In my project I will be looking at a new potential location for a restaurant in Toronto. I will be analysing the area to identify the most popular restaurants and identify a gap in the market. If the area is saturated with too much of a cuisine then it will be tough for a new restaurant to survive, therefore we will be deciding what cuisine after analysing the data.

This would be helpful to anyone who wants to open a restaurants in future as this document will analyse the area and the restaurant market providing information on competitions.

The information for Toronto was obtained through the Wikipedia, which we scraped through and built a data frame. I will clean it and use Foursquare location data to obtain the latitude and longitude for each point.

To build the data frame I took the following steps of installing BeautifulSoup4 and importing necessary program

```
: pip install requests beautifulSoup4
```

```
Requirement already satisfied: requests in /home/jupyterlab/conda/envs/python/lib/python3.6/sit
e-packages (2.25.0)
Collecting beautifulSoup4
  Downloading https://files.pythonhosted.org/packages/d1/41/e6495bd7d3781cee623ce23ea6ac73282a3
73088fcd0ddc809a047b18eae/beautifulsoup4-4.9.3-py3-none-any.whl (115kB)
     |████████████████████████████████| 122kB 6.3MB/s eta 0:00:01
Requirement already satisfied: chardet<4,>=3.0.2 in /home/jupyterlab/conda/envs/python/lib/pyth
on3.6/site-packages (from requests) (3.0.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/jupyterlab/conda/envs/python/lib/
python3.6/site-packages (from requests) (1.25.11)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/python/lib/pyt
hon3.6/site-packages (from requests) (2020.12.5)
Requirement already satisfied: idna<3,>=2.5 in /home/jupyterlab/conda/envs/python/lib/python3.
6/site-packages (from requests) (2.10)
Collecting soupsieve>1.2; python_version >= "3.0" (from beautifulSoup4)
  Downloading https://files.pythonhosted.org/packages/02/fb/1c65691a9aeb7bd6ac2aa505b84cb8b49ac
29c976411c6ab3659425e045f/soupsieve-2.1-py3-none-any.whl
Installing collected packages: soupsieve, beautifulSoup4
Successfully installed beautifulSoup4-4.9.3 soupsieve-2.1
Note: you may need to restart the kernel to use updated packages.
```

```
: from bs4 import BeautifulSoup
  import requests
  import parser
  import pandas as pd
```

We will use the following URL to download the data

```
List_url = "https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M"
source = requests.get(List_url).text
```

```
pip install lxml
```

```
Collecting lxml
  Downloading https://files.pythonhosted.org/packages/bd/78/56a7c88a57d0d14945472535d0df9fb4bba
d7d34ede658ec7961635c790e/lxml-4.6.2-cp36-cp36m-manylinux1_x86_64.whl (5.5MB)
     |████████████████████████████████| 5.5MB 6.0MB/s eta 0:00:01     |███████▉
| 1.3MB 6.0MB/s eta 0:00:01
Installing collected packages: lxml
Successfully installed lxml-4.6.2
Note: you may need to restart the kernel to use updated packages.
```

```
Toronto=BeautifulSoup(source)
```

```
table=Toronto.find('table')
```

I then structed it and cleaned it for analyse

```
columns_names=['PostalCode','Borough','Neighborhood']
df = pd.DataFrame(columns = columns_names)
```

```
for tr_cell in table.find_all('tr'):
    row_data=[]
    for td_cell in tr_cell.find_all('td'):
        row_data.append(td_cell.text.strip())
    if len(row_data)==3:
        df.loc[len(df)] = row_data
```

```
df.head()
```

	PostalCode	Borough	Neighborhood
0	M1A	Not assigned	Not assigned
1	M2A	Not assigned	Not assigned
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront

```
df=df[df['Borough'] !='Not assigned']
```

```
df['Neighborhood'].loc[df['Neighborhood']=='Not assigned']
df.head()
```

	PostalCode	Borough	Neighborhood
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront
5	M6A	North York	Lawrence Manor, Lawrence Heights
6	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government

```
df.shape
```

```
(103, 3)
```

My next step was to download the latitude and longitude to each postal code area and add them to the table.

```
Geo_data=pd.read_csv("Geospatial_Coordinates.csv")
```

```
Geo_data.head()
```

	Postal Code	Latitude	Longitude
0	M1B	43.806686	-79.194353
1	M1C	43.784535	-79.160497
2	M1E	43.763573	-79.188711
3	M1G	43.770992	-79.216917
4	M1H	43.773136	-79.239476

```
Geo_data.rename(columns={'Postal Code':'PostalCode'}, inplace=True)
```

```
df.set_index("PostalCode")
```

	Borough	Neighborhood
PostalCode		
M3A	North York	Parkwoods
M4A	North York	Victoria Village
M5A	Downtown Toronto	Regent Park, Harbourfront
M6A	North York	Lawrence Manor, Lawrence Heights
M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government
...	...	...
M8X	Etobicoke	The Kingsway, Montgomery Road, Old Mill North
M4Y	Downtown Toronto	Church and Wellesley
M7Y	East Toronto	Business reply mail Processing Centre, South C...
M8Y	Etobicoke	Old Mill South, King's Mill Park, Sunnylea, Hu...
M8Z	Etobicoke	Mimico NW, The Queensway West, South of Bloor,...

103 rows × 2 columns

```
Geo_data.set_index("PostalCode")
```

	Latitude	Longitude
PostalCode		
M1B	43.806686	-79.194353
M1C	43.784535	-79.160497
M1E	43.763573	-79.188711
M1G	43.770992	-79.216917
M1H	43.773136	-79.239476
...	...	...
M9N	43.706876	-79.518188
M9P	43.696319	-79.532242
M9R	43.688905	-79.554724
M9V	43.739416	-79.588437
M9W	43.706748	-79.594054

103 rows × 2 columns

```
df=pd.merge(df,Geo_data)
```

```
df.head()
```

	PostalCode	Borough	Neighborhood	Latitude	Longitude
0	M3A	North York	Parkwoods	43.753259	-79.329656
1	M4A	North York	Victoria Village	43.725882	-79.315572
2	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
3	M6A	North York	Lawrence Manor, Lawrence Heights	43.718518	-79.464763
4	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494

I then added my foursquare detail

```
CLIENT_ID = 'FMFUJSQWIPKHL4XJUKDILFJH5GPUABK1U2LUOCM4JZO5LWH2'
CLIENT_SECRET = 'JWK25OQ2PGHRTMNKV1ML2OBESTQJ55Y11FTAG4AMSLI3MLVQ'
VERSION = '20210201'
```

My next step was to find venues and point of interest around Toronto

```
def getNearbyVenues(names, latitudes, longitudes):
    radius=500
    LIMIT=100
    venues=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)
```

create the API request URL

```
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    lat,
    lng,
    radius,
    LIMIT)
```

make the GET request

```
results = requests.get(url).json()["response"]['groups'][0]['items']
```

return only relevant information for each nearby venue

```
        venues.append([(
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']) for v in results])

    nearby_venues = pd.DataFrame([item for venue in venues for item in venue])
    nearby_venues.columns = ['Neighborhood',
                  'Neighborhood Latitude',
                  'Neighborhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

```
Venues = getNearbyVenues(names=df['Neighborhood'],
                                    latitudes=df['Latitude'],
                                    longitudes=df['Longitude'])
```

I then created a list of unique venue categories

```
Venues.head()
```

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Parkwoods	43.753259	-79.329656	Brookbanks Park	43.751976	-79.332140	Park
1	Parkwoods	43.753259	-79.329656	Variety Store	43.751974	-79.333114	Food & Drink Shop
2	Parkwoods	43.753259	-79.329656	Corrosion Service Company Limited	43.752432	-79.334661	Construction & Landscaping
3	Victoria Village	43.725882	-79.315572	Victoria Village Arena	43.723481	-79.315635	Hockey Arena
4	Victoria Village	43.725882	-79.315572	Portugril	43.725819	-79.312785	Portuguese Restaurant

This provided 271 unique categories

```
print('There are {} uniques categories.'.format(len(Venues['Venue Category'].unique())))
```

```
There are 271 uniques categories.
```

```
print(list(dict.fromkeys(Venues['Venue Category'])))
```

```
['Park', 'Food & Drink Shop', 'Construction & Landscaping', 'Hockey Arena', 'Portuguese Restaurant', 'Coffee Shop', 'French Restaurant',
'Bakery', 'Distribution Center', 'Spa', 'Restaurant', 'Breakfast Spot', 'Gym / Fitness Center', 'Historic Site', 'Chocolate Shop', 'Farme
rs Market', 'Dessert Shop', 'Pub', 'Performing Arts Venue', 'Yoga Studio', 'Café', 'Theater', 'Event Space', 'Shoe Store', 'Brewery', 'Ar
t Gallery', 'Cosmetics Shop', 'Asian Restaurant', 'Electronics Store', 'Beer Store', 'Bank', 'Hotel', 'Wine Shop', 'Antique Shop', 'Bouti
que', 'Furniture / Home Store', 'Vietnamese Restaurant', 'Clothing Store', 'Accessories Store', 'Carpet Store', 'Miscellaneous Shop', 'It
alian Restaurant', 'Beer Bar', 'Creperie', 'Sushi Restaurant', 'Mexican Restaurant', 'Hobby Shop', 'Diner', 'Burrito Place', 'Fried Chick
en Joint', 'Discount Store', 'Nightclub', 'Japanese Restaurant', 'Fast Food Restaurant', 'Smoothie Shop', 'Sandwich Place', 'Gym', 'Colle
ge Auditorium', 'Bar', 'College Cafeteria', 'Caribbean Restaurant', 'Gastropub', 'Pharmacy', 'Pizza Place', 'Intersection', 'Flea Marke
t', 'Athletics & Sports', 'Comic Shop', 'Plaza', 'Burger Joint', 'Music Venue', 'Ramen Restaurant', 'Tanning Salon', 'Tea Room', 'Steakho
use', 'Shopping Mall', 'College Rec Center', 'Sporting Goods Shop', 'Thai Restaurant', 'Modern European Restaurant', 'Bookstore', 'Lake',
'Middle Eastern Restaurant', 'Lounge', 'Ethiopian Restaurant', 'Seafood Restaurant', 'Video Game Store', 'Bubble Tea Shop', 'New American
Restaurant', 'Department Store', 'Juice Bar', 'Other Great Outdoors', 'Chinese Restaurant', 'Lingerie Store', 'Poutine Place', 'Movie The
ater', 'Office', 'Wine Bar', 'Ice Cream Shop', 'Smoke Shop', 'Hookah Bar', 'Filipino Restaurant', 'Home Service', 'Bike Shop', 'Dim Sum R
estaurant', 'Supermarket', 'Skating Rink', 'Curling Ice', 'Video Store', 'Food Truck', 'BBQ Joint', 'American Restaurant', 'Cocktail Ba
r', 'Jazz Club', 'Vegetarian / Vegan Restaurant', 'Fountain', 'Tailor Shop', 'Grocery Store', 'Cheese Shop', 'Fish Market', 'German Resta
urant', 'Comfort Food Restaurant', 'Salon / Barbershop', 'Irish Pub', 'Moroccan Restaurant', 'Bistro', 'Belgian Restaurant', 'Field', 'Tr
ail', 'Tennis Court', 'Liquor Store', 'Rental Car Location', 'Medical Center', 'Health Food Store', 'Neighborhood', 'Concert Hall', 'Muse
um', 'Basketball Stadium', 'Greek Restaurant', 'Beach', 'Gourmet Shop', 'Bagel Shop', 'Eastern European Restaurant', "Women's Store", 'Po
ol', 'Korean BBQ Restaurant', 'Indian Restaurant', 'Sports Bar', 'Fish & Chips Shop', 'Pet Store', 'Poke Place', 'Art Museum', 'Falafel R
estaurant', 'Salad Place', 'Donut Shop', 'Bike Rental / Bike Share', 'Korean Restaurant', 'Candy Store', 'Baby Store', 'Hakka Restauran
t', 'Gas Station', 'Golf Course', 'Mediterranean Restaurant', 'Dog Run', 'Deli / Bodega', 'Bridal Shop', 'Warehouse Store', 'Speakeasy',
'Opera House', 'Food Court', 'Monument / Landmark', 'Colombian Restaurant', 'Gluten-free Restaurant', 'Brazilian Restaurant', 'Noodle Hou
se', 'Latin American Restaurant', 'Gift Shop', 'Building', 'Cupcake Shop', 'Soup Place', 'Playground', 'Toy / Game Store', 'Supplement Sh
op', 'Luggage Store', 'Jewelry Store', 'Bus Station', 'Convenience Store', 'Baseball Field', 'Massage Studio', 'Metro Station', 'IT Servi
ces', 'Roof Deck', 'Dance Studio', 'Train Station', 'Aquarium', 'History Museum', 'Scenic Lookout', 'Baseball Stadium', 'Indie Movie Thea
ter', 'Hotel Bar', 'Cuban Restaurant', 'Record Shop', "Men's Store", 'Airport', 'Bus Stop', 'Fruit & Vegetable Store', 'Tibetan Restauran
t', 'Frozen Yogurt Shop', 'General Travel', 'General Entertainment', 'Taco Place', 'Garden', 'Climbing Gym', 'Stadium', 'Bus Line', 'Socc
er Field', 'Light Rail Station', 'Basketball Court', 'Motel', 'Business Service', 'Gay Bar', 'Stationery Store', 'Coworking Space', 'Butc
her', 'College Stadium', 'Arts & Crafts Store', 'Swim School', 'Cajun / Creole Restaurant', 'Auto Garage', 'Health & Beauty Service', 'Fl
ower Shop', 'Mobile Phone Shop', 'College Gym', 'College Arts Building', 'Post Office', 'Lawyer', 'Organic Grocery', 'Gaming Cafe', 'Dump
ling Restaurant', 'Doner Restaurant', 'Hospital', 'Bed & Breakfast', 'Airport Lounge', 'Harbor / Marina', 'Airport Food Court', 'Airport
Terminal', 'Airport Gate', 'Plane', 'Airport Service', 'Sculpture Garden', 'Boat or Ferry', 'Molecular Gastronomy Restaurant', 'Church',
'Optical Shop', 'Drugstore', 'Garden Center', 'Truck Stop', 'Taiwanese Restaurant', 'Market', 'Snack Place', 'River', 'Theme Restaurant',
'Martial Arts School', 'Adult Boutique', 'Sake Bar', 'Strip Club', 'Skate Park', 'Auto Workshop', 'Wings Joint', 'Hardware Store', 'Socia
l Club']
```

I then filtered for restaurants which gave me 485 rows of places.

```
Restaurant = Venues[Venues['Venue Category'].str.contains("Restaurant")]
Restaurant
```

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
4	Victoria Village	43.725882	-79.315572	Portugril	43.725819	-79.312785	Portuguese Restaurant
6	Victoria Village	43.725882	-79.315572	The Frig	43.727051	-79.317418	French Restaurant
11	Regent Park, Harbourfront	43.654260	-79.360636	Impact Kitchen	43.656369	-79.356980	Restaurant
28	Regent Park, Harbourfront	43.654260	-79.360636	Cluny Bistro & Boulangerie	43.650565	-79.357843	French Restaurant
44	Regent Park, Harbourfront	43.654260	-79.360636	Izumi	43.649970	-79.360153	Asian Restaurant
...	...	...	...	...	...	...	...
2081	Church and Wellesley	43.665860	-79.383160	Asahi Sushi	43.669874	-79.382943	Sushi Restaurant
2083	Church and Wellesley	43.665860	-79.383160	McDonald's	43.668854	-79.385962	Fast Food Restaurant
2092	Business reply mail Processing Centre, South C...	43.662744	-79.321558	Chick-n-Joy	43.665181	-79.321403	Fast Food Restaurant
2093	Business reply mail Processing Centre, South C...	43.662744	-79.321558	The Green Wood	43.664728	-79.324117	Restaurant
2111	Mimico NW, The Queensway West, South of Bloor,...	43.628841	-79.520999	McDonald's	43.630007	-79.518041	Fast Food Restaurant

Out of which only 12 had be categorised as an Indian restaurants

```
Food = Venues[Venues['Venue Category'].str.contains("Indian Restaurant")]
Food
```

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
415	Woburn	43.770992	-79.216917	Al-Hamd Biryani & Pizza	43.767585	-79.219570	Indian Restaurant
483	Central Bay Street	43.657952	-79.387383	Colaba Junction	43.660940	-79.385635	Indian Restaurant
561	Thorncliffe Park	43.705369	-79.349372	Iqbal Kebab & Sweet Centre	43.705923	-79.351521	Indian Restaurant
568	Thorncliffe Park	43.705369	-79.349372	Hakka Garden	43.704578	-79.349770	Indian Restaurant
850	Harbourfront East, Union Station, Toronto Islands	43.640816	-79.381752	Indian Roti House	43.639060	-79.385422	Indian Restaurant
962	The Danforth West, Riverdale	43.679557	-79.352188	Sher-E-Punjab	43.677308	-79.353066	Indian Restaurant
1282	Bedford Park, Lawrence Manor East	43.733283	-79.419750	The Copper Chimney	43.736195	-79.420271	Indian Restaurant
1364	Dorset Park, Wexford Heights, Scarborough Town...	43.757410	-79.273304	Kairali	43.754915	-79.276945	Indian Restaurant
1365	Dorset Park, Wexford Heights, Scarborough Town...	43.757410	-79.273304	Karaikudi Chettinad South Indian Restaurant	43.756042	-79.276276	Indian Restaurant
1448	The Annex, North Midtown, Yorkville	43.672710	-79.405678	Roti Cuisine of India	43.674618	-79.408249	Indian Restaurant
1505	Davisville	43.704324	-79.388790	Marigold Indian Bistro	43.702881	-79.388008	Indian Restaurant
1865	St. James Town, Cabbagetown	43.667967	-79.367675	Butter Chicken Factory	43.667072	-79.369184	Indian Restaurant
2036	Church and Wellesley	43.665860	-79.383160	Kothur Indian Cuisine	43.667872	-79.385659	Indian Restaurant

We will now have a look at these restaurants on a map to see the spread across Toronto

```
import folium
```

```
Locations = Food[['Venue Latitude','Venue Longitude']]
Locationlist = Locations.values.tolist()
len(Locationlist)
Locationlist[0]
```

```
[43.767584639731936, -79.21956957790067]
```

```
FoodDrink = folium.Map(location=[43.6532,-79.3832], zoom_start=12)
for i in list(range(len(Locationlist))):
    CinemaMarker = folium.Marker(Locationlist[i])
    CinemaMarker.add_to(FoodDrink)
FoodDrink
```

```
print('There are {} uniques neighborhood.'.format(len(Venues['Neighborhood'].unique())))
```

```
There are 96 uniques neighborhood.
```

Looking at the map we can see a small cluster of Indian restaurants near central Toronto

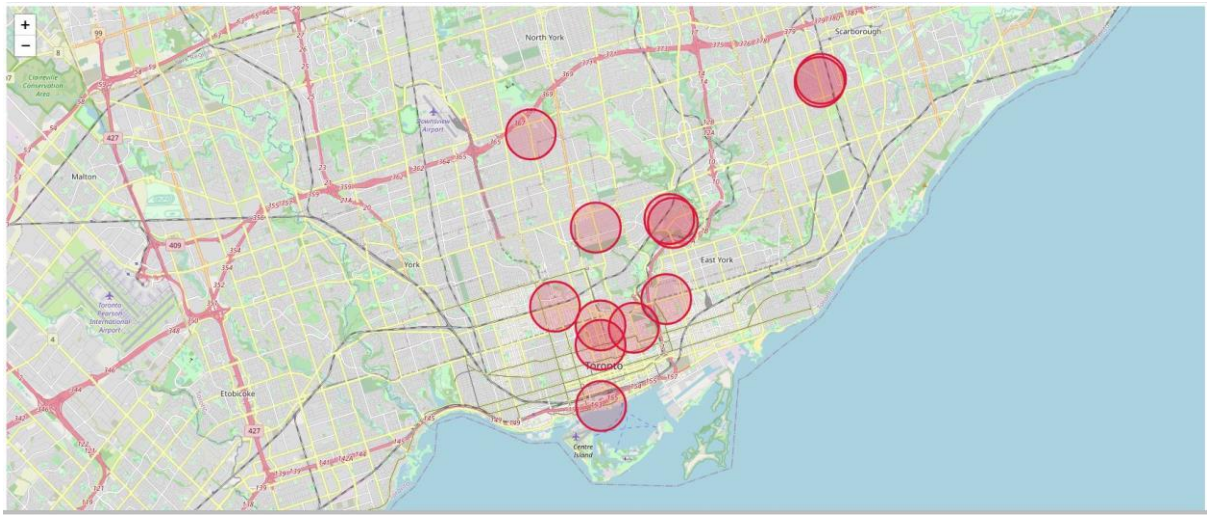
```
IndianLocations = Food[['Venue Latitude','Venue Longitude']]
IndianLocationlist = IndianLocations.values.tolist()
len(IndianLocationlist)
IndianLocationlist[0]
```

```
[43.767584639731936, -79.21956957790067]
```

We will now look at delivery area of each restaurant to see which area is untouched.

```
Indian = folium.Map(location=[43.6532,-79.3832], zoom_start=12)
for i in list(range(len(IndianLocationlist))):
    CinemaMarker = folium.Circle(Locationlist[i], radius=1000, color='crimson', fill=True, fill_color='crimson')
    CinemaMarker.add_to(Indian)

Indian
```



Looking at the map above I would suggest the best location for a new Indian restaurant is in York as there is no competition for a radius of 1000 and would be able to target untouched consumers.