CALGARY HOUSING PRICE and COMMUNITY EXPLORATION

Introduction

On June 21st, 2019, the Governor General of Canada officially signed into law Bill C-97, which formally recognized the right to adequate housing and shelter as a part of federal legislation Canadian Housing & Renewal Association, 2019). The reality for Canadians today, is that home ownership and property prices mean much more than a right to shelter and lodging, but rather for many, an instrument of investment and a proxy for household wealth. According to a recent poll conducted by Ipsos, over 77% of Canadian household wealth, as measured by assets, are linked directly to the value of real estates; this ratio is as high as 89% for young adults (Carrick, 2022).
For many, the Covid-fueled rally in home prices have proved to a windfall, but current macro-economic events are putting significant downward pressure on both transaction prices as well as homeowner and buyer sentiment alike. Reputable analysts and major financial institutions alike have shared perspective that the housing market will continue to keep over listing priced, but that a move to a more balanced market is possible in the rest of the year. Recent central bank interest rate adjustments have also made headlines, in attempt to utilize monetary policy to cool housing prices by controlling the cost of mortgage debt-service.
Our project looks to take a closer look at the Calgary real-estate market. Although prices have increased largely in-line with the greater market trends nation-wide, the Calgary market has shown signs of stubbornness in the face of tightening restrictions on loan supplies. While larger markets have shown pricing slowdowns recently, the Calgary resale market remains largely robust and affordable in comparison with wages. Specifically, what factors play into determining the value of a home? To explore this further, our project will investigate recent transaction prices in the Calgary region, in relation to specific location and neighborhood amenities. By taking these datasets into account, we would look to explore whether it is possible to predict the period between an initial listing and a firm sale.

Individual Datasets

• **Real Estate Transaction Data**
Obtained from the Canadian Real Estate Board, our project will investigate 150 days of transaction (sold) data of single-detached homes under $1 Million. The data was provided by a license Calgary real estate agent, solely for this project.

• **Calgary C-Train Station Locations**
From this dataset we can learn the name of the LRT station, the direction of travel, the neighborhood. We predict that most potential buyers will consider the ease of transport when considering a purchase. One study has shown, for example, that a certain range of proximity to a public rail or bus station is likely to influence the value of a property. Specifically, studies have shown that being located between 500-800 meters from a transit station may in fact maximize a home's value (Blair, 2019). This data is available on City of Calgary's Open Data Portal and is publicly available.

• **Community Crime Statistics**
Data is provided monthly by the Calgary Police Service. This dataset contains crime by community in Calgary from 2017 to 2022, including time, community name, community coordinates, community population, type of crime, and the number of crimes. However, some of the missing data prior to 2022 is significant and may impact the final conclusions.

• **Starbucks Locations**
This dataset includes the store locations for Starbucks worldwide in operations as of November 2021, and is publicly available on Kaggle, and was scraped from the Starbucks store locator webpage by Github user KUKUROO3. Addtional updates to the dataset was made by manually scrapping new stores from the Starbucks store locator webpage for an updated dataset for 2022. A study conducted by real estate research group, Zillow, in the United States predicted that there was a positive effect on home prices in areas surrounding a Starbucks location. The study between occurred between 1997 and 2014, which found that properties closer to a Starbucks within a quarter mile increased in value over 96%, compared to 65% of all U.S residential properties (AIP, 2017). Our project aims to explore if the same trend can still be observed in Calgary post Covid-19.

• **School Locations (High Schools within Calgary city boundaries)**
This dataset from City of Calgary's Open Data Portal provides the location of all high schools in Calgary, including school name, address, and board of trustees. We would expect many single-detached homebuyers to be either growing families

looking to upsize, or those looking to move as children begin to reach secondary school-age. Therefore, we would expect school rankings and proximity to good schools to have an influence on average community prices and attract buyers of different socio-economic status and family backgrounds.

In [1]:
```
#Import packages
import pandas as pd
import sqlalchemy as sq
import plotly.express as px
from urllib.request import urlopen
import json
with urlopen('https://raw.githubusercontent.com/plotly/datasets/master/geojson-counties-fips.json') as respo
    counties = json.load(response)
import matplotlib.pyplot as plt
import seaborn as sns
#pip install folium
```

In [ ]:
```
# connect to the database
engine = sq.create_engine('mysql+mysqlconnector://an_yan1:3LJR88NQD@datasciencedb2.ucalgary.ca/an _yan1')
```

Data Exploration

Housing prices and C-Train Station Locations

As described in our initial project proposal, we are looking to observe factors that influence home-prices, with a particular focus on geo-spatial characteristics including proximity to amenities, transportation, schools (and their respective rankings). The LRT data obtained from the City of Calgary public database included LRT coordinates for each station. By using the online tool, GPS Visualizer, I was able to manually geo-code all the home addresses in our data set. Within python, we use the latitude and longitude of each home-sold to identify and tabulate the nearest LRT station for further analysis in our final project.

Data Wrangling, Cleaning and Exploration

In [4]:
```
LRT_df = pd.read_csv("Transit_LRT_Stations.csv")
Address_df = pd.read_csv("Addresses.csv")
LRT_df.head(3)
Address_df.head(3)
# LRT_df['the_geom']
# LRT_df.iloc[0]
```

Out[4]:

	Address	Coordinate
0	4716 6 Street SW	26.24561,-98.21833,4716 6 Street SW,"4716 6th ...
1	18 Crestridge Mews SW	51.08195,-114.26373,18 Crestridge Mews SW,"18 ...
2	60 Mt Alberta Green SE	50.91449,-113.98933,60 Mt Alberta Green SE,"60...

In [5]:
```
# write the dataframe into a table
LRT_df.to_sql('lrt_table', engine, if_exists = 'replace' )
Address_df.to_sql('address_table', engine, if_exists = 'replace' )
```

Out[5]: 4934

In [6]:
```
LRT_import_df = pd.read_sql_table("lrt_table", engine)
LRT_import_df.head(3)
Address_import_df = pd.read_sql_table("address_table", engine)
Address_import_df.head(3)
```

Out[6]:

	index	Address	Coordinate
0	0	4716 6 Street SW	26.24561,-98.21833,4716 6 Street SW,"4716 6th ...
1	1	18 Crestridge Mews SW	51.08195,-114.26373,18 Crestridge Mews SW,"18 ...
2	2	60 Mt Alberta Green SE	50.91449,-113.98933,60 Mt Alberta Green SE,"60...

Query 1 - Retrieve LRT Station Name & Location Coordinates

Required to obtain the name of all LRT stations from the City of Calgary data. This allows us to insert this into our final tabular output, using naming conventions that will be easy to interpret. In this query, we will also be fetching the

the_geom field, which contains longitude and lattitude information. From this, we will use pandas to return only the lat & long values as x & y coordinates to enable distance calculations later on.

In [7]:
```
# Retrieve LRT Station Name
query_name = ("SELECT STATIONNAM, the_geom FROM LRT_table;")
station_names = engine.execute(query_name).fetchall()
```

In [ ]:
```
station_df = pd.DataFrame(columns=['LRT_name', 'RLT_x', 'LRT_y'])
for item in station_names:
    name = item[0]
    point = item[1]
    cord_y = point.split(' ')[1][1:]
    cord_x = point.split(' ')[2][: -1]
    new_row = {'LRT_name': name, 'RLT_x':cord_x, 'LRT_y':cord_y}
    # station_df = pd.concat([station_df, new_df], axis=1,  ignore_index=True)
    station_df = station_df.append(new_row, ignore_index=True)
station_df
```

Query 2 - Retrieve Coordinates for Each Home Sold

Simple query to extract all coordinates for each home sold from our CREB sales data. As mentioned earlier, this data was obtained using a web-based geo-coding platform.

In [9]:
```
# Retrieve Home Address Coordinates

query_name = ("SELECT Coordinate FROM Address_table;")

cords = engine.execute(query_name).fetchall()
print(cords[0])
```

```
('26.24561,-98.21833,4716 6 Street SW,"4716 6th St, McAllen, Hidalgo, TX, US",,MapQuest,address',)
```

In [10]:
```
    name = cords[0]
#     # cord_x = cords[0][1]
#     # cord_y = cords[0][2]

#     # station = [[name, cord_x, cord_y]]
#     # print(station_names[0])
    print('name = ', name[0].split(','))
#     # # print('point = ', point)
#     # print('cord_x = ', cord_x)
#     # print('cord_y = ', cord_y)
```

```
name =  ['26.24561', '-98.21833', '4716 6 Street SW', '"4716 6th St', ' McAllen', ' Hidalgo', ' TX', ' US"', '', 'MapQuest', 'address']
```

Use pandas to perform data wrangling/ELT. We will use the x & y coordinates of each location to calculate the closest LRT station to each home sold.

In [ ]:
```
def findLRTMinIndex(lrt_df, ax, ay):
    num = lrt_df.shape[0]
    X = lrt_df["RLT_x"]
    Y = lrt_df["LRT_y"]

    x = float(X.iloc[0])
    y = float(Y.iloc[0])
    dis2 = (x-ax)*(x-ax) + (y-ay)*(y-ay)
    min_index = 0
    for i in range(num):
        x = float(X.iloc[i])
        y = float(Y.iloc[i])
        temp_dis2 = (x-ax)*(x-ax) + (y-ay)*(y-ay)
        if dis2 > temp_dis2:
            dis2 = temp_dis2
            min_index = i
    # print(lrt_df.iloc[min_index])
    return min_index

#LRT_df = pd.read_csv("LRT_Name_x_y.csv")
Address_df = pd.read_csv("Address_x_y.csv")
num = address_df.shape[0]
Address = address_df["Address"]
X = address_df["A_x"]
Y = address_df["A_y"]

output_df = pd.DataFrame(columns=["Address", "A_x", "A_y", "LRT_Name", "LRT_x", "LRT_y"])
```

```
LRT_NAME = LRT_df["LRT_name"]
LRT_X = LRT_df["RLT_x"]
LRT_Y = LRT_df["LRT_y"]
for i in range(num):
    address = Address.iloc[i]
    x = float(X.iloc[i])
    y = float(Y.iloc[i])
    lrt_min_index = findLRTMinIndex(LRT_df, x, y)
    lrt_name = LRT_NAME.iloc[lrt_min_index]
    lrt_x = LRT_X.iloc[lrt_min_index]
    lrt_y = LRT_Y.iloc[lrt_min_index]

    output_df.loc[i] = [address, X.iloc[i], Y.iloc[i], lrt_name, lrt_x, lrt_y]

output_df.to_csv("Address_LRT_Mapping.csv")
```

In [ ]:
```
address_df = pd.DataFrame(columns=['Address', 'A_x', 'A_y'])
for item in cords:
    # print('item = ' , item)
    cord = item[0].split(',')
    name = cord[2]
    cord_x = cord[0]
    cord_y = cord[1]
#      #print('name = ', name[0].split(','))
#     # station = [[name, cord_x, cord_y]]
#     # print(station_names[0])
#     # print('name = ', name)
#     # # print('point = ', point)
#     # print('cord_x = ', cord_x)
#     # print('cord_y = ', cord_y)
    new_row = {'Address': name, 'A_x':cord_x, 'A_y':cord_y}
    # station_df = pd.concat([station_df, new_df], axis=1,  ignore_index=True)
    address_df = address_df.append(new_row, ignore_index=True)
    # address_df.to_csv("Address_x_y.csv")
address_df
```

Finally, use a mask to return a seperate table containing all addresses that were erroneously mapped to geographic locations outside of Calgary.

In [73]:
```
import pandas as pd
df = pd.read_csv("Addresses.csv")
mask = (df["Coordinate"].str.find("Calgary") < 0)
new_df = df[mask]
new_df.to_csv("Address_without_Calgary.csv")
```

Discussion

Although this section itself does not reveal any insights in and of itself, it is, nevertheless, an important piece of data wrangling that must be performed in order to conduct geospatial analysis. However, a limitation of this section is that we have only considered absolute linear distances, without taking into consideration whether or not the "closest" linear station is actually accessible. By inspection there are several homes that have been "mapped" to LRT stations that are located on opposing riverbanks, for instance, which we would have little to no bearing on purchasing decisions.

Housing and Community Crime Statistics

“One of the long-standing historical challenges to affordable housing in local communities is the fear by local residents that crime will go up and housing values will go down,” (George Tita) Safty is an important factor affecting home prices and community safety, and it is usually negatively correlated with home prices in a community, with high crime rates resulting in lower home prices and low crime rates resulting in higher home prices. Safety is one of the top considerations for home buyers, and the next step is to explore its relationship with home prices by analyzing the crime situation in Calgary.

Data Wrangling, Cleaning and Exploration

In [28]:
```
# read in CSV as dataframes
price=pd.read_csv("transaction_price.csv")
price.head(5)
```

Out[28]:

	time	originalprice	closeprice	WCS	CPI	unemployment	mortgagerate	netmigration	Year	Month
0	2012/1/1	345950	325000	86.47	127.1	5.1	4.31	6054	2012	1
1	2012/2/1	349900	338000	83.04	126.6	5.2	4.23	6054	2012	2
2	2012/3/1	352900	337000	75.01	126.6	5.0	4.21	6054	2012	3
3	2012/4/1	349900	342000	70.40	127.0	4.8	4.36	7025	2012	4
4	2012/5/1	359900	343000	75.10	126.6	4.5	4.35	7025	2012	5

In [29]:
```
crime=pd.read_csv("Community_Crime_Statistics.csv")
crime.head(5)
```

Out[29]:

	Sector	Community Name	Category	Crime Count	Resident Count	Date	Year	Month	long	lat	ID	Communi
0	SOUTH	MAPLE RIDGE	Theft OF Vehicle	1	1916.0	2022/07	2022	JUL	-114.034678	50.957639	2022-JUL-MAPLE RIDGE-Theft OF Vehicle	(-114.0346 50.95763
1	NaN	AMBLETON	Break & Enter - Other Premises	1	NaN	2022/07	2022	JUL	NaN	NaN	2022-JUL-AMBLETON-Break & Enter - Other Premises	
2	NORTHWEST	01K	Assault (Non-domestic)	1	0.0	2022/04	2022	APR	-114.222716	51.168702	2022-APR-01K-Assault (Non-domestic)	(-114.2227 51.1687
3	CENTRE	BANKVIEW	Assault (Non-domestic)	1	5256.0	2022/04	2022	APR	-114.100489	51.034131	2022-APR-BANKVIEW-Assault (Non-domestic)	(-114.1004 51.03413
4	NORTHEAST	FRANKLIN	Theft OF Vehicle	2	0.0	2022/04	2022	APR	-113.987525	51.058927	2022-APR-FRANKLIN-Theft OF Vehicle	(-113.9875 51.05892

In [30]:
```
#Drop all rows with NaN value in columns "long", "lat" and "Sector"
crime=crime.dropna(subset=["long", "lat", "Sector"])
crime.to_csv("new_crime.csv")
new_crime=pd.read_csv("new_crime.csv")
new_crime.head(5)
```

Out[30]:

	Unnamed: 0	Sector	Community Name	Category	Crime Count	Resident Count	Date	Year	Month	long	lat	
0	0	SOUTH	MAPLE RIDGE	Theft OF Vehicle	1	1916.0	2022/07	2022	JUL	-114.034678	50.957639	2022-JU MAI RIDGE-Th OF Vehi
1	2	NORTHWEST	01K	Assault (Non-domestic)	1	0.0	2022/04	2022	APR	-114.222716	51.168702	2022-AI 01K-Assa (No domes
2	3	CENTRE	BANKVIEW	Assault (Non-domestic)	1	5256.0	2022/04	2022	APR	-114.100489	51.034131	2022-AI BANKVIE Assault (No domes
3	4	NORTHEAST	FRANKLIN	Theft OF Vehicle	2	0.0	2022/04	2022	APR	-113.987525	51.058927	2022-AI FRANKL Theft Vehi
4	5	NORTHEAST	MARTINDALE	Violence Other (Non-domestic)	1	14245.0	2022/04	2022	APR	-113.958387	51.117914	2022-AI MARTINDA Violer Other (No domes

In [31]:

```
# write the dataframe into a table
price.to_sql("price", engine, if_exists='replace')
new_crime.to_sql("crime", engine, if_exists='replace')
```

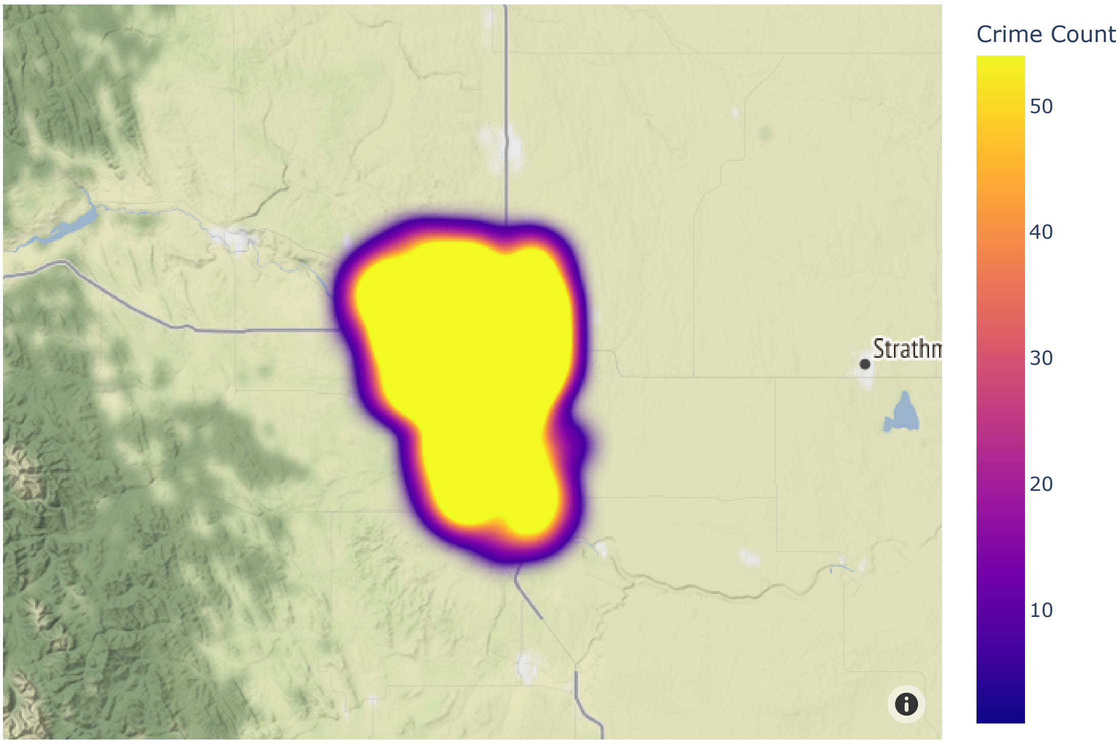
Out[31]: 66462

Query 1- Distribution of crime in each community in 2022

Plot a heatmap to show the dirstiburions of crime happened in 2022 for each community by using Python.

In [32]:

```
#Plot a heatmap to show the distribution of crime in Calgary in 2022
crime_2022=new_crime.loc[new_crime["Year"] == 2022]
coordinate_2022=crime_2022[["lat", "long", "Crime Count"]]
fig=px.density_mapbox(coordinate_2022, lat='lat', lon='long', z='Crime Count',
                        mapbox_style="stamen-terrain", width=800, height=600)
fig
```



Based on this heatmap, we can see that the downtown area has the most crime count, the east part also has lots of crime reported in 2022.

Query 2- Order the crime counts for each community in 2022

Join the two tables by using the column "Year" to explore the relationship between the number of crime and original prices in 2022, house prices in a neighborhood are closely related to the corresponding crime rate as house buyers want to live in a safe and low-crime neighborhood.

In [33]:
```
crime_count=crime[["Community Name", "Crime Count", "Year"]]
crime_count.head(5)
```

Out[33]:

	Community Name	Crime Count	Year
0	MAPLE RIDGE	1	2022
2	01K	1	2022
3	BANKVIEW	1	2022
4	FRANKLIN	2	2022
5	MARTINDALE	1	2022

In [34]:
```
cr_count = pd.read_sql_query('''
                                select crime.`Community Name`, crime.`crime count`, price.Year
                                from price
                                inner join crime
                                on price.Year=crime.Year
                                group by `Community Name`
                                order by `crime count` DESC;
                                ''', engine)
print (cr_count)
```

```
                    Community Name  crime count  Year
0                         BOWNESS           16  2022
1                     FOREST LAWN           12  2022
2    CALGARY INTERNATIONAL AIRPORT            9  2022
3                     MARLBOROUGH            9  2022
4                         RUNDLE             9  2022
..                             ...          ...   ...
299                           03S            1  2017
300                           05E            1  2019
301                           12I            1  2020
302                           05F            1  2021
303                 KEYSTONE HILLS            1  2019

[304 rows x 3 columns]
```

In [35]:
```
crime_count=pd.DataFrame(cr_count, columns=["Community Name", "crime count", "Year"])
crime_count.head(10)
```

Out[35]:

	Community Name	crime count	Year
0	BOWNESS	16	2022
1	FOREST LAWN	12	2022
2	CALGARY INTERNATIONAL AIRPORT	9	2022
3	MARLBOROUGH	9	2022
4	RUNDLE	9	2022
5	RENFREW	8	2022
6	SADDLE RIDGE	8	2022
7	FOREST LAWN INDUSTRIAL	7	2022
8	TUXEDO PARK	6	2022
9	ERIN WOODS	6	2022

The dataframe above shows the top ten community that have the most crime count in Calgary, the first community is Bowness which has 16 crimes happened, the second is Forest Lawn with 12 crimes; the rest of the community crime cases are between 6 and 10. It is obviously that all of these occurred in 2022, even though 2022 is not yet over, which could indicate a safety concern for Calgary in 2022.

Query 3- Distributon of average housing price in ten years

In recent years, housing prices have been affected to a greater or lesser extent due to various reasons, such as epidemics or political factors.

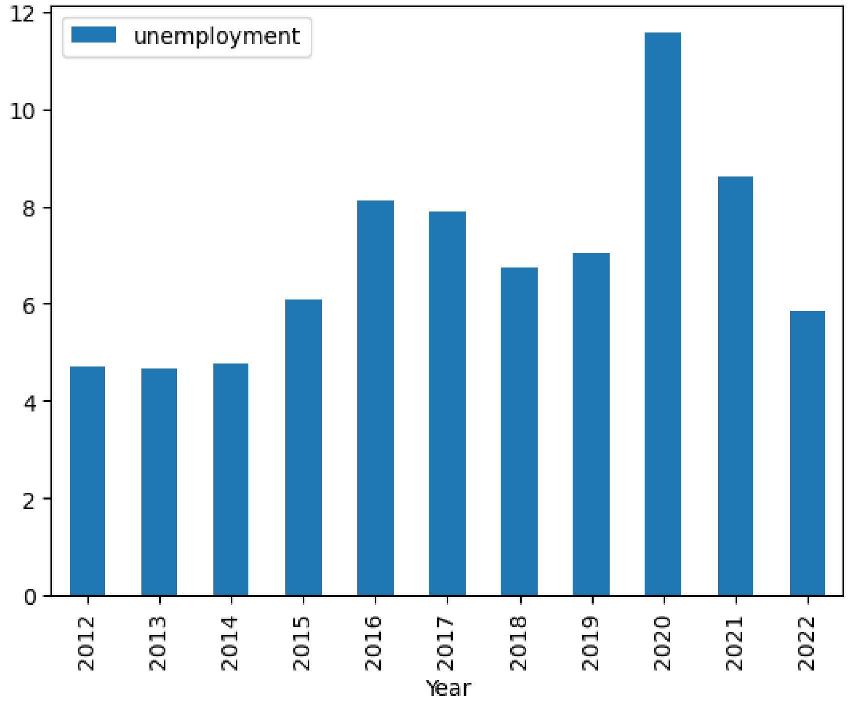
In [36]:
```
unemployment=price[["closeprice","unemployment", "Year"]].groupby("Year").mean()
#price_unemployment=pd.DataFrame(price_unemployment)
unemployment
```

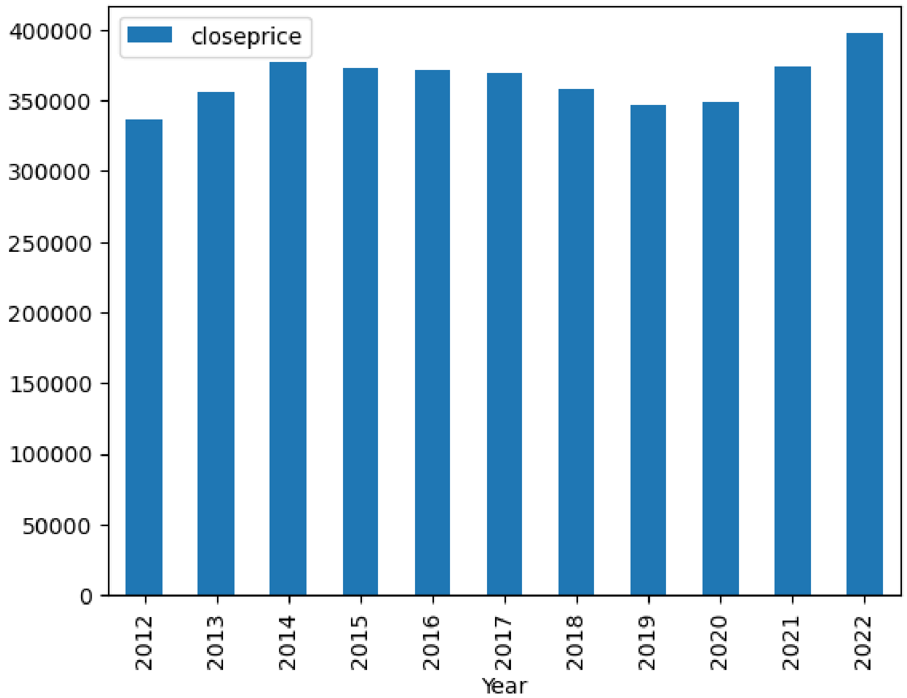
Out[36]:

Year	closeprice	unemployment
2012	336869.000000	4.700000
2013	355741.666667	4.658333
2014	377269.833333	4.775000
2015	373000.000000	6.091667
2016	371217.416667	8.133333
2017	369137.166667	7.883333
2018	358258.333333	6.725000
2019	347095.833333	7.025000
2020	348528.000000	11.558333
2021	374054.166667	8.616667
2022	397125.000000	5.850000

In [37]:
```
#fig, axes = plt.subplots(2, 1)
unemployment.plot.bar(y="unemployment")
unemployment.plot.bar(y="closeprice")
```

Out[37]: <AxesSubplot:xlabel='Year'>





The unemployment rate was stable from 2012-2014, showed a sign of increase from 2015, and due to the covid-19 outbreak it increased in 2020 and exceeded the figures of previous years, moderating in 2021 and 2022 but still not returning to the pre-pandemic levels.
The average house price has a little increase from 2012 to 2014, and kept decreasing slowly to 2019, after 2020, it continued rasing until now.
Combined these two plots, there is no significant relationship between the unemployment and the average housing prices over the ten years, since they are slightly positive correlated before 2018 and correlated negatively since 2019.

Query 4- calculate the crime rate in each community

In [38]:
```
crime_counts=new_crime.dropna(subset=["Community Name", "Crime Count", "Resident Count", "Year"])
crime_count=crime_counts[["Community Name", "Crime Count", "Resident Count", "Year"]]
crime_count.head(5)
```

Out[38]:

	Community Name	Crime Count	Resident Count	Year
0	MAPLE RIDGE	1	1916.0	2022
1	01K	1	0.0	2022
2	BANKVIEW	1	5256.0	2022
3	FRANKLIN	2	0.0	2022
4	MARTINDALE	1	14245.0	2022

In [39]:

```
# write the dataframe into a table
crime_count.to_sql("crime_count", engine, if_exists='replace')
```

Out[39]: 9660

In [40]:

```
crime_rate = pd.read_sql_query('''
                              select Year, `Community Name` as Community, (`Crime Count`/`Resident Count`)*
                              from crime_count
                              group by Community
                              order by `Crime Rate` DESC;
                              ''', engine)
print (crime_rate)
```

```
     Year                 Community  Crime Rate
0    2022                  SUNRIDGE    9.090909
1    2022  SADDLE RIDGE INDUSTRIAL    8.000000
2    2022                 YORKVILLE    7.142857
3    2022                PINE CREEK    7.142857
4    2022   FOREST LAWN INDUSTRIAL    5.691057
..    ...                       ...         ...
280  2022                       01H         NaN
281  2022                       12C         NaN
282  2022                       09P         NaN
283  2022                       12K         NaN
284  2022                       02C         NaN

[285 rows x 3 columns]
```

In [41]:

```
#Show first 10 crime rate communities
crime_rate=pd.DataFrame(crime_rate, columns=["Year", "Community", "Crime Rate"])
crime_rate.head(10)
```

Out[41]:

	Year	Community	Crime Rate
0	2022	SUNRIDGE	9.090909
1	2022	SADDLE RIDGE INDUSTRIAL	8.000000
2	2022	YORKVILLE	7.142857
3	2022	PINE CREEK	7.142857
4	2022	FOREST LAWN INDUSTRIAL	5.691057
5	2022	BELVEDERE	2.439024
6	2022	SHEPARD INDUSTRIAL	2.352941
7	2022	BELMONT	1.162791
8	2022	GREENVIEW INDUSTRIAL PARK	0.444444
9	2022	EAGLE RIDGE	0.331126

The table above shows the top 10 communities with the highest crime rates in 2022, obviously, the comunity Sunridge has the highest crime rate in Calgary.

Query 5- explore the relationship between unemployment rate and crime rate from 2017 to 2022

In [42]:

```
# write the dataframe into a table
unemployment.to_sql("unemployment", engine, if_exists='replace')
crime_rate.to_sql("crime_rate", engine, if_exists='replace')
```

Out[42]: 285

In [43]:

```
price_cr = pd.read_sql_query('''
                            select cr.Year, Community, un.unemployment, cr.`Crime Rate`
                            from crime_rate as cr
```

```
                              join unemployment as un
                              on cr.Year=cr.Year
                              group by Community
                              ;''', engine)
print (price_cr)
```

```
     Year                 Community  unemployment  Crime Rate
0    2022                       01B           4.7         NaN
1    2022                       01F           4.7         NaN
2    2022                       01H           4.7         NaN
3    2022                       01K           4.7         NaN
4    2022                       02C           4.7         NaN
..    ...                       ...           ...         ...
280  2022  WINSTON HEIGHTS/MOUNTVIEW           4.7    0.055021
281  2022               WOLF WILLOW           4.7         NaN
282  2022                  WOODBINE           4.7    0.056395
283  2022                 WOODLANDS           4.7    0.033322
284  2022                 YORKVILLE           4.7    7.142857

[285 rows x 4 columns]
```

In [44]:
```
price_cr=pd.DataFrame(price_cr, columns=["Year", "Community", "closeprice", "Crime Rate"])
price_cr.head(5)
```

Out[44]:

	Year	Community	closeprice	Crime Rate
0	2022	01B	NaN	NaN
1	2022	01F	NaN	NaN
2	2022	01H	NaN	NaN
3	2022	01K	NaN	NaN
4	2022	02C	NaN	NaN

Based on the output, there is no available data shown in this query, it may caused by the limitation of the dataset "transcation_price", it only contains a few information for each year, also, the crime data is not complete between 2017 and 2021, for example, the values in the column "Community Count" all are missed, thus we can not get enough information based on that condition.

Discussion

In general, the small amount of data from the first house price prevents us from exploring too much and joining with the another dataset, but we can see that covid-19 caused some impact on Calgary's job market leading to the highest unemployment rate in a decade in 2020, and even after some easing later on, it did not return to the situation before the pandemic. The heat map and the processed data also show that there is no direct and significant relationship between crime rate and crime count, probably because the population of each community is different. Through the above explorations, we can see that some communities are not suitable for home buying and living compared to others, but deeper explorations need more data to support this, which is the limitation of this section.

Housing Prices and Starbuck Locations

The guiding questions this data set hopes to ultimately answer is does the Calgary market exhibit the “Starbucks Effect”? By looking at geographic data for transactions and locations of Starbucks cafes, we will look to see if the “Starbucks Effect” is can still be exhibited in the Calgary market after Covid-19, and the subsequent closing of multiple Starbucks locations in Calgary. The dataset used for this contains 89 locations within the Calgary boundary, and is separated into corporate owned locations and licenced stores.

In [45]:
```
Star_df = pd.read_csv("calgary_2022_updated.csv")
Star_df['zip_code'] = Star_df['postalCode'].str[:3]
Star_df=Star_df.dropna()
Star_df.info()
Star_df.head(2)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 89 entries, 0 to 88
Data columns (total 11 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   storeNumber             89 non-null     object 
 1   ownershipType           89 non-null     object 
 2   slug                    89 non-null     object 
 3   streetAddress           89 non-null     object 
 4   city                    89 non-null     object 
 5   countrySubdivisionCode  89 non-null     object 
 6   postalCode              89 non-null     object 
 7   countryCode             89 non-null     object 
 8   latitude                89 non-null     float64
 9   longitude               89 non-null     float64
 10  zip_code                89 non-null     object 
dtypes: float64(2), object(9)
memory usage: 7.8+ KB
```

Out[45]:

	storeNumber	ownershipType	slug	streetAddress	city	countrySubdivisionCode	postalCode	countryCode	latitu
0	4853-107841	Company Owned	52-st-mc-ivor-blvd-se-15566-mc-ivor-boulevard-...	15566 McIvor Boulevard	Calgary	AB	T2Z 4Y2	CA	50.913
1	64671-298188	Licensed	sobeys-mc-kenzie-town-20-mc-kenzie-towne-ave-s...	20 McKenzie Towne Ave SE	Calgary	AB	T2Z 3S7	CA	50.917

In [49]:
```python
Star_df.to_sql('starbucks_table', engine, if_exists='replace' )
```

Out[49]:
```
89
```

In [50]:
```python
#running first query to make sure everything is added
starbucks2022_query1 = pd.read_sql_query('SELECT count(*) AS "Number of Stores that are Starbucks in Calgary
starbucks2022_query1
```

Out[50]:

	Number of Stores that are Starbucks in Calgary 2022
0	89

Query 1- Distribution of Starbucks In Grocery Stores

In [51]:
```python
safeway = pd.read_sql_query('SELECT ownershipType, count(*) AS "Total Stores that are in Safeway" FROM Starb
safeway
sobeys = pd.read_sql_query('SELECT ownershipType, count(*) AS "Total Stores that are in Sobeys in 2022" FROM
sobeys
```

Out[51]:

	ownershipType	Total Stores that are in Sobeys in 2022
0	Licensed	3

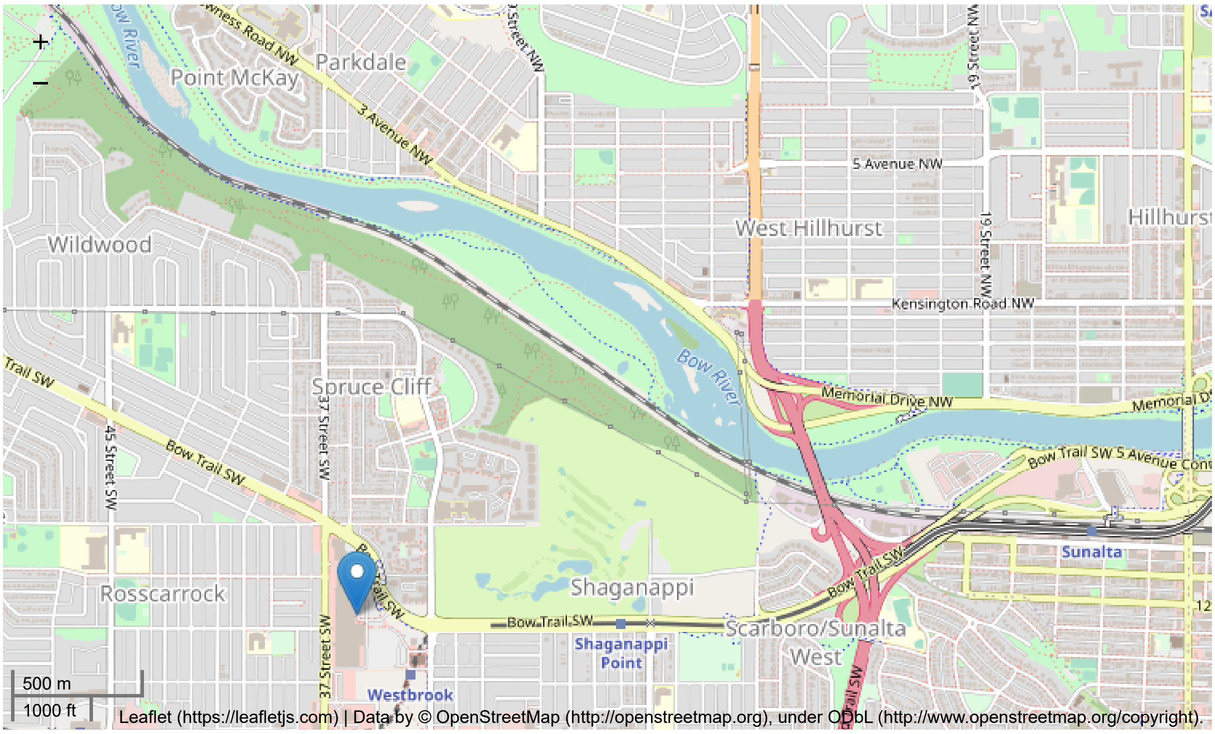
Interestingly, unlike other companies like McDonalds or Tim Hortons, Starbucks does not allow franchises, and it prefers to own each of their locations. Starbucks CEO Howard Schultz once said "I always viewed franchising as a way to get access to capital, because you're using other people's money to grow, essentially. And we were dealing with a premium product -- something that can be hard to learn, that you have to explain to the customer, that requires an educated staff. It would have been hard to provide the level of sensitivity to customers and knowledge of the product needed to create those Starbucks values if we franchised. You can be just as entrepreneurial and experimental in a company-owned model" (Kurlantzick, 2003).

However, the Seattle coffee chain does allow licences, so owned businesses or locations that can help Starbucks reach new demographics, it is possible to reach out about adding a Starbucks to that locations (McCreary, 2019). The addition of Starbucks kiosks in Safeway locations after Safeways rebranding in the 2000's is a common site in several locations. We also see stores in Sobeys grocery stores, as Sobey's purchased Safeway in 2003 (CBC, 2013), and combined similar business models. These Starbucks kiosk locations helps the company grow further into Calgary markets without the added risks of stand alone stores. It seems that this will further increase, as the coffee chain annouced a '5-year transformation strategy' during COVID-19 pandemic, and a shifting change in consumer habits, so the company can "best meet our customers where they are now" (CBC, 2021)

Query 2- Distribution of Starbucks Among Calgary Forward Sortation Areas (FSA's)

In [74]:
```
import folium
starbucks_2022_locations = Star_df[["latitude", "longitude", "slug"]]
map = folium.Map(location=[starbucks_2022_locations.latitude.mean(), starbucks_2022_locations.longitude.mear
for index, location_info in starbucks_2022_locations.iterrows():
    folium.Marker([location_info["latitude"], location_info["longitude"]], popup=location_info["slug"]).add_
map
```

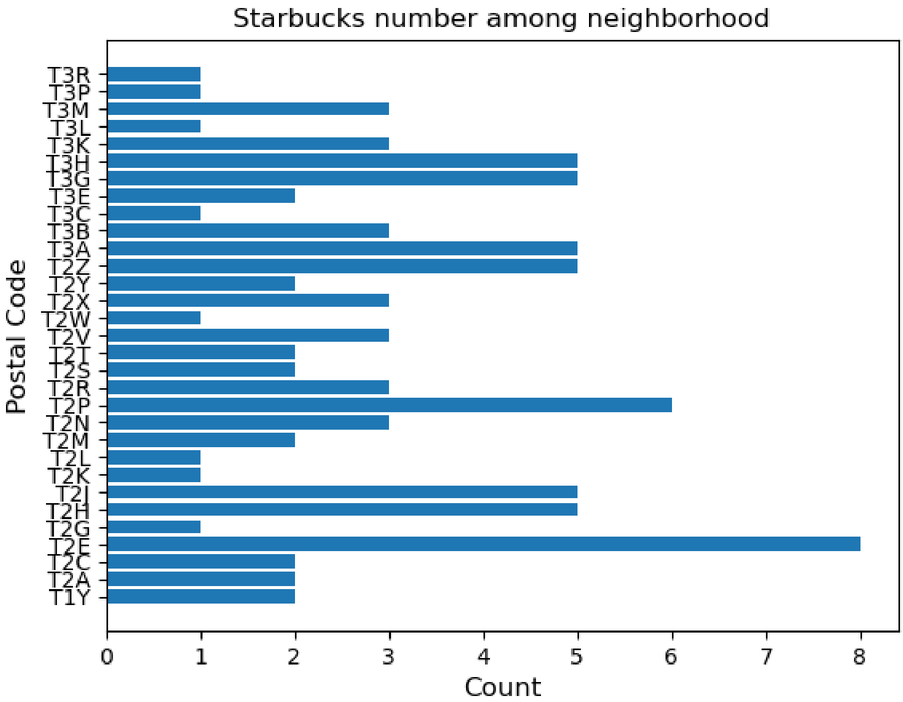
Out[74]:



In [61]:
```
query2=pd.read_sql_query('select zip_code,count(distinct storeNumber) as number\
                          from Starbucks_table\
                          group by zip_code\
                          ;', engine)
print(query2)
```

```
   zip_code  number
0       T1Y       2
1       T2A       2
2       T2C       2
3       T2E       8
4       T2G       1
5       T2H       5
6       T2J       5
7       T2K       1
8       T2L       1
9       T2M       2
10      T2N       3
11      T2P       6
12      T2R       3
13      T2S       2
14      T2T       2
15      T2V       3
16      T2W       1
17      T2X       3
18      T2Y       2
19      T2Z       5
20      T3A       5
21      T3B       3
22      T3C       1
23      T3E       2
24      T3G       5
25      T3H       5
26      T3K       3
27      T3L       1
28      T3M       3
29      T3P       1
30      T3R       1
```

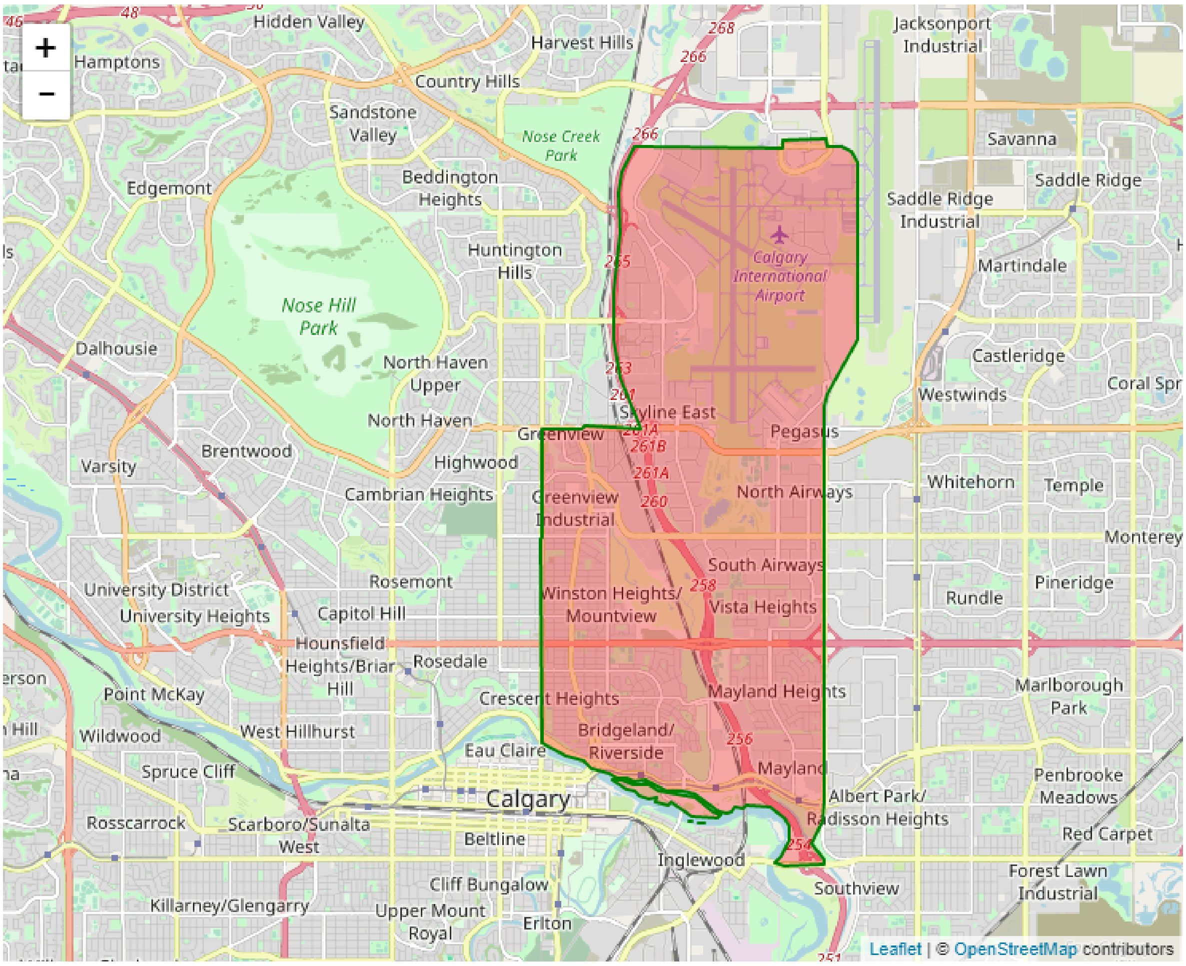
In [62]:
```
x=query2['zip_code']
y=query2['number']
plt.barh(x,y)
plt.ylabel('Postal Code', fontsize=12)
plt.xlabel('Count', fontsize=12)
plt.title('Starbucks number among neighborhood')
plt.show()
```



We looked that the the most Starbucks locations in Calgary based on the first three digits in the postal code. The first three characters of the postal codeOM ('ANA') represent a set of well-defined and stable areas known as forward sortation areas© (FSAs©). The FSA© represents a specific area within a major geographical region, a province or a territory (Government of Canada, Statistics Canada, 2014). As this makes each specific area different from eachother, we can sort in regards to the postal code.

From the bar chart, we can see that the most number of Starbucks in one area is FSA "T2E". This area is in the north-east quadrant of the city and includes the Calgary International Airport, McCall, North Airways, South Airways, Pegasus, Vista Heights, Maryland Heights, Mayland, Renfrew, Winston Heights/Mountainview, Greenview Industrial, and Bridgeland/Riverside. This area includes 8 Starbucks locations.

FSA T2E Boundary Map



In [63]:
```python
t2e_locations = pd.read_sql_query('SELECT storeNumber, slug AS "T2E Stores", streetAddress, postalCode FROM t2e_locations
```

Out[63]:

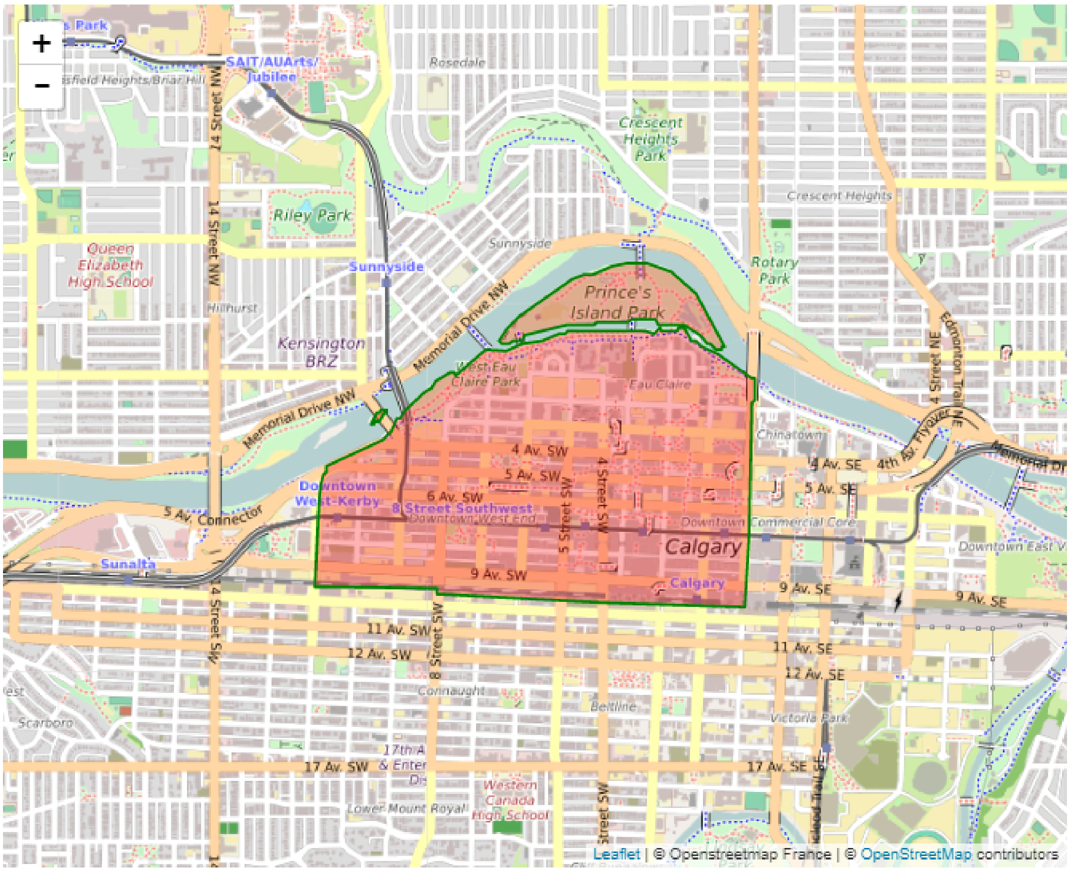
	storeNumber	T2E Stores	streetAddress	postalCode
0	4567-96456	the-bridges-951-general-avenue-ne-the-piazza-c...	951 General Avenue NE	T2E 9E1
1	4875-119885	memorial-edmonton-trail-11-edmonton-trail-ne-c...	11 Edmonton Trail NE	T2E 8R4
2	1549-108548	safeway-8823-beacon-heights-calgary-1818-centr...	1818 Centre Street NE	T2E 2S6
3	22945-207164	mc-knight-blvd-aviation-rd-252-aviation-blvd-n...	252 Aviation Blvd NE	T2E 7H8
4	3494-141218	west-jet-calgary-campus-22-aerial-pl-se-calgar...	22 Aerial Pl SE	T2E 3J1
5	75704-94633	yyc-b-gates-2000-airport-road-calgary-ab-t-2-e...	2000 Airport Road	T2E 6W5
6	70002-120504	yyc-c-gates-2000-airport-road-ne-calgary-ab-t-...	2000 Airport Road NE	T2E 6W5
7	47118-238035	yyc-us-transborder-2000-airport-rd-ne-calgary-...	2000 Airport Rd NE	T2E6W5

In [64]:
```python
moststarbucks=pd.read_sql_query('select zip_code,neighborhood\
                                 from postcode_table\
                                 where zip_code = "T2E"\
                                 ;', engine)
print(moststarbucks)
```

```
  zip_code                 neighborhood
0      T2E                    Bridgeland
1      T2E  Calgary International Airport
2      T2E                   Calgary Zoo
3      T2E                     Greenview
```

The second most Starbucks locaitons in Calgary is within the Calgary downtwon district with a total of 6 stores.

FSA T2P Boundary Map



In [65]:
```python
moststarbucks2=pd.read_sql_query('select zip_code,neighborhood\
                                 from postcode_table\
                                 where zip_code = "T2P"\
                                 ;', engine)
print(moststarbucks2)
```

```
  zip_code    neighborhood
0      T2P  Calgary Tower
1      T2P    City Centre
```

Query 3- the starbucks and house locations

In [66]:
```python
# caculate the distiance between Starbucks and houses address

output_store = pd.DataFrame(columns=["Address", "A_x", "A_y", "Store_Num", "Store_x", "Store_y"])

STAR_NAME = Star_df["storeNumber"]
STAR_X = Star_df["latitude"]
STAR_Y = Star_df["longitude"]
for i in range(num):
    address = Address.iloc[i]
    x = float(X.iloc[i])
    y = float(Y.iloc[i])
    star_min_index = findMinIndex(Star_df, x, y)
    store_name = STAR_NAME.iloc[star_min_index]
    store_x = STAR_X.iloc[star_min_index]
    store_y = STAR_Y.iloc[star_min_index]

    output_store.loc[i] = [address, X.iloc[i], Y.iloc[i], store_name, store_x, store_y]
output_store.head()
```

Out[66]:

	Address	A_x	A_y	Store_Num	Store_x	Store_y
0	4716 6 Street SW	26.24561	-98.21833	27311-246876	50.882996	-113.954328
1	18 Crestridge Mews SW	51.08195	-114.26373	4549-94631	51.124200	-114.246914
2	60 Mt Alberta Green SE	50.91449	-113.98933	4602-94979	50.932441	-113.970461
3	96 Scenic Ridge Crescent NW	51.10883	-114.21805	74907-101612	51.126009	-114.201215
4	183 Springbluff Heights SW	51.02170	-114.19400	4134-141704	51.041243	-114.208948

In [68]:
```
output_store.to_sql('store_distance', engine, if_exists='replace')
```

Out[68]:
```
4934
```

In [69]:
```
##count the associated stores
query_count= pd.read_sql_query('select count(distinct Address),Store_Num\
                              from store_distance\
                              group by Store_Num\
                              order by count(distinct Address) desc\
                              ;', engine)
print(query_count)
```

```
    count(distinct Address)     Store_Num
0                       374   4930-106303
1                       231   4864-103821
2                       210  74853-100964
3                       204    4537-95048
4                       199    4543-95050
..                      ...           ...
75                        2  74984-102932
76                        1  22945-207164
77                        1  54359-286116
78                        1  57038-290654
79                        1  75976-100261

[80 rows x 2 columns]
```

In [71]:
```
price_df = pd.read_csv("RegionPrice.csv")
price_df.info()
price_df=price_df.sort_values(by="BenchmarkPrice", ascending=False)
price_df
```

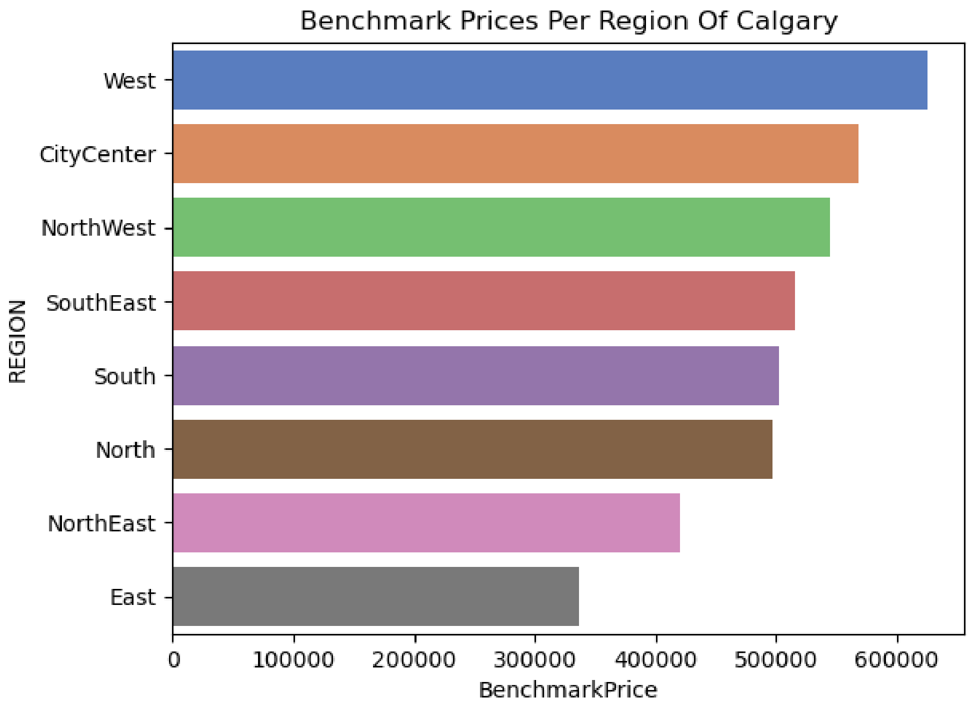
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8 entries, 0 to 7
Data columns (total 2 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   REGION          8 non-null      object
 1   BenchmarkPrice  8 non-null      float64
dtypes: float64(1), object(1)
memory usage: 256.0+ bytes
```

Out[71]:

	REGION	BenchmarkPrice
3	West	624700.0
4	CityCenter	567900.0
1	NorthWest	544400.0
7	SouthEast	515200.0
6	South	501700.0
0	North	496200.0
2	NorthEast	420400.0
5	East	337100.0

In [72]:
```
benchmark = sns.barplot(data=price_df, x="BenchmarkPrice", y="REGION",order=None, palette="muted")
benchmark.set_title("Benchmark Prices Per Region Of Calgary")
benchmark
```

Out[72]:
```
<AxesSubplot:title={'center':'Benchmark Prices Per Region Of Calgary'}, xlabel='BenchmarkPrice', ylabel='REG
ION'>
```



Looking at the THe above graph that compares the average benchmark of house prices in Calgary by quadrant, the most expensive housing prices are in the west, city center (downtown), and the north-east parts of Calgary. When comparing to the Starbucks locations in Calgary, The west side of the city has the most number of Starbucks across multiple postal codes (T2S, T2T, T2V, T2W, T2Y, T3B, T3C, T3E, T3H). The second most expensive benchmark price boundary is city center (downtown) with 11 stores (T2G, T2P, T2R, T3R), and third most expensive ares is the north-west with 12 stores (T2L, T2M, T2N, T3A, T3L). This seems to correlate that the most expensive areas in the city are generally in closer proximity with a Starbucks location.

THe least expensive areas for housing is generally around the north side, east side, and north-east side of the city. However, the number of Starbucks in the north-east side of the city is tied for 3rd overall at 12 stores (T1Y, T2A, T2E). This would suggest that there are some outliers to this theory.

Housing prices and School Locations

Data Wrangling, Cleaning and Exploration

In [13]:
```
post_df = pd.read_csv("PostCode_Community.csv")
post_df.to_sql('postcode_table', engine, if_exists='replace')
school_df=pd.read_csv("School_Locations.csv")
```

In [14]:
```
school_df.head()
```

Out[14]:

	NAME	BOARD	EMAIL	ADDRESS	CITY	PROVINCE	POSTAL_COD	PHONE_NO	FAX_NO	GI
0	SAIT - MAIN CAMPUS - CAMPUS CENTRE	NaN	NaN	616 BOYCE CR NW	Calgary	AB	NaN	(877) 284-7248	NaN	
1	Palliser Beyond Borders at Calgary	The Palliser School Division	NaN	2635 37 AV NE	Calgary	AB	T1Y5Z6	(403) 291-0584	NaN	Un
2	Marlborough School	The Calgary School Division	marlborough@cbe.ab.ca	4711 MARYVALE DR NE	Calgary	AB	T2A3A1	(403) 777-8190	NaN	Elem
3	Carousel Children's Centre	Montessori Education Preschool & ECS Institute...	admin@montcc.ca	100 CASTLEBROOK WY NE	Calgary	AB	T3J2A1	(403) 255-8664	NaN	
4	Haysboro School	The Calgary School Division	haysboro@cbe.ab.ca	1123 87 AV SW	Calgary	AB	T2V0W2	(403) 777-8530	NaN	Elem

In [15]:

```
## wrangle the geographical coordinate data
## split the lat-lon cordinate into two columns
school_df = pd.read_sql_table("school_info1", engine)
school_df["the_geom"]=school_df["the_geom"].str.replace("[POINT]","")
school_df[['latitude', 'longitude']] = school_df['the_geom'].str.extract(pat = '(-?\d+\.\d+) \s*(-?\d+\.\d+)
school_df['zip_code'] = school_df['POSTAL_COD'].str[:3]
school_df.info()
school_df.head(2)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 546 entries, 0 to 545
Data columns (total 20 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   index        546 non-null    int64
 1   NAME         546 non-null    object
 2   BOARD        538 non-null    object
 3   EMAIL        507 non-null    object
 4   ADDRESS      542 non-null    object
 5   CITY         546 non-null    object
 6   PROVINCE     546 non-null    object
 7   POSTAL_COD   535 non-null    object
 8   PHONE_NO     537 non-null    object
 9   FAX_NO       0 non-null      float64
 10  GRADES       538 non-null    object
 11  POSTSECOND   546 non-null    object
 12  DATA_SOURCE  546 non-null    object
 13  ELEM         538 non-null    object
 14  JUNIOR_H     538 non-null    object
 15  SENIOR_H     538 non-null    object
 16  the_geom     546 non-null    object
 17  latitude     546 non-null    object
 18  longitude    546 non-null    object
 19  zip_code     535 non-null    object
dtypes: float64(1), int64(1), object(18)
memory usage: 85.4+ KB
/tmp/ipykernel_75/4261219246.py:4: FutureWarning: The default value of regex will change from True to False
in a future version.
  school_df["the_geom"]=school_df["the_geom"].str.replace("[POINT]","")
```

Out[15]:

	index	NAME	BOARD	EMAIL	ADDRESS	CITY	PROVINCE	POSTAL_COD	PHONE_NO	FAX_NO	GRADES	POSTSECOND
0	0	SAIT - MAIN CAMPUS - CAMPUS CENTRE	None	None	616 BOYCE CR NW	Calgary	AB	None	(877) 284-7248	NaN	None	Y
1	1	Palliser Beyond Borders at Calgary	The Palliser School Division	None	2635 37 AV NE	Calgary	AB	T1Y5Z6	(403) 291-0584	NaN	Unknown	N

In [16]:
```
Address_df = pd.read_csv("Address_LRT_Mapping.csv")
Address_df.info()
Address_df.head(2)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4934 entries, 0 to 4933
Data columns (total 7 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Unnamed: 0  4934 non-null   int64  
 1   Address     4934 non-null   object 
 2   A-x         4934 non-null   float64
 3   A_y         4934 non-null   float64
 4   LRT_Name    4934 non-null   object 
 5   LRT_x       4934 non-null   float64
 6   LRT_y       4934 non-null   float64
dtypes: float64(4), int64(1), object(2)
memory usage: 270.0+ KB
```

Out[16]:

	Unnamed: 0	Address	A-x	A_y	LRT_Name	LRT_x	LRT_y
0	0	4716 6 Street SW	26.24561	-98.21833	Somerset-Bridlewood Station	50.899088	-114.069001
1	1	18 Crestridge Mews SW	51.08195	-114.26373	Tuscany Station	51.134470	-114.235560

In [17]:
```
service_df = pd.read_csv("Community_Services.csv")
service_df["POINT"]=service_df["POINT"].str.replace("[POINT]","")
service_df[['latitude', 'longitude']] = service_df['POINT'].str.extract(pat = '(-?\d+\.\d+) \s*(-?\d+\.\d+)'
service_df.info()
service_df.head(2)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 206 entries, 0 to 205
Data columns (total 7 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   TYPE       206 non-null    object
 1   NAME       206 non-null    object
 2   ADDRESS    204 non-null    object
 3   COMM_CODE  202 non-null    object
 4   POINT      206 non-null    object
 5   latitude   206 non-null    object
 6   longitude  206 non-null    object
dtypes: object(7)
memory usage: 11.4+ KB
/tmp/ipykernel_75/2496842458.py:2: FutureWarning: The default value of regex will change from True to False in a future version.
  service_df["POINT"]=service_df["POINT"].str.replace("[POINT]","")
```

Out[17]:

	TYPE	NAME	ADDRESS	COMM_CODE	POINT	latitude	longitude
0	Community Centre	Rosemont Community Centre	2807 10 ST NW	CAP	(-114.0860375 51.076753)	-114.0860375	51.076753
1	Attraction	WinSport's Canada Olympic Park	88 Canada Olympic RD SW	COP	(-114.2154069 51.0823997)	-114.2154069	51.0823997

Query 1- Distribution of schools among district/neighborhood

In [18]:
```
school_df.to_sql('Schoolinfo_table1', engine, if_exists='replace' )
```

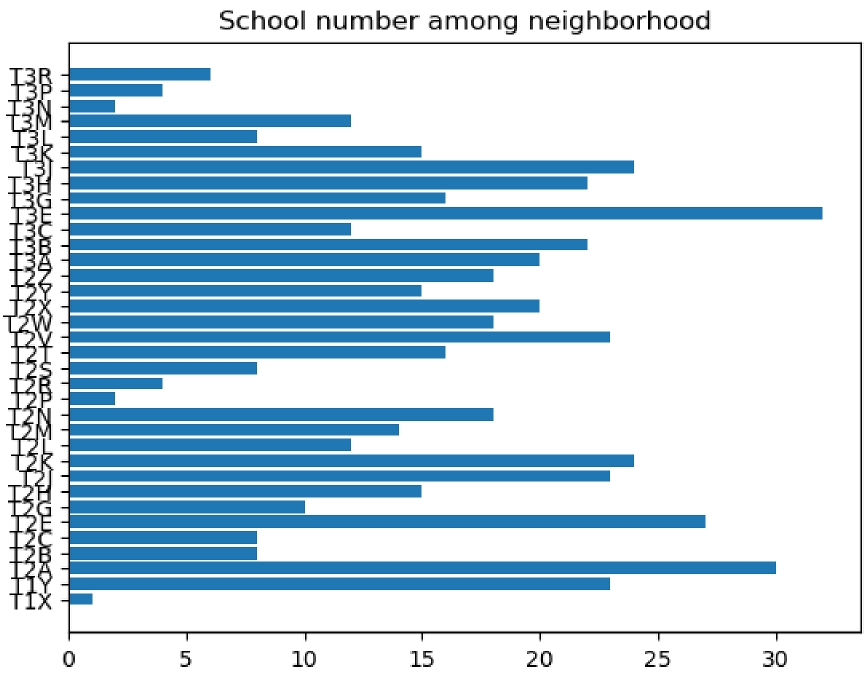
```
/opt/conda/lib/python3.10/site-packages/pandas/io/sql.py:1663: UserWarning: The provided table name 'Schooli
nfo_table1' is not found exactly as such in the database after writing the table, possibly due to case sensi
tivity issues. Consider using lower case table names.
  warnings.warn(msg, UserWarning)
```

Out[18]: 546

In [19]:
```
query1=pd.read_sql_query('select zip_code,count(distinct NAME) as number\
                         from Schoolinfo_table\
                         where zip_code <> "None"\
                         group by zip_code\
                         ;', engine)
print(query1)
```

```
   zip_code  number
0       T1X       1
1       T1Y      23
2       T2A      30
3       T2B       8
4       T2C       8
5       T2E      27
6       T2G      10
7       T2H      15
8       T2J      23
9       T2K      24
10      T2L      12
11      T2M      14
12      T2N      18
13      T2P       2
14      T2R       4
15      T2S       8
16      T2T      16
17      T2V      23
18      T2W      18
19      T2X      20
20      T2Y      15
21      T2Z      18
22      T3A      20
23      T3B      22
24      T3C      12
25      T3E      32
26      T3G      16
27      T3H      22
28      T3J      24
29      T3K      15
30      T3L       8
31      T3M      12
32      T3N       2
33      T3P       4
34      T3R       6
```

In [20]:
```
x=query1['zip_code']
y=query1['number']
plt.barh(x,y)
plt.title('School number among neighborhood')
plt.show()
```



Query 2- the neighborhood/community which has most schools

In [21]:
```
mostschools=pd.read_sql_query('select zip_code,neighborhood\
                         from postcode_table\
                         where zip_code = "T3E"\
                         ;', engine)
print(mostschools)
```

```
  zip_code neighborhood
0      T3E    Glamorgan
1      T3E     Glendale
2      T3E    Killarney
3      T3E     Lakeview
```

Query 3- the houses and surrounding schools

In [22]:
```
#Define a function to caculate the linear distance by using latitude and longitude

def findMinIndex(dataframe, ax, ay):
    num = dataframe.shape[0]
    X = dataframe["latitude"]
    Y = dataframe["longitude"]
    x = float(X.iloc[0])
    y = float(Y.iloc[0])
    dis2 = (x-ax)*(x-ax) + (y-ay)*(y-ay)
    min_index = 0
    for i in range(num):
        x = float(X.iloc[i])
        y = float(Y.iloc[i])
        temp_dis2 = (x-ax)*(x-ax) + (y-ay)*(y-ay)
        if dis2 > temp_dis2:
            dis2 = temp_dis2
            min_index = i
    return min_index

num = Address_df.shape[0]
Address = Address_df["Address"]
X = Address_df["A-x"]
Y = Address_df["A_y"]
```

In [23]:
```
# caculate the linear distiance between schools and houses address

output_schl = pd.DataFrame(columns=["Address", "A_x", "A_y", "NAME", "Schl_x", "Schl_y"])

SCHL_NAME = school_df["NAME"]
SCHL_X = school_df["latitude"]
SCHL_Y = school_df["longitude"]
for i in range(num):
    address = Address.iloc[i]
    x = float(X.iloc[i])
    y = float(Y.iloc[i])
```

```
    schl_min_index = findMinIndex(school_df, x, y)
    schl_name = SCHL_NAME.iloc[schl_min_index]
    schl_x = SCHL_X.iloc[schl_min_index]
    schl_y = SCHL_Y.iloc[schl_min_index]

    output_schl.loc[i] = [address, X.iloc[i], Y.iloc[i], schl_name, schl_x, schl_y]

output_schl.head()
```

Out[23]:

	Address	A_x	A_y	NAME	Schl_x	Schl_y
0	4716 6 Street SW	26.24561	-98.21833	Joane Cardinal-Schubert High School	-113.9440104	50.8780222
1	18 Crestridge Mews SW	51.08195	-114.26373	Joane Cardinal-Schubert High School	-113.9440104	50.8780222
2	60 Mt Alberta Green SE	50.91449	-113.98933	Joane Cardinal-Schubert High School	-113.9440104	50.8780222
3	96 Scenic Ridge Crescent NW	51.10883	-114.21805	Joane Cardinal-Schubert High School	-113.9440104	50.8780222
4	183 Springbluff Heights SW	51.02170	-114.19400	Joane Cardinal-Schubert High School	-113.9440104	50.8780222

In [26]:

```
output_schl.to_sql('school_distance', engine, if_exists='replace')
```

Out[26]: 4934

In [27]:

```
##count the associated shcools
query_count= pd.read_sql_query('select count(distinct Address), NAME\
                               from school_distance\
                               group by NAME\
                               order by count(distinct Address) desc\
                               ;', engine)
print(query_count)
```

```
   count(distinct Address)                               NAME
0                     4832  Joane Cardinal-Schubert High School
1                       54                 Divine Mercy School
2                       22              Tyndale Christian School
3                       10                   Prairie Sky School
```

Discussion

The query shows that the zip code with the most schools is T3E, representing communities: Glamorgan, Glendale, Killarney, and Lakeview. Most of these neighbourhoods are located in the southwest quadrant of Calgary, where the house price is relatively higher than in other districts in Calgary.
The connection between school location and house location turns out that most of the schools were dropped off the data frame, which means that the walking distance is not that related to the demand for houses. People may be more interested in the ranking of schools, rather than the distance between houses and schools. The linear distance could be more significant to the rental market, for instance, college students are always looking for convenient locations around schools.
So house buyers may more care about the schools in the residential community, not the linear distance.

In [ ]:

```
# CLEANUP
engine.dispose()
```

Conclusion

In [ ]:

```

```

References

• Original Price vs. Sold Price (September 29, 2022) CREB Realtor, Calgary.
• data.calgary.ca. (n.d.). Transit LRT Stations | Open Calgary. [online] Available at: https://data.calgary.ca/Transportation-Transit/Transit-LRT-Stations/2axz-xm4q [Accessed 2 Nov. 2022].
• data.calgary.ca. (n.d.). School Locations | Open Calgary. [online] Available at: https://data.calgary.ca/Services-and-Amenities/School-Locations/fd9t-tdn2 [Accessed 2 Nov. 2022].
• data.calgary.ca. (n.d.). Community Crime Statistics | Open Calgary. [online] Available at: https://data.calgary.ca/Health-and-Safety/Community-Crime-Statistics/78gh-n26t.
• www.kaggle.com. (n.d.). Starbucks Locations Worldwide. [online] Available at: https://www.kaggle.com/datasets/starbucks/store-locations [Accessed 2 Nov. 2022].

• Right to Housing Is Now Law in Canada: So Now What? – Canadian Housing & Renewal Association. https://chra-achru.ca/blog_article/right-to-housing-is-now-law-in-canada-so-now-what-2/. [Accessed 2 Nov. 2022].
• Carrick, Rob. “Our Other Real Estate Problem – People Have Too Much Wealth Tied up in Houses.” The Globe and Mail, 12 Sept. 2022. www.theglobeandmail.com, https://www.theglobeandmail.com/investing/personal-finance/household-finances/article-our-other-real-estate-problem-people-have-too-much-wealth-tied-up-in/.
• Blair, Peggy. “How Does LRT Affect Property Values?” Kiss and Sell, 8 Aug. 2019, https://peggyblairrealtor.wordpress.com/2019/08/08/how-does-lrt-affect-property-values/.
• Affordable housing decreases crime, increases property values. Affordable housing decreases crime, increases property values | School of Social Ecology. (n.d.). [online] Available at: https://socialecology.uci.edu/news/affordable-housing-decreases-crime-increases-property-values [Accessed 2 Nov. 2022].
• All Postal Codes Of Calgary - Alberta Postal Code, Canada. [online] Available at: https://www.zipcodesonline.com/2020/07/postal-code-of-calgary-in-2020.html [Accessed 2 Nov. 2022].
• List of neighbourhoods in Calgary [online] Available at: https://en.wikipedia.org/wiki/List_of_neighbourhoods_in_Calgary [Accessed 2 Dec. 2022].
• McCreary, M. (2019). You Can’t Buy a Starbucks Franchise: Here’s Why and What You Can Do Instead. [online] Entrepreneur. Available at: https://www.entrepreneur.com/franchise/you-cant-buy-a-starbucks-franchise-heres-why-and-what/311377.
• Kurlantzick, J. (2003). Entrepreneurial, Franchising - Serving Up Success. [online] Entrepreneur. Available at: https://www.entrepreneur.com/growing-a-business/entrepreneurial-franchising-serving-up-success/65008 [Accessed 11 Dec. 2022].
• CBC. (2013). Sobeys to acquire Canada Safeway stores for $5.8B. [online] Available at: https://www.cbc.ca/news/business/sobeys-to-acquire-canada-safeway-stores-for-5-8b-1.1302212 [Accessed 11 Dec. 2022].
• Starbucks to close up to 300 locations in Canada by the end of March | CBC News. (n.d.). CBC. [online] Available at: https://www.cbc.ca/news/starbucks-closures-1.5871231 [Accessed 11 Dec. 2022].
• Government of Canada, Statistics Canada (2014). 4. Technical specifications. [online] Statcan.gc.ca. Available at: https://www150.statcan.gc.ca/n1/pub/92-154-g/2015001/tech-eng.htm [Accessed 11 Dec. 2022].

In [ ]: