Introduction:

Assumption: A friend of mine called Freddy is looking to extend his fast food business to Texas, USA. He is already operating within NewYork area and now he wishes to introduce his Fast Food Busines sin Texas Area. Freddy discussed the situation with me and I offered him my services as a Data Analysts. So this project is basically about Freddy's business and how exactly can I portray the true picture in Texas as a whole given a number of its cities especially Houston to begin with as far as the Fast Food Restaurants are concerned.

In terms of Area, Texas is second in USA and with its capital Austin, Houston is the largest city of this state/province with an estimated 2019 population of 2,320,268. There are more than 10,000 restaurants in the Houston area with culinary choices that represent more than 70 countries and American regions. However, our main concern is with the Fast food Restaurants and which specific areas have more of them. The competition in terms of population and number of restaurants will also be an important factor to consider.

Problem:

We need to find solution to following questions;

1. List and visualize the major parts of Houston, Texas where the Fast Food Restaurants are located.
2. Specifying the most convenient location for Fast Food Restaurant.
3. Which areas have potential market for a Fast Food Restaurant.
4. The Areas which lack Fast Food Restaurants.

Data Section:

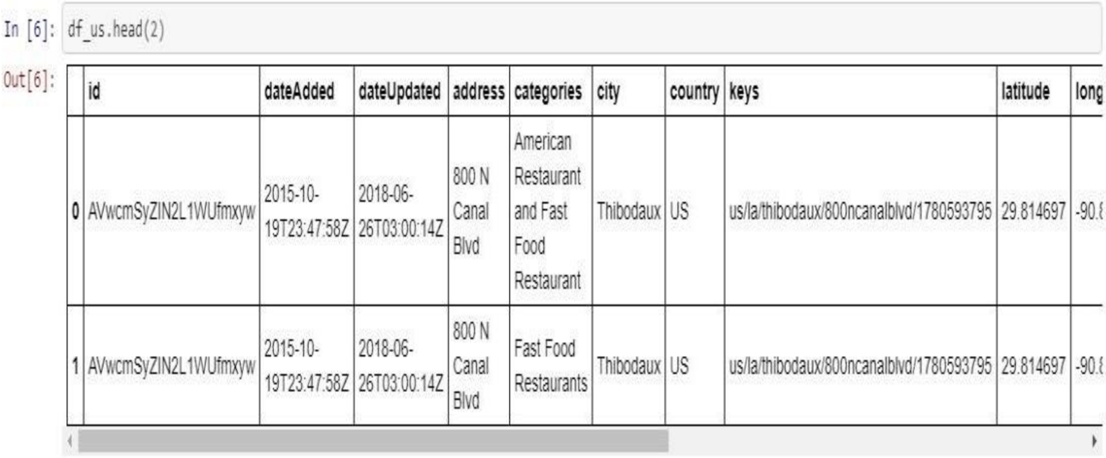
Houston is the biggest city of the state of Texas and a well-diversified in terms of ethnicity as well. There are various restaurants for the people belonging to different ethnicity like Chinese, Indian Cuisine, Mexican. However we are interested in Fast Food Restaurants.

- Houston, Texas City data will be arranged containing the details of all the restaurants in all the cities of the State of Texas. *Source*: www.datafinity.co The data will be downloaded and dealt with. *Why*: The data set will be converted toa CSV format file and then used in the project. The dataset includes the details of all types of restaurants in all the cities of Texas including Latitude & Longitude. We will in fact slice the data to get to Houston Texas and then to the Fast Food Restaurants in this city.
- Fast Food Restaurants in the Houston city. *Source:* Foursquare API. *Why:* This will help us to locate the Fast Food Restaurants in the Area of Houston by getting the relevant venues.

Methodology:

Although the data was finally available but it has been a very challenging task to find a cleaner and more relevant data. However when the data was arranged it was not just for the Houston Cit, nor the State of Texas but the complete US states and Cities which required a lot of wrangling and cleaning in terms of narrowing it down to the specific data.

The dataset was in a csv file and was read through pandas of which the initial DataFrame looked like following;

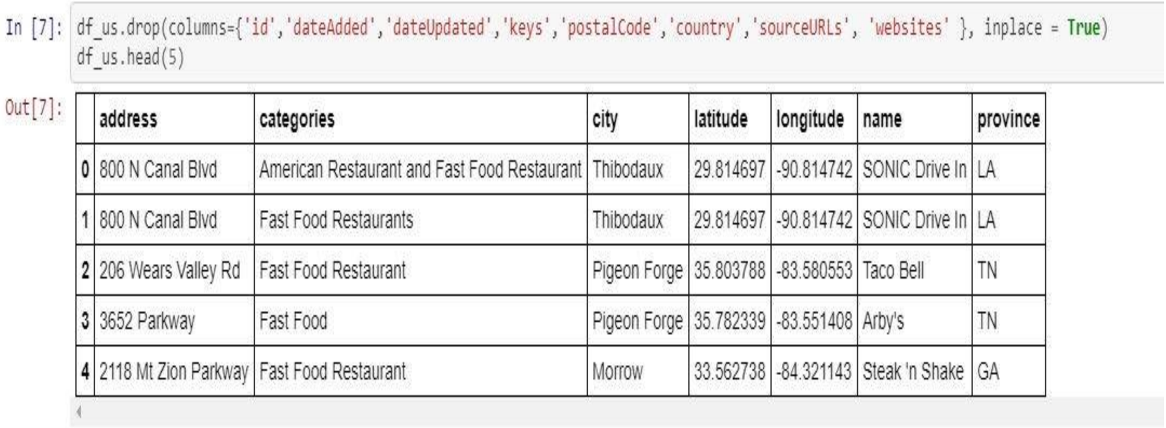


```
In [6]: df_us.head(2)
```

Out[6]:

	id	dateAdded	dateUpdated	address	categories	city	country	keys	latitude	long
0	AVwcmSyZIN2L1WUfmxyw	2015-10-19T23:47:58Z	2018-06-26T03:00:14Z	800 N Canal Blvd	American Restaurant and Fast Food Restaurant	Thibodaux	US	us/la/thibodaux/800ncanalblvd/1780593795	29.814697	-90.8
1	AVwcmSyZIN2L1WUfmxyw	2015-10-19T23:47:58Z	2018-06-26T03:00:14Z	800 N Canal Blvd	Fast Food Restaurants	Thibodaux	US	us/la/thibodaux/800ncanalblvd/1780593795	29.814697	-90.8

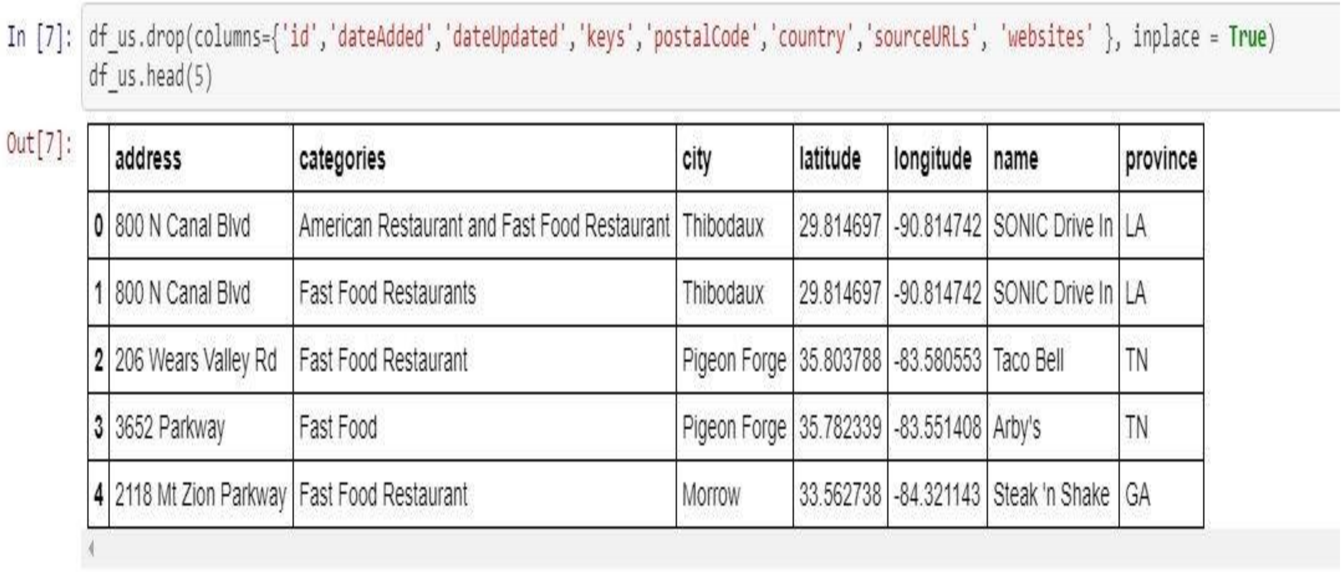
Proceeding further since the data set had a huge shape i.e. (10000, 15) it needed to be dealt with to bring it to a more usable format given the requirement of this project. So we dropped *'id','dateAdded','dateUpdated','keys','postalCode','country','sourceURLs', 'websites'* columns from the DataFrame which brought us to a more convenient looking dataset to proceed with.



```
In [7]: df_us.drop(columns={'id','dateAdded','dateUpdated','keys','postalCode','country','sourceURLs', 'websites' }, inplace = True)
        df_us.head(5)
```

Out[7]:

	address	categories	city	latitude	longitude	name	province
0	800 N Canal Blvd	American Restaurant and Fast Food Restaurant	Thibodaux	29.814697	-90.814742	SONIC Drive In	LA
1	800 N Canal Blvd	Fast Food Restaurants	Thibodaux	29.814697	-90.814742	SONIC Drive In	LA
2	206 Wears Valley Rd	Fast Food Restaurant	Pigeon Forge	35.803788	-83.580553	Taco Bell	TN
3	3652 Parkway	Fast Food	Pigeon Forge	35.782339	-83.551408	Arby's	TN
4	2118 Mt Zion Parkway	Fast Food Restaurant	Morrow	33.562738	-84.321143	Steak 'n Shake	GA

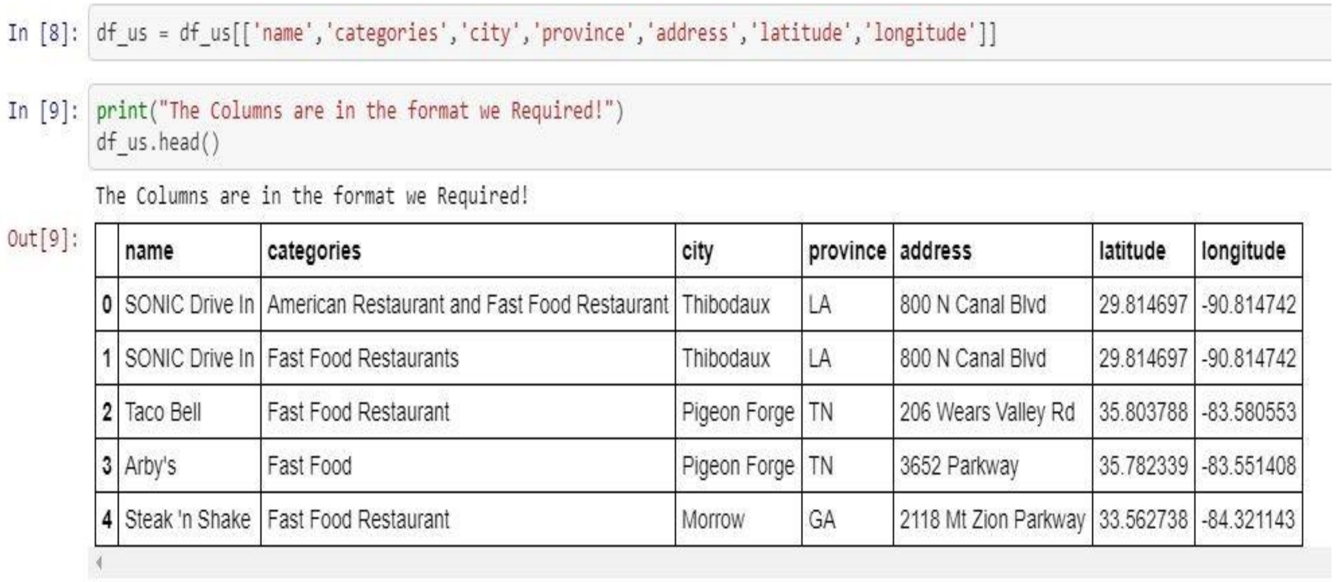


```
In [7]: df_us.drop(columns={'id','dateAdded','dateUpdated','keys','postalCode','country','sourceURLs', 'websites' }, inplace = True)
        df_us.head(5)
```

Out[7]:

	address	categories	city	latitude	longitude	name	province
0	800 N Canal Blvd	American Restaurant and Fast Food Restaurant	Thibodaux	29.814697	-90.814742	SONIC Drive In	LA
1	800 N Canal Blvd	Fast Food Restaurants	Thibodaux	29.814697	-90.814742	SONIC Drive In	LA
2	206 Wears Valley Rd	Fast Food Restaurant	Pigeon Forge	35.803788	-83.580553	Taco Bell	TN
3	3652 Parkway	Fast Food	Pigeon Forge	35.782339	-83.551408	Arby's	TN
4	2118 Mt Zion Parkway	Fast Food Restaurant	Morrow	33.562738	-84.321143	Steak 'n Shake	GA

Then it was imperative to bring the columns in a specific order for a convenient understanding while looking at the Dataframe. So we got it this way;

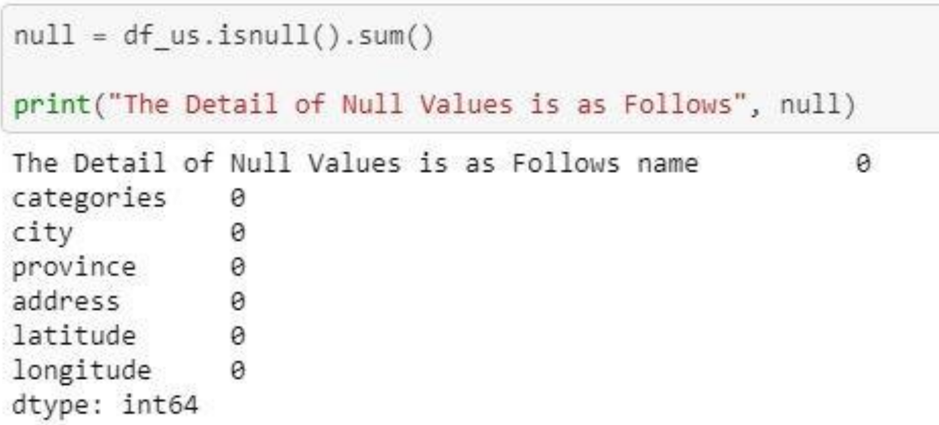


```
In [8]: df_us = df_us[['name','categories','city','province','address','latitude','longitude']]

In [9]: print("The Columns are in the format we Required!")
        df_us.head()

The Columns are in the format we Required!
```

Out[9]:

	name	categories	city	province	address	latitude	longitude
0	SONIC Drive In	American Restaurant and Fast Food Restaurant	Thibodaux	LA	800 N Canal Blvd	29.814697	-90.814742
1	SONIC Drive In	Fast Food Restaurants	Thibodaux	LA	800 N Canal Blvd	29.814697	-90.814742
2	Taco Bell	Fast Food Restaurant	Pigeon Forge	TN	206 Wears Valley Rd	35.803788	-83.580553
3	Arby's	Fast Food	Pigeon Forge	TN	3652 Parkway	35.782339	-83.551408
4	Steak 'n Shake	Fast Food Restaurant	Morrow	GA	2118 Mt Zion Parkway	33.562738	-84.321143

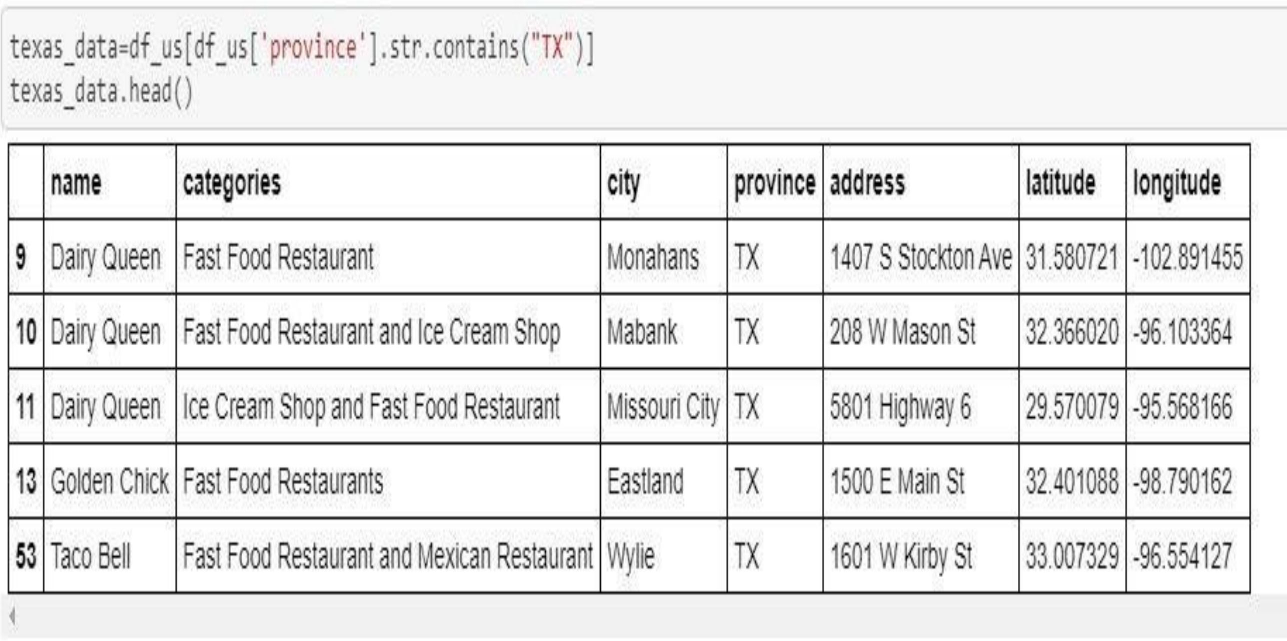
Now it was necessary to check the Dataframe for any missing, Null values and the address them.



```
null = df_us.isnull().sum()

print("The Detail of Null Values is as Follows", null)
```

```
The Detail of Null Values is as Follows name          0
categories    0
city          0
province      0
address       0
latitude      0
longitude     0
dtype: int64
```

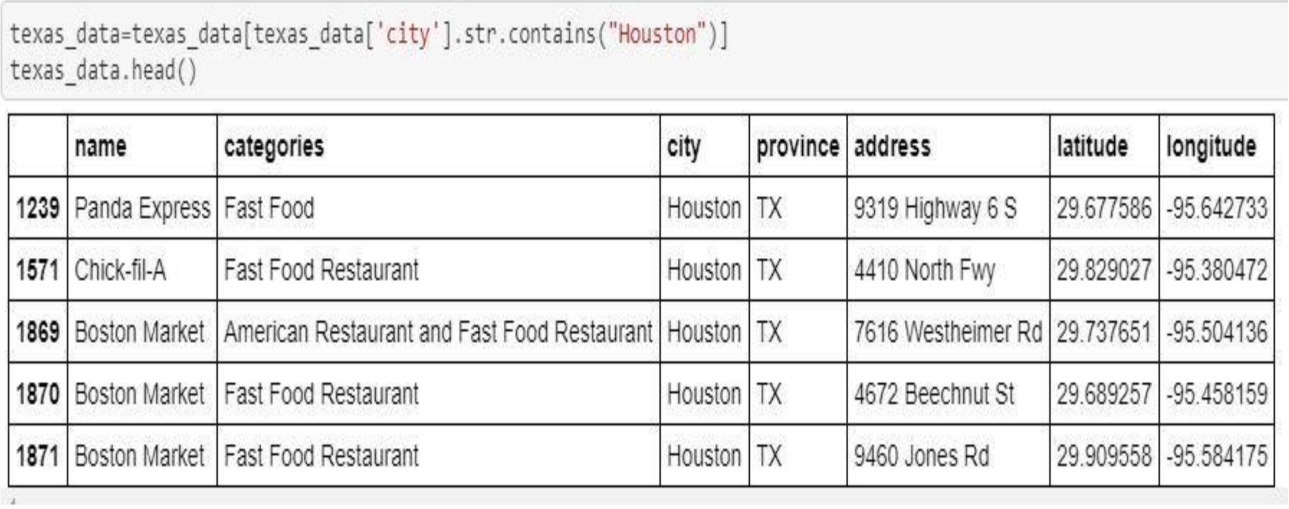
This helped in a growing confidence in the dataset as it appears to be lot more clean and workable now. However, with such a huge DataFrame it was now necessary to slice it in two folds, first to narrow it down to the state of Texas only, which will not only reduce the number of rows and will get us closer to our most specific data.



```
texas_data=df_us[df_us['province'].str.contains("TX")]
texas_data.head()
```

	name	categories	city	province	address	latitude	longitude
9	Dairy Queen	Fast Food Restaurant	Monahans	TX	1407 S Stockton Ave	31.580721	-102.891455
10	Dairy Queen	Fast Food Restaurant and Ice Cream Shop	Mabank	TX	208 W Mason St	32.366020	-96.103364
11	Dairy Queen	Ice Cream Shop and Fast Food Restaurant	Missouri City	TX	5801 Highway 6	29.570079	-95.568166
13	Golden Chick	Fast Food Restaurants	Eastland	TX	1500 E Main St	32.401088	-98.790162
53	Taco Bell	Fast Food Restaurant and Mexican Restaurant	Wylie	TX	1601 W Kirby St	33.007329	-96.554127

So now we have a DataFrame that is specific with the state/province of Texas only which also includes the names of the venues their category as well as the specific city along with the address and both Latitude & Longitude. But still one more step is to be taken and that is to slice it down to the Restaurants in Houston City only and this is how it was sliced further.



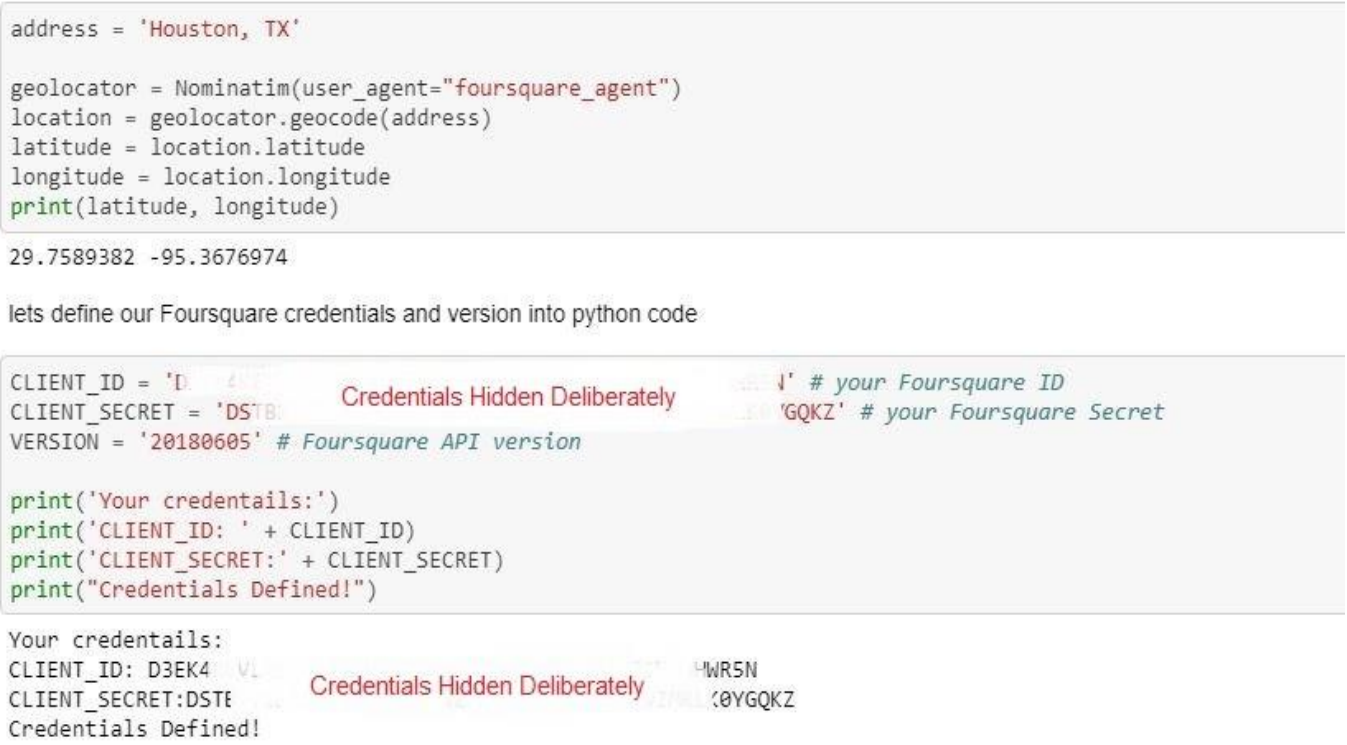
```
texas_data=texas_data[texas_data['city'].str.contains("Houston")]
texas_data.head()
```

	name	categories	city	province	address	latitude	longitude
1239	Panda Express	Fast Food	Houston	TX	9319 Highway 6 S	29.677586	-95.642733
1571	Chick-fil-A	Fast Food Restaurant	Houston	TX	4410 North Fwy	29.829027	-95.380472
1869	Boston Market	American Restaurant and Fast Food Restaurant	Houston	TX	7616 Westheimer Rd	29.737651	-95.504136
1870	Boston Market	Fast Food Restaurant	Houston	TX	4672 Beechnut St	29.689257	-95.458159
1871	Boston Market	Fast Food Restaurant	Houston	TX	9460 Jones Rd	29.909558	-95.584175

Description:

Now we engaged the FourSquare-API for which we had to bring in the "Nominatom" and select "Houston, TX" as our location to proceed further. We also called in the "Foursquare Credentials" function to establish the connection in this environment with the Forusquare online platform.

As shown in the figure



```
address = 'Houston, TX'

geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

```
29.7589382 -95.3676974
```

lets define our Foursquare credentials and version into python code

```
CLIENT_ID = '[redacted]' # your Foursquare ID
CLIENT_SECRET = '[redacted]' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version

print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
print("Credentials Defined!")
```

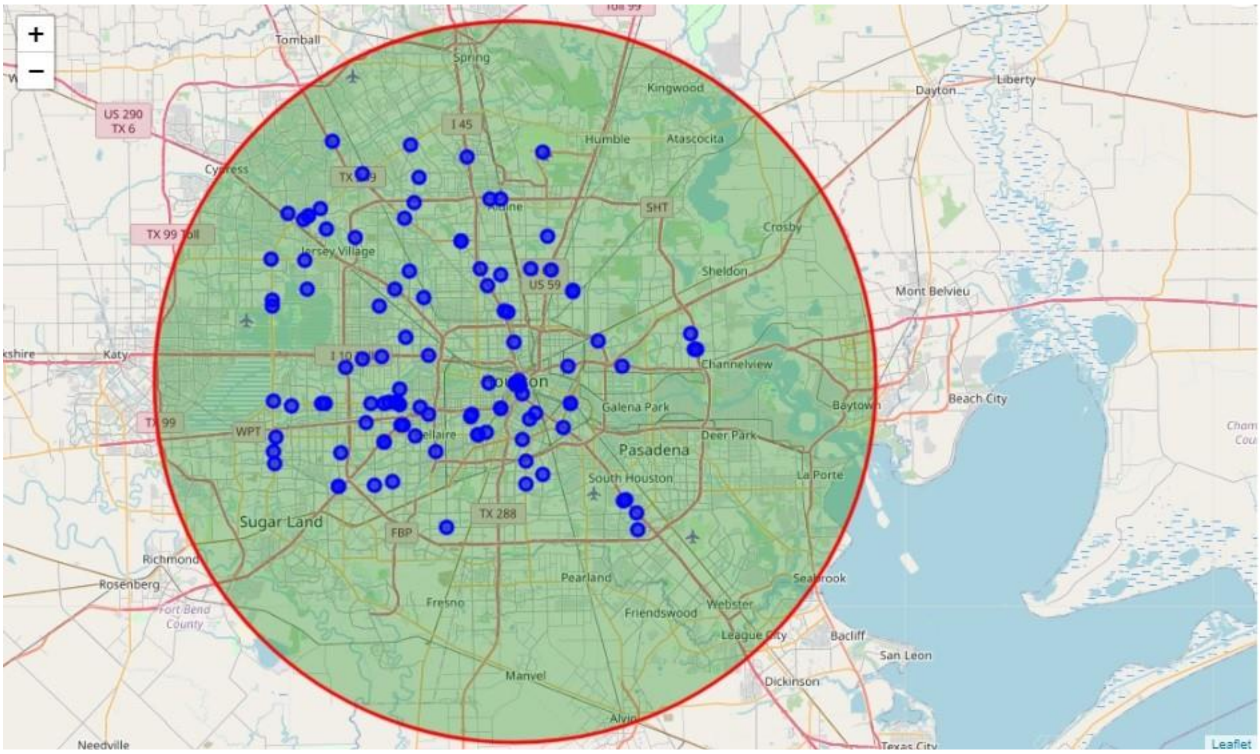
Credentials Hidden Deliberately

```
Your credentails:
CLIENT_ID: [redacted]
CLIENT_SECRET:[redacted]
Credentials Defined!
```

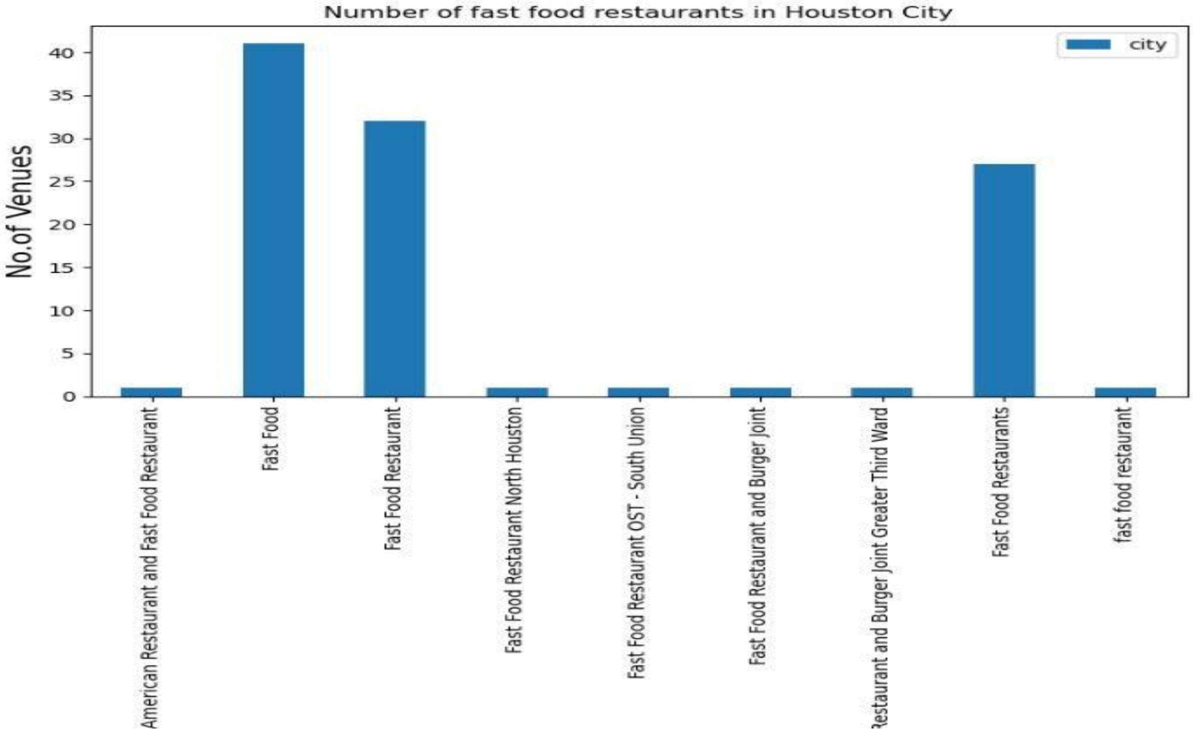
Credentials Hidden Deliberately

Foursquare API, further helped us to surface only the results related to Fast Food Restaurants option and the new DataFrame. Yet we plotted them on folium in order to get some idea that how populated the city is with the Fast food

Restaurants and which area needs one so that we could decide on it. The call included a radius of 1000 meters and a limit of 100 venues only.

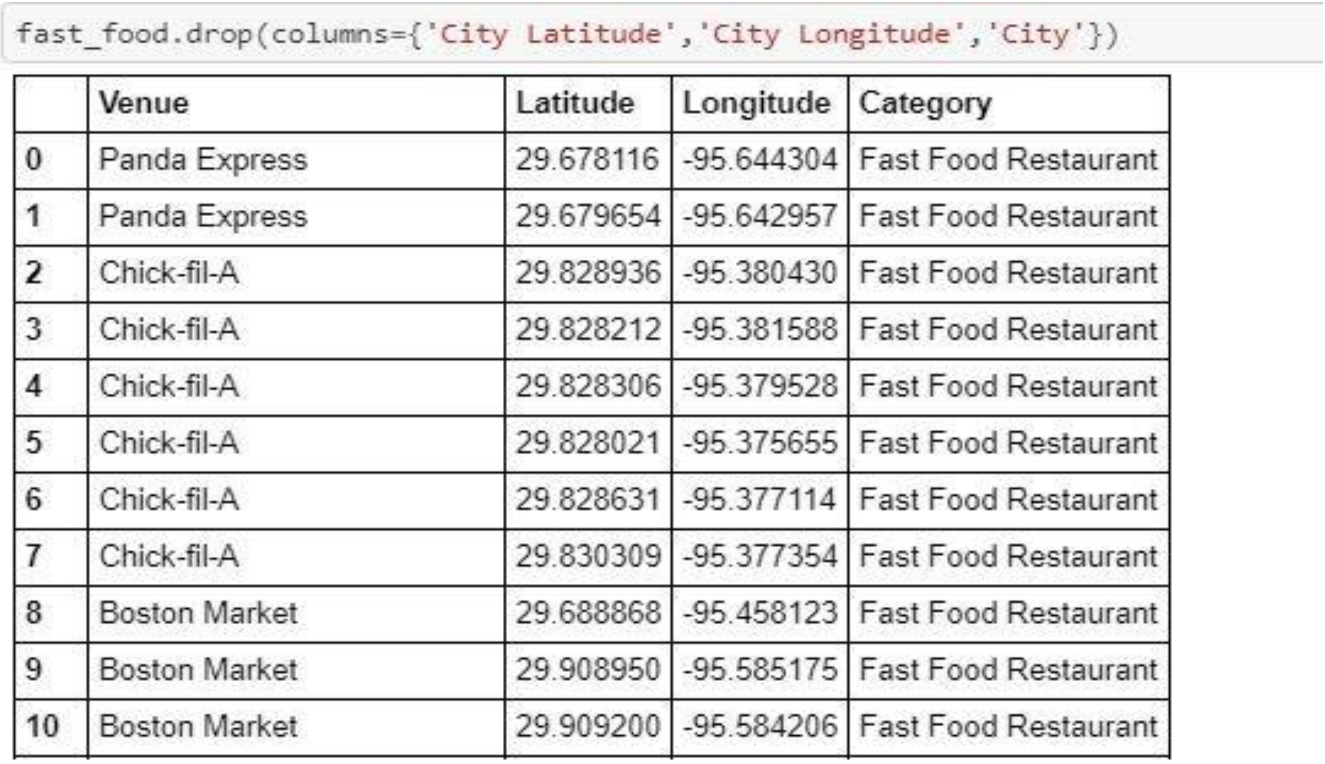


The plot clearly shows that there are at least 9 different categories that resonate with the “Fast Food Restaurants” which in fact is somewhat troubling for us and our new DataFrame needs more cleaning in order to get the real picture when we decide on it.



This posited a new challenge in order to get all the categories under one head, especially the ones like “Fast Food Restaurant”, “Fast Food Restaurants”, “Fast Food, and others. So we decided on making another call on Foursquare API instead of further cleaning the data. We preferred getting a new Dataframe that includes all the venues belonging to various categories and then slicing the Fast Food Restaurant form the said data to which we succeeded.

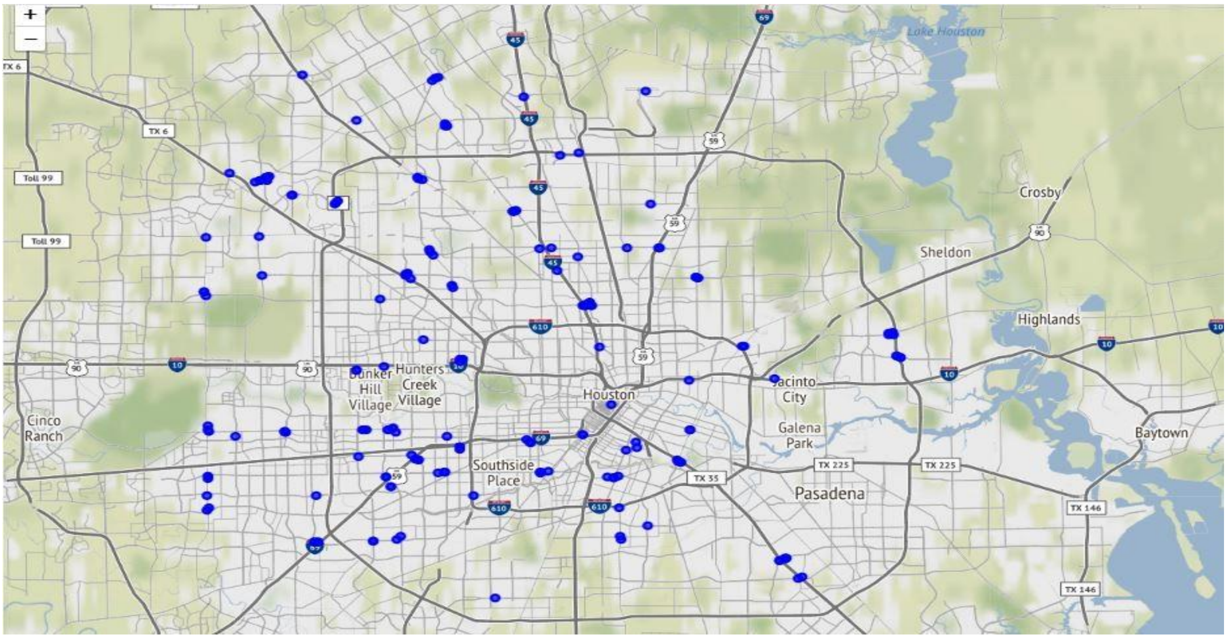
This brought us to our Final Data Frame which included all the venues in the Houston City Downtown Area and this helped us getting to our actual requisite data to be portrayed later on the folium map and get the answers to all four of our problems.



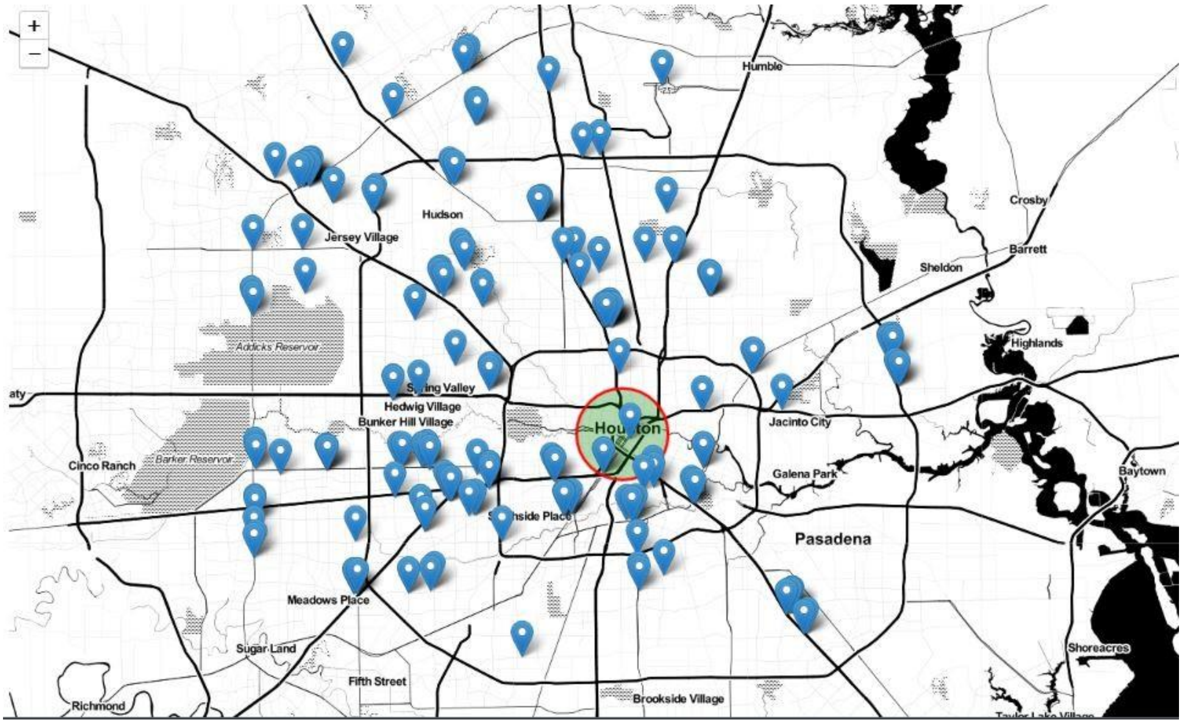
```
fast_food.drop(columns={'City Latitude','City Longitude','City'})
```

	Venue	Latitude	Longitude	Category
0	Panda Express	29.678116	-95.644304	Fast Food Restaurant
1	Panda Express	29.679654	-95.642957	Fast Food Restaurant
2	Chick-fil-A	29.828936	-95.380430	Fast Food Restaurant
3	Chick-fil-A	29.828212	-95.381588	Fast Food Restaurant
4	Chick-fil-A	29.828306	-95.379528	Fast Food Restaurant
5	Chick-fil-A	29.828021	-95.375655	Fast Food Restaurant
6	Chick-fil-A	29.828631	-95.377114	Fast Food Restaurant
7	Chick-fil-A	29.830309	-95.377354	Fast Food Restaurant
8	Boston Market	29.688868	-95.458123	Fast Food Restaurant
9	Boston Market	29.908950	-95.585175	Fast Food Restaurant
10	Boston Market	29.909200	-95.584206	Fast Food Restaurant

As we can see that the DataFrame “fast_food” includes all the venues from different companies along with their latitude and longitudes, ready to be plotted and examined accordingly.



and a in a bit more detail now;

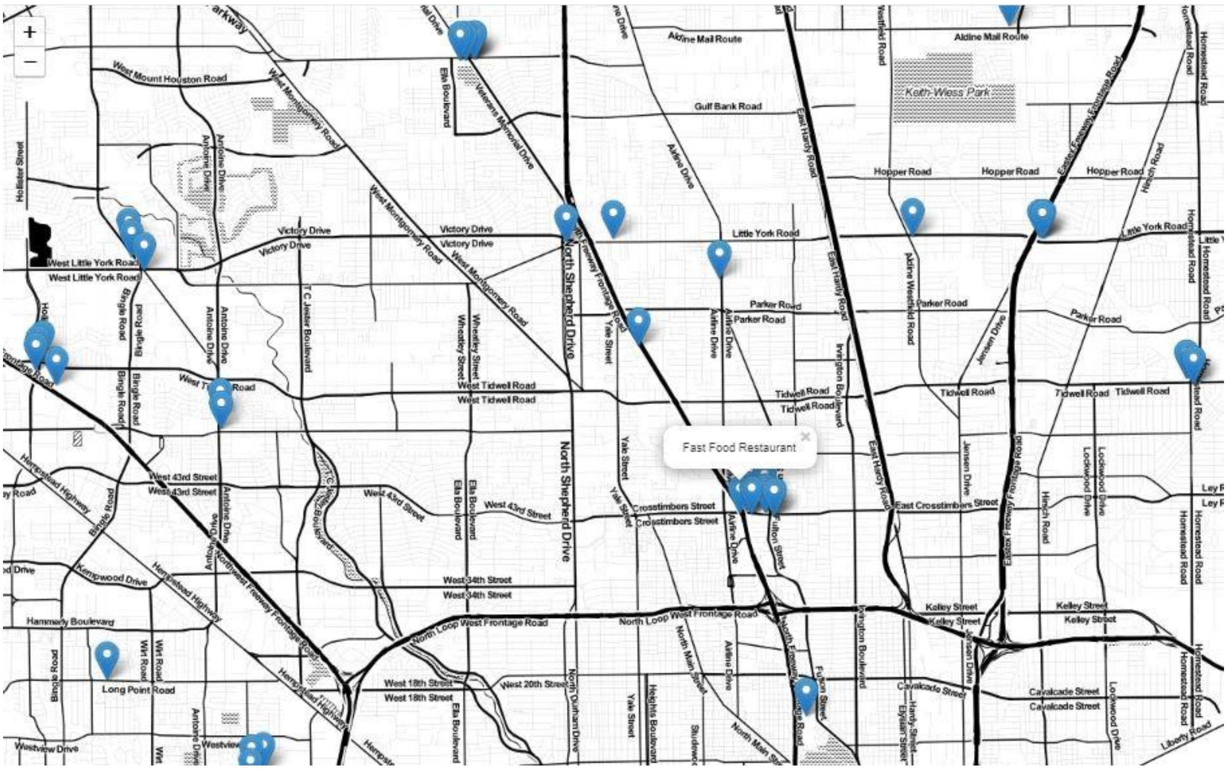


So here we have the most detailed map for anyone to look in to and get the idea where they would like to open their chain of Fast Food Restaurants in Houston Area.

Conclusion:

We can understand that a lot more could be done only if the availability of the data could have been convenient. It was a problem within a problem to find the clean and relevant data and it took us a lot of time to even conclude it. However, our main aim was to bring about a cleaner data verified through Foursquare API and visualize it in a way that the decision making becomes more convenient.

Zooming in will help Freddy look in to any specific area of Houston and see which restaurants are already located there and what gaps can he exploit to open his own restaurant.


+
−
Fast Food Restaurant
×
Aldine Mail Route
West Mount Houston Road
Gulf Bank Road
Keith-Wiess Park
Hopper Road
Little York Road
West Little York Road
Victory Drive
West Montgomery Road
Veterans Memorial Drive
Ella Boulevard
Antoine Drive
Hollister Street
Bingle Road
T C Jester Boulevard
Wheatley Street
North Shepherd Drive
North Freeway Frontage Road
Yale Street
Aldine Drive
Airline Drive
Parker Road
East Hardy Road
Irvington Boulevard
Aldine Westfield Road
Jensen Drive
Eastex Freeway Frontage Road
Homestead Road
West Tidwell Road
Tidwell Road
West 43rd Street
Crosstimbers Street
East Crosstimbers Street
Fulton Street
Hinch Road
Lockwood Drive
Hempstead Highway
Kempwood Drive
Hammerly Boulevard
Northwest Freeway Frontage Road
West 34th Street
North Loop West Frontage Road
Kelley Street
North Main Street
Long Point Road
Wirt Road
West 18th Street
West 20th Street
North Durham Drive
Heights Boulevard
Studewood
Cavalcade Street
Hardy Street
Elysian Street
Liberty Road
Westview Drive