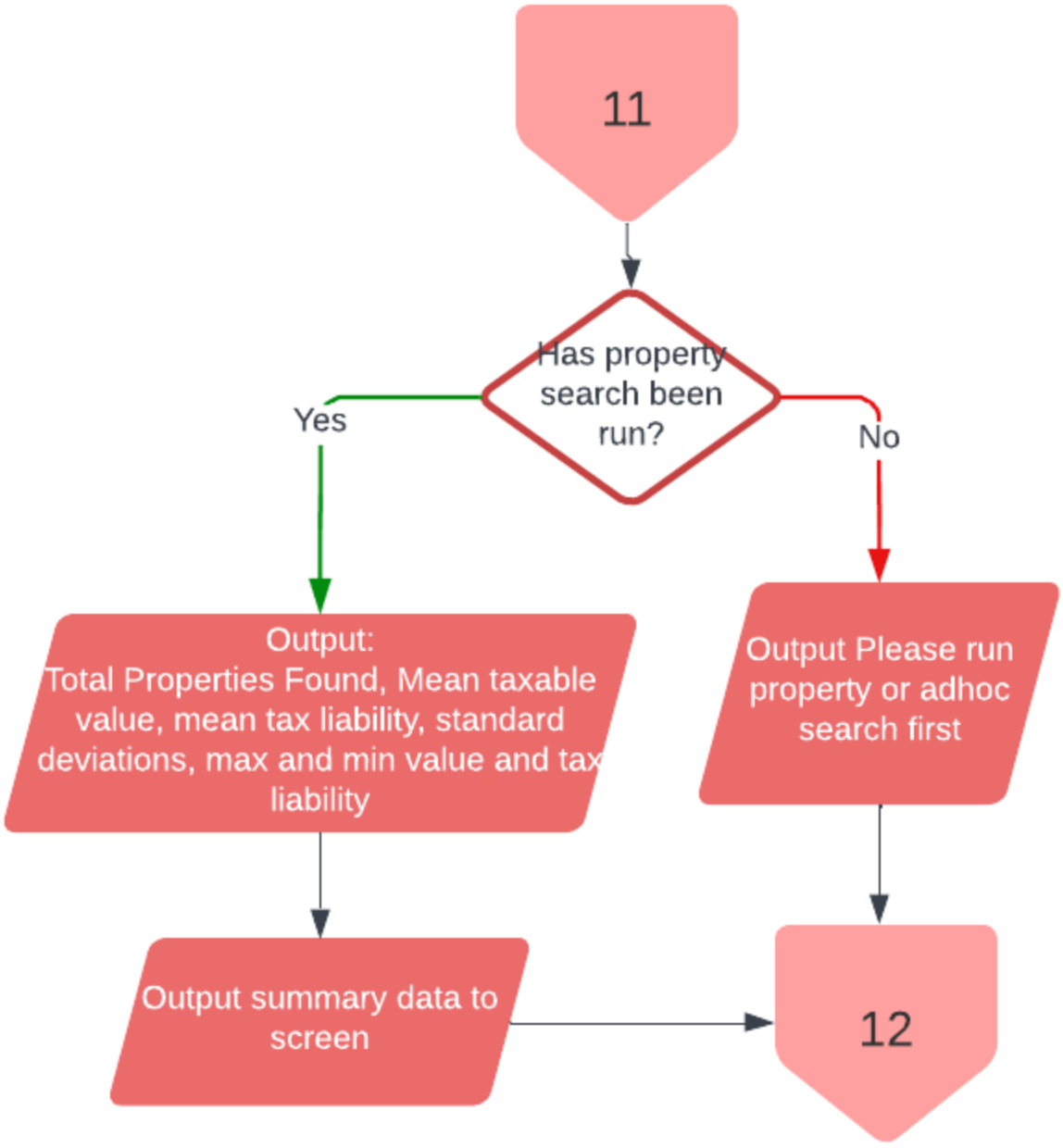


Figure 14: Summary Report Only

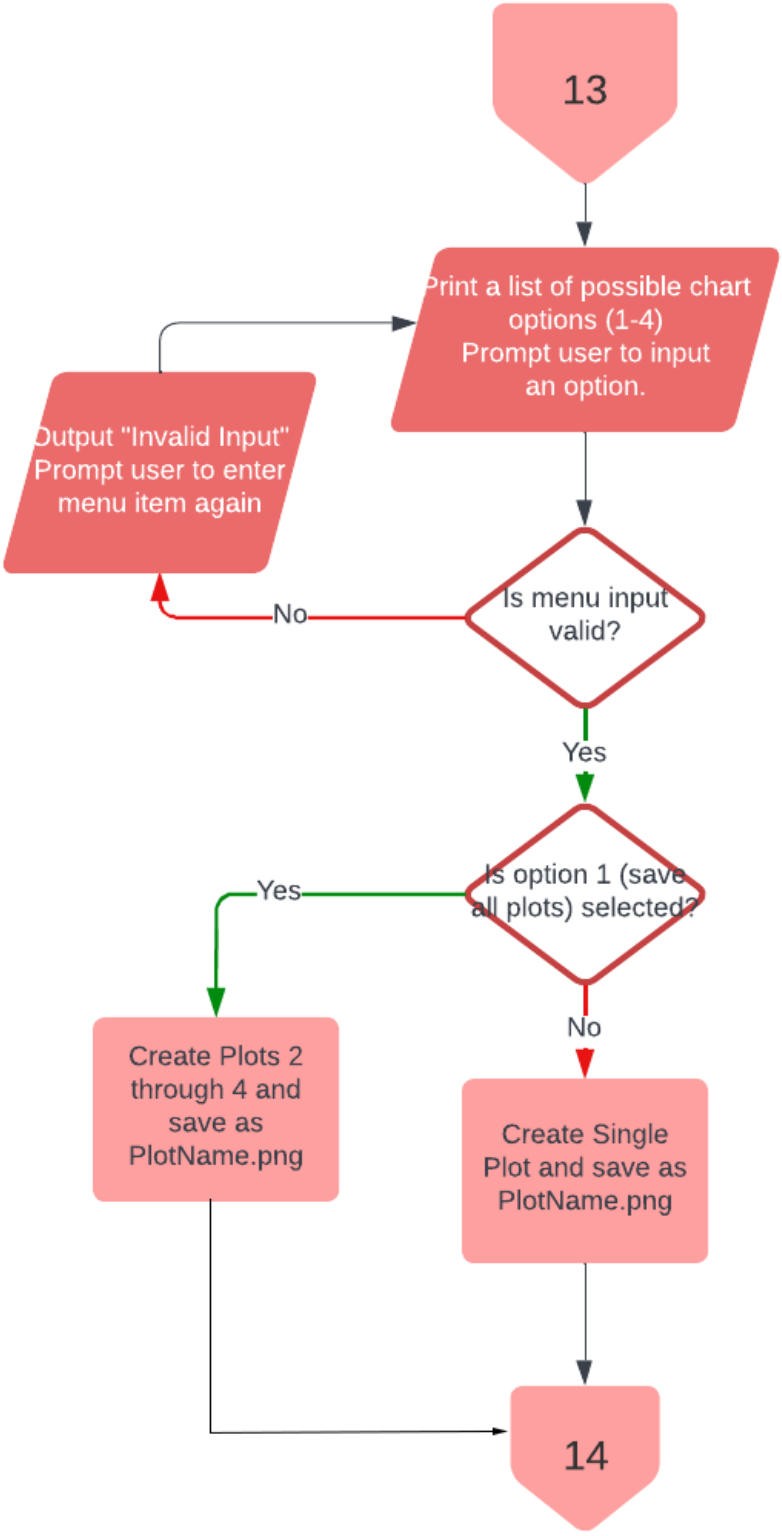


Figure 15: Generate Plots

Data Design

The data for this project is saved in CSVs that will be read by the program. Output will be generated in a txt file after manipulating data from the CSV. The sample data currently has 43 columns and over two million rows. Not all these columns will be used by the program, and the requisite columns are represented in the data dictionary below. Using the data dictionary to standardize file formats will allow the business to apply this program to any CSV file by renaming columns to the column headers found in the dictionary. The source for all the data is currently the same CSV file from the City of San Francisco for historical secured property tax rolls.

Data Dictionary: Columns Used by Program			
Field Name	Type	Length	Description
assessed_improvement_value	Integer	Variable	Value of buildings and other improvements as assessed by the county tax office.
assessed_land_value	Integer	Variable	Value of the land with no improvements as assessed by the county tax office.
assessed_fixture_value	Integer	Variable	Value of fixed installments, furniture, or equipment as assessed by the county tax office.
assessor_neighborhood_district	Integer	Variable	Integer representing the district of the neighborhood the home is found in.
assessor_neighborhood_code	String	Variable	String representing the neighborhood the home is found in.

assessor_neighborhood	String	Variable	Name of the neighborhood the home is found in.
analysis_neighborhood	String	Variable	Name of the district the neighborhood is found in.
misc_exemption_value	Integer	Variable	Integer representing the tax-exempt value of the property.
tax_rate_area_code	Integer	4	Integer that determines the tax rate paid for the property. The tax rate is determined by dividing this number by 1000.
parcel_number	String	Variable	String identifying a property by parcel number.
property_location	String	Variable	String representing the street address of the property.

Table 1: Data Dictionary

Table 1 shows the columns found in the CSV that are used in the program. These are used to handle property look-up, determination of the total taxable value and exempt value, determination of the tax rate, and aggregation of data in reports. Property look-up is shown in figure 2 and 6 and can search for the input parcel number or street address in the parcel_number and property_location columns after importing the CSV, as shown in figure 4. The program adds assessed_land_value, assessed_fixture_value, and assessed_improvement_value to determine the taxable value. The tax_rate_area_code determines the tax rate for the property, and the taxes due is determined by multiplying the tax rate by the taxable value and subtracting the misc_exemption_value.

Reports output by this program will be txt files containing the property location, parcel number, total taxable value, and taxes due. Aggregation of this information will be supported for neighborhoods and districts which are identified by the assessor_neighborhood and analysis_neighborhood respectively. In the case of

aggregation, the property location and parcel number will be replaced by the neighborhood or district, and the property and tax values will represent the entirety of that neighborhood or district.

Project Fulfillment Estimation

Variables:

parcelNumber : User input
address: User input
propertyList: User input (file name)
dataset: User input (file name)

Flow Control:

If/else condition:

1. The program allows users to select an option (multiple options in the future) from the menu list. It verifies and asks the user to submit the correct input if they enter an invalid value.
2. We are using If/else conditions to validate the user input data of zip code. We are checking if the input value is numeric or not. If true it will set the flag useAddress to True.
3. The programming is using the nested if/else statement to check the user's input and based on that, it is selecting the function that needs to be executed.
4. One if statement is used inside a for loop to check if the city is present in the keys of cityValue dictionary.
5. Similarly, many if/else statements are used to check the condition as above.

for/while loop:

1. This program uses a while loop to validate the user's input until the user enters the correct input.
2. This program is using the while loop to read the data from the file using the readline method, until the end of the file.
3. This program uses a for loop to select property from propertyArray and adds the property value, address, and parcel to the list.
4. For loop is used on propertyArray to choose property and using the if condition to check if it is in the cityTax dictionary's key.
5. Similarly many for and while loop is used in the program for various purposes like above.

Functions:

displayMainMenu: This function is used to print the main menu, and validate the user's menu input.

propertySearch: This function is used to search the property from the datasets available for the property. If the dataset is larger than 50 megabytes, the function will iterate through the dataset rather than parse it directly to memory.

generateReport: This function generates one or all reports based on the user's input.

Lists and Dictionaries, and/or data frame:

1. This program uses a list named propertyArray to store the property information in the form of a dictionary (a list of dictionaries).
2. Used a dictionary named zipValue to store the value of the property for the particular property in the city.

File or DBMS operations:

1. This program uses CSV files to store the data of the property. These data are retrieved to show the requested information from the user.

Data Computation and Visualization:

1. Different types of reports are used here in the program.
2. Total of 7 different types of reports are used here in the program. It will display the report based on the request data by the user. The user can also choose to generate all the reports at once.

Data visualization:
Currently, our program provides no visualization, but this may change in a future version. Visualization can be provided by any program that can import CSVs and visualize the data, like Excel, etc.

USER MANUAL

Purpose:

This User Manual provides users with the necessary information to employ the system. This manual includes a description of the system functions and step-by-step instructions on how to access it.

Scope:

This Information System (IS) is intended to provide an efficient way of finding data by selecting data from a file containing property values or property taxes based on the city or the zip code. Using this IS, the company can also generate reports based on some characteristics.
This system reduces the man hours, the errors caused by human indulgence, and the time taken to search for the data.

System Functions:

1) Ad Hoc Property Search

2) Import Property List

3) Import Data Set

4) Run Property Search

5) Generate Report(s)

6) Display Summary Report

7) Generate Plots

Function Descriptions:

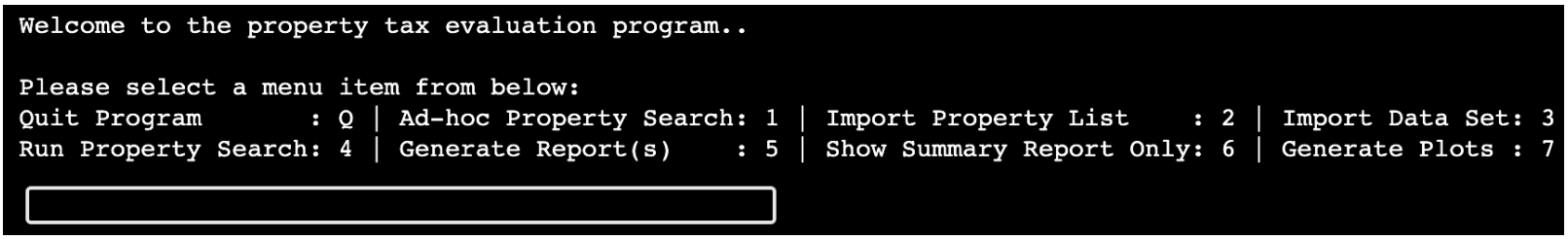
- *Ad Hoc Property Search:* This function asks the user to property search on the basis of Address or Parcel Number.
- *Import Property List:* This function allows the user to import a property list dataset in ".csv" format. This function also allows the use of CRUD.
- *Import Data Set:* This function allows the user to import a tax data set in ".csv" format.
- *Run Property Search:* This function allows the user to search for a property on the basis of some characteristics and generates a short table of summary data.
- *Generate Report(s):* This function generates reports available from the report types according to the requirement of the user and based on the Property Search.
- *Summary Report:* This function allows the user to view the summarized version of the reports they have chosen to generate by displaying the total summary, mean, standard deviation, and the range.

- Generate Plots: This function allows the user to generate some visualizations including pie charts showing the total taxable value and tax liability aggregated by neighborhood or use code and a histogram showing the distribution of taxable values. The user can choose to save these plots.

This section illustrates the step-by-step process of using the system from a user point of view.

Main Step:

The user is welcomed with an introduction message, a menu that contains all the items that are offered by the program, and a dialog box that takes the input regarding the user's choice of menu item.



The program then takes the user to the corresponding process of the item that is given as input by the user.

Process 0 - "Quit Program":

At any point in time, whenever the main menu appears at the user's screen, the user has the option of quitting the program immediately.

If the user enters "q" on the dialog box, He/She gets a confirmation message whether to stop the program or not.

Then, If the user enters "y", the program quits.

```
Welcome to the property tax evaluation program..

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
q
-- ARE YOU SURE YOU WANT TO QUIT? (y/N) --
y
```

Process 1 - "Ad Hoc Property Search":

This process is implemented when the user selects "1" i.e. "Ad-hoc Property Search".

```
Welcome to the property tax evaluation program..

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
1
You have selected menu item: 1


-- Please import the property data set first. Menu item 3--



Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

For this process to run, the user has to import the data set required for the search i.e the user has to run process 3 (“Import Data Set”).
The user should select a file to run.

```
Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
3
You have selected menu item: 3

Please enter the path and filename for the property tax dataset (in CSV format):
SampleData-100records.csv
Please wait, this may take several seconds...


Property tax dataset imported. Checking for minimum required data...
Tax dataset has been successfully imported.

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

After importing the data set, the user inputs “1” again for the Ad-hoc property search and is asked to search by either an address or a parcel number.

```
Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
1
You have selected menu item: 1


Please enter search type. 1 for address search. 2 for parcel number search:
```

Process 1, Branch 1:

If the user selects to search by address i.e. inputs “1”:
He/She is asked to enter the address in a dialog box.

```
You have selected menu item: 1

Please enter search type. 1 for address search. 2 for parcel number search: 1

Enter an address to search. Do NOT input the street type (st, ave, etc.):
```

The user should enter an address excluding the street abbreviations like “st”, “ave”, etc.

```
Please enter search type. 1 for address search. 2 for parcel number search: 1
Enter an address to search. Do NOT input the street type (st, ave, etc.): 0000 2655 HYDE                ST0B04

Property found!
```

The user is then asked to enter a tax year for the property.

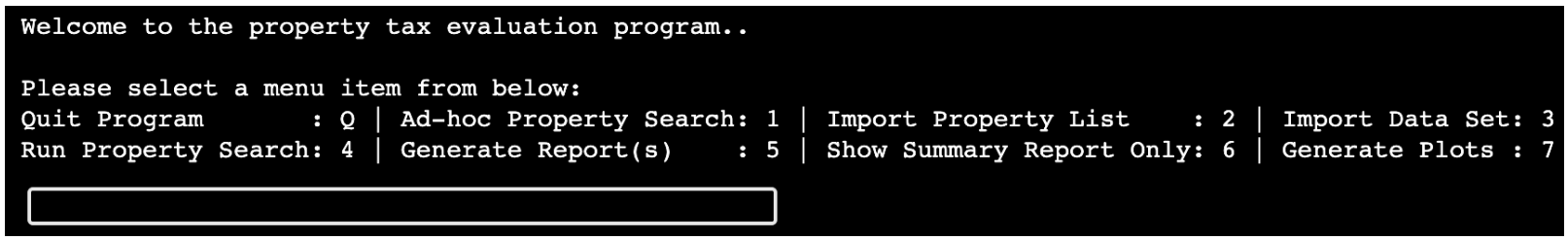
```
Property found!
Please enter a tax year:
2014
```

After entering the year, the tax data for that area and for that year is displayed.

```
Property found!
Please enter a tax year:
2013
Total properties found in dataset     : 1
Total Fixtures Taxable Value          : $0
Total Improvements Taxable Value      : $700
Total Land Taxable Value              : $700
```

Now, the Ad-hoc property search for the address is done.

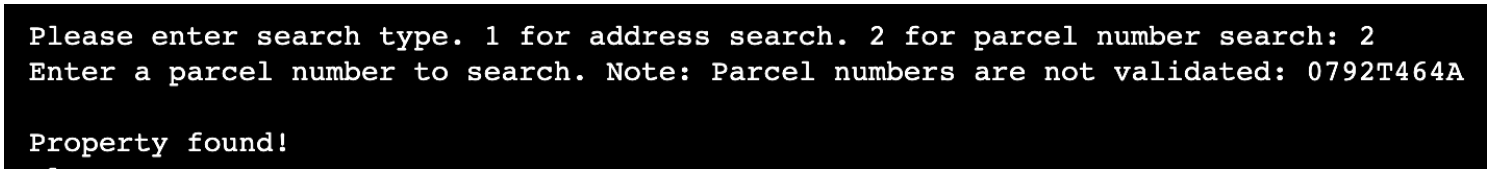
The user is then asked to select any of the menu items again in order to proceed into the process for that item with the data that the user has inputted in process 1.



Process 1, Branch 2:

If the user selects to search by Parcel Number:
He/She is asked to enter the number in a dialog box.

The user then should enter the parcel number in the dialog box.



The user is then asked to enter the tax year for that parcel number.

```
Property found!
Please enter a tax year:
2014
```

After entering the year, the tax data for that parcel number and in that year is displayed.

```
Total properties found in dataset      : 1
Total Fixtures Taxable Value           : $0
Total Improvements Taxable Value       : $1,400
Total Land Taxable Value               : $1,400
Total Personal Property Taxable Value  : $0
Total Taxable Value                    : $2,800
Total Tax Liability                    : $32
```

Now, the Ad-hoc property search for the parcel number is done.

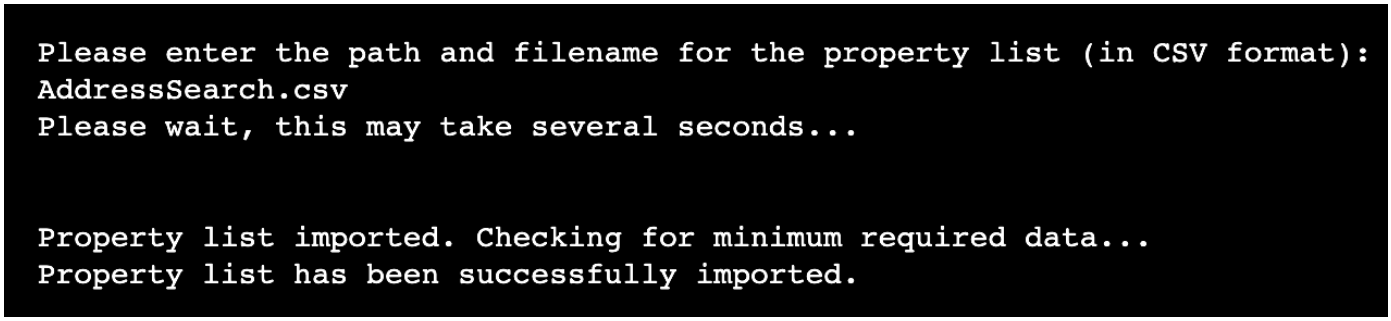
The user is then asked to select any of the menu items again in order to proceed into the process for that item with the data that the user has inputted in process 1.

```
Welcome to the property tax evaluation program..

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

Process 2 - “Import Property List”:

This process is implemented when the user selects “2” i.e. “ Import Property List”.

The user should select a file to import into the program and the file is imported successfully, if it is found.



The user is then asked whether he/she wants to update or delete any record from the file.

Process 2, Branch 1:

If the user enters “y”:
He/She is asked to enter “1” to update and “2” to delete a record.

```
Property list imported. Checking for minimum required data...
Property list has been successfully imported.
Would you like to update or delete any records? (Y/N)
y
Enter 1 to update a record or 2 to delete a record, or any other input to cancel.
```

Process 2, Branch 1(a):

If the user chooses to enter “1” (update):
He/She is asked to enter the address or parcel number they wish to update

```
Would you like to update or delete any records? (Y/N)
y
Enter 1 to update a record or 2 to delete a record, or any other input to cancel.
1
Please type the address or parcel number you wish to update:
```

After choosing a file name, the user is asked to enter the address or parcel number that they want the previous entry to update to

```
Enter 1 to update a record or 2 to delete a record, or any other input to cancel.
1
Please type the address or parcel number you wish to update:
0000 0441 MASON               ST4-06

Property found!

Please enter what you would like to update the address to:
```

The user should input a value. Next, they are asked if they would like to save the file. If the user types “y”, a new file will be created with the updated change. If they enter “n”, the file will not be saved.

```
Please type the address or parcel number you wish to update:
0000 0441 MASON                ST4-06

Property found!

Please enter what you would like to update the address to:
0000 0441 MASON ST4-06
Would you like to save the file? (Y/N)
y
File written to: AddressSearch-Updated.csv
```

Process 2, Branch 1(b):

If the user chooses to enter “2” (delete):
He/She is asked to enter the address or parcel number they wish to update

```
Property list imported. Checking for minimum required data...
Property list has been successfully imported.
Would you like to update or delete any records? (Y/N)
y
Enter 1 to update a record or 2 to delete a record, or any other input to cancel.
2
Please type the address or parcel number you wish to delete:
```

After entering the address/parcel number, the user gets asked if they are sure about deleting the property. If the user enters “y”, the property is deleted. If they enter “n”, the property is not deleted.

```
Please type the address or parcel number you wish to delete:
0000 0710 POWELL              ST0000

Property found! Are you SURE you want to delete the property? (Y/N)
```

enters “n”, the data is not saved.

```
Property found! Are you SURE you want to delete the property? (Y/N)
y
Would you like to save the file? (Y/N)
y
File written to: AddressSearch-Updated.csv
```

Process 2, Branch 2:

If the user enters “n”:
The main menu is displayed and the user is again asked to select one of the menu items that considers the inputted property list when processing.

```
Property list imported. Checking for minimum required data...
Property list has been successfully imported.
Would you like to update or delete any records? (Y/N)
n

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

Process 3 - “Import Data Set”:

This process is implemented when the user selects “3” i.e. “ Import Data Set”.

```
You have selected menu item: 3

Please enter the path and filename for the property tax dataset (in CSV format):
```

The user selects a file to import into the program and the file is imported successfully if the file is found.

```
Please enter the path and filename for the property tax dataset (in CSV format):
SampleData-100records.csv
Please wait, this may take several seconds...


Property tax dataset imported. Checking for minimum required data...
Tax dataset has been successfully imported.
```

The user is again asked to select one of the menu items that considers the inputted data set when processing.

```
Welcome to the property tax evaluation program..

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

Process 4 - “Run Property Search”:

This process is implemented when the user selects “4” i.e. “Run Property Search”.

For this process to run, the user has to import the data set required for the search i.e the user has to run process 3 (“Import Data Set”).

```
Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
4
You have selected menu item: 4


-- Please import the property data set first. Menu item 3--



Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

The user should select a file to import into the program and the file is imported successfully, if it is found.

```
Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
3
You have selected menu item: 3

Please enter the path and filename for the property tax dataset (in CSV format):
SampleData-100records.csv
Please wait, this may take several seconds...


Property tax dataset imported. Checking for minimum required data...
Tax dataset has been successfully imported.

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

For this program to run, the user has to also import the property list dataset (Menu Item #2):

```
Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
4
You have selected menu item: 4


-- Please import the property search list first. Menu item 2--



Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

The user selects a file to run. It could be either address dataset or parcel number dataset and they are asked if they want to update or delete any record from the file. The user can choose “y” or “n”.(Look at process 2 for y)

```
2
You have selected menu item: 2

Please enter the path and filename for the property list (in CSV format):
AddressSearch.csv
Please wait, this may take several seconds...


Property list imported. Checking for minimum required data...
Property list has been successfully imported.
Would you like to update or delete any records? (Y/N)
n

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

If The user hits “4” again, the property search runs with the imported datasets.
The program then shows the property and outputs the results to the user.

```
You have selected menu item: 4

Currently, only data files that are all addresses or all parcel numbers (not a mix) are supported
Please wait, this may take several seconds...


Total properties found in dataset     : 18
Total Fixtures Taxable Value          : $0
Total Improvements Taxable Value      : $1,299,241
Total Land Taxable Value              : $2,350,258
Total Personal Property Taxable Value : $0
Total Taxable Value                   : $3,649,499
Total Tax Liability                   : $42,778
```

Process 5 - “Generate Report(s)”:

This process is implemented when the user selects “5” i.e. “Generate Report(s)”.

For this process to run, the user has to import the data set required for the search i.e the user has to run process 4 (“Run Property Search”).

```
Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
5
You have selected menu item: 5

Please execute property search before generating reports. Note: Plots are not available for ad-hoc searches

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

The user has to follow the whole procedure of “Process 4 - Run Property Search” in order to be able to run this process because we need the output data from the property search to generate report(s).

After following the procedure of Process 4, when the user hits “5” again, He/She is provided with 8 types of reports to choose from:

```
Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
5
You have selected menu item: 5

Please execute property search before creating plots. Note: Plots are not available for ad-hoc searches
Please choose a report option:

Generate all reports               : 1
Property Values Only               : 2
Property Tax Owed Only             : 3
Property Values and Tax Owed       : 4
Total Tax By Property Type         : 5
Total Tax Per Neighborhood         : 6
Property Tax Value By Property Type: 7
Property Tax Value By Neighborhood : 8
```

If the user enters “1”, All the reports are generated and saved to the current working directory. The generated reports include:

Property Values report, Property Tax Owed report, Property Values and Tax Owed report (combined), Total Tax by Property Type report, Total Tax Per Neighborhood report, Property Tax Value By Property Type report, and Property Tax Value By Neighborhood.

```
Generate all reports               : 1
Property Values Only               : 2
Property Tax Owed Only             : 3
Property Values and Tax Owed       : 4
Total Tax By Property Type         : 5
Total Tax Per Neighborhood         : 6
Property Tax Value By Property Type: 7
Property Tax Value By Neighborhood : 8
1
Report generated and saved to:  PropertyValuesOnly-1670315119.csv
Report generated and saved to:  PropertyTaxOnly-1670315119.csv
Report generated and saved to:  PropertyValuesandTax-1670315119.csv
Report generated and saved to:  PropertyTaxLiabilityByPropertyType-1670315119.csv
Report generated and saved to:  PropertyTaxLiabilityByNeighborhood-1670315119.csv
Report generated and saved to:  PropertyTaxValueByPropertyType-1670315119.csv
Report generated and saved to:  PropertyTaxValueByNeighborhood-1670315119.csv

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

If the user wants to generate individual reports instead of all reports, He/She can input numbers in the range of 2-8 according to the user's preference. After entering the desired number, the corresponding report will be generated and saved to the working directory.

Example: Here user enters "5":

```
Generate all reports               : 1
Property Values Only               : 2
Property Tax Owed Only             : 3
Property Values and Tax Owed       : 4
Total Tax By Property Type         : 5
Total Tax Per Neighborhood         : 6
Property Tax Value By Property Type: 7
Property Tax Value By Neighborhood : 8
5
Report generated and saved to:  PropertyTaxLiabilityByPropertyType-1670315866.csv
```

Process 6 - “Show Summary Report Only”:

This process is implemented when the user selects “6” i.e. “Show Summary Report Only”.

In order to run process 6, The user has to import the property search list first, i.e. Menu Item #2.

```
Welcome to the property tax evaluation program..

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
6
You have selected menu item: 6


-- Please import the property search list first. Menu item 2--



Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

The user should select a file to run and is asked whether they want to update or delete any record. (Look at process 2 for ‘y’)

```
2
You have selected menu item: 2

Please enter the path and filename for the property list (in CSV format):
AddressSearch.csv
Please wait, this may take several seconds...
```

The user also has to run the property search (Menu Item #4) to generate the summary report, as any kind of report is dependent on the data from the property search.

```
Property list imported. Checking for minimum required data...
Property list has been successfully imported.

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
6
You have selected menu item: 6


-- Please import the property search list first. Menu item 2--



Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

The user has to follow the whole procedure of “Process 4 - Run Property Search” in order to activate the generation of reports.

After following the procedure, when the user hits “6”,
He/She has the outputs in four tables :

1) Totals/Summations
2) Means
3) Standard Deviations
4) Ranges

All these generate exactly what their names indicate.
Here are all the outputs that the program generates based on the datasets and the property search function:

```
6
You have selected menu item: 6


TOTALS:
Sum of fixtures taxable value          : $0.00
Sum of improvements taxable value      : $1,299,241.00
Sum of land taxable value              : $2,350,258.00
Sum of personal property taxable value: $0.00
TOTAL TAXABLE VALUE                    : $3,649,499.00
TOTAL TAX LIABILITY                    : $42,778.14

MEANS:
Mean of fixtures taxable value          : $0.00
Mean of improvements taxable value      : $72,180.06
Mean of land taxable value              : $130,569.89
Mean of personal property taxable value: $0.00
MEAN OF TOTAL TAXABLE VALUE             : $202,749.94
MEAN OF TOTAL TAX LIABILITY             : $2,376.56
```

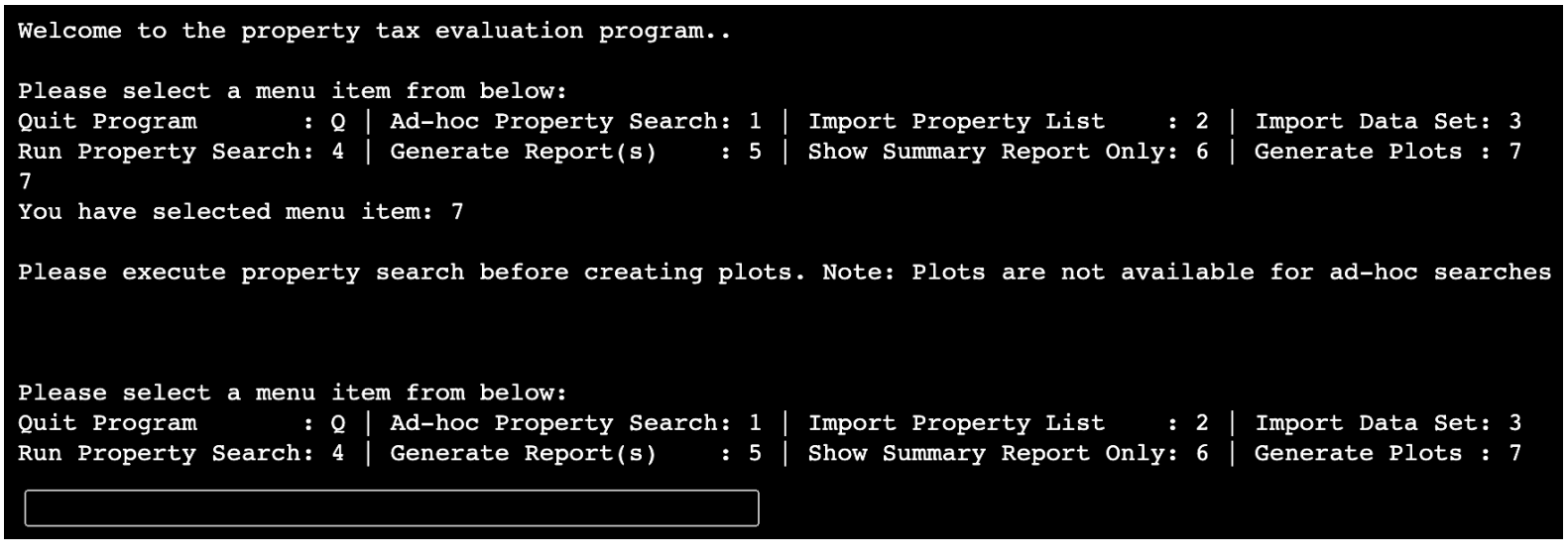
```
STANDARD DEVIATIONS:
STD of fixtures taxable value          : $0.00
STD of improvements taxable value      : $257,101.60
STD of land taxable value              : $401,195.05
STD of personal property taxable value: $0.00
STD OF TOTAL TAXABLE VALUE             : $648,215.59
STD OF TOTAL TAX LIABILITY             : $7,636.63

RANGES:
Max taxable property value: $2,738,987.00
Min taxable property value: $0.00
Max tax liability          : $32,298.13
Min tax liability          : $0.00

Please select a menu item from below:
```

Then, the user is prompted to select a menu item again.

Process 7 - “Generate Plots”:

This process is implemented when the user selects “7” i.e. “Generate Plots”.

In order to run process 7, The user has to run the property search first, i.e. Menu Item #4 to generate the plots, as the plots are dependent on the data from the property search.



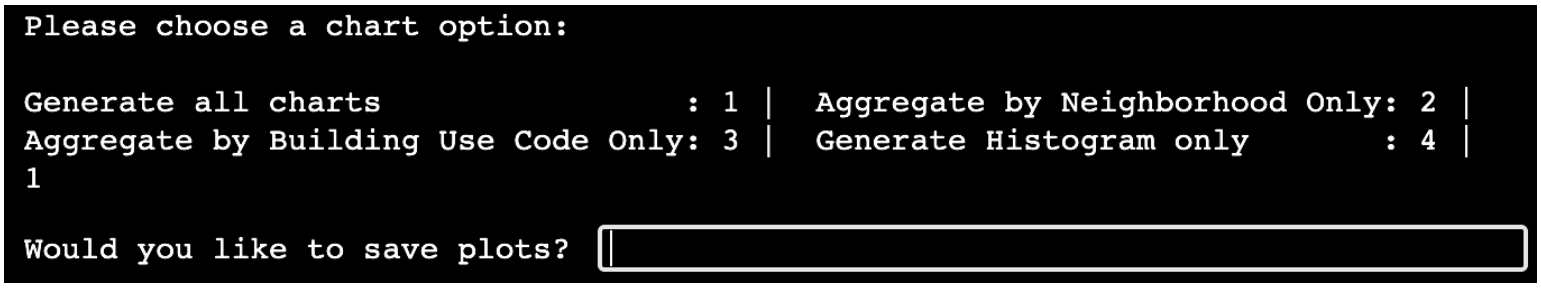
After following the steps in Process “4” and inputting “7” into the dialog box,

The user has 4 options to choose from, those are:

1) Generate All Charts
2) Aggregate by Neighborhood Only
3) Aggregate by Building Use Code Only
4) Generate Histogram Only

If the user inputs any of the four numbers, He/She is asked if they want to save that particular plot.

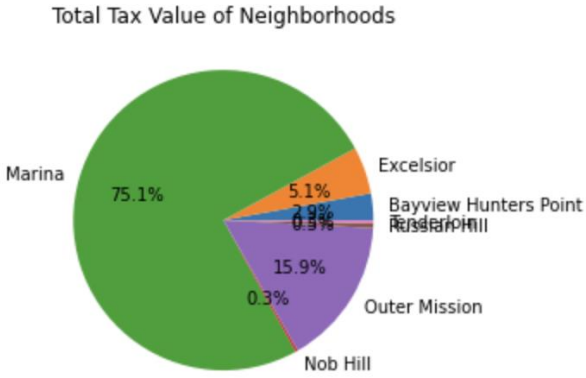
Example: If the user hits “1”:
He/She has to choose whether to save the plot or not:

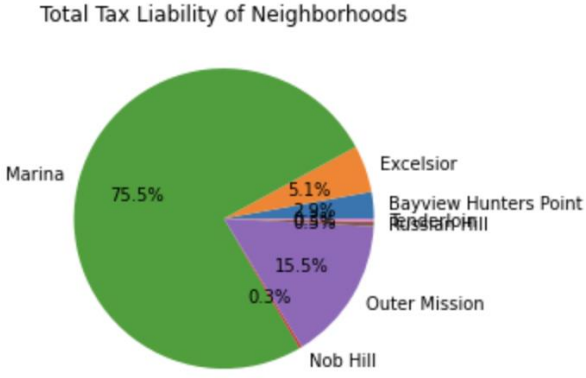


If the user inputs “y”:

The charts are generated and the user is asked again if he/she wants to save the plots.

```
Would you like to save plots? y
```






```
Would you like to save plots?
```

If the user inputs “y”:
All plots are generated and saved and the user is asked again to select a menu item if he/she wants to.

```
Would you like to save plots? y
Would you like to save plots? y

Please select a menu item from below:
Quit Program       : Q | Ad-hoc Property Search: 1 | Import Property List    : 2 | Import Data Set: 3
Run Property Search: 4 | Generate Report(s)    : 5 | Show Summary Report Only: 6 | Generate Plots : 7
```

—----------END OF THE MANUAL—----------

Response Document

Group 3:
Curtis Wa
Jason Jasper
Shanthan Venkat Pochampally
Kumar Rishav
Luke Stevens

After presenting to our fellow students, we received a helpful amount of feedback from the other students. Here are our responses

Students' Comments

1. The group should have practiced their presentation beforehand to summarize and present their project idea better, to be able to complete in the allotted time. Also the presentation need not be detailed because you might lose the audience if it gets too technical. Also the flow charts on the decision boxes had 2 lines coming in, it would have been better if they used the circular reference connector.

Our Response: They are right. We should have practiced our presentation a bit more and we did underestimate the time we needed for the presentation. We should have done a less technical high-level overview of our flowchart diagrams.

2. Have you considered using a property search based on the property amount as well? I think searching properties based on their amount would

be more effective for users who want to search properties within a certain range.

Our Response: Yes, that is something that we have considered. We were thinking about adding it to our final code.

3. The only suggestion is to be sure the presentation slides do not have too many paragraphs as that was a bit hard to follow.

Our Response: We did have a very diagram heavy presentation and that should be altered for next time.

4. I wish they could briefly explain or show how the data of selling or purchasing properties would be entered automatically. Also, this information could be somewhat confidential, so I found it helpful if they added a login function in the program.

Our Response: The addition of purchasing and selling property would be a great function to add as well as logins. We could add this in the future to increase security for our document

Professor's Comments

1. Just a reminder that the data operation should also include delete and update. You may consider updating or deleting some data from the imported file.
Data visualization is required. Please make sure you include charts in the program.

Response: Yes, we will be adding the final portion of CRUD to our design. We have also added data visualization to our project.