

In [98]:

```
!pip install beautifulsoup4
```

```
Requirement already satisfied: beautifulsoup4 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (4.9.3)
```

```
Requirement already satisfied: soupsieve>1.2; python_version >= "3.0" in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from beautifulsoup4) (2.1)
```

In [99]:

```
!pip install lxml
```

```
Requirement already satisfied: lxml in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (4.6.2)
```

In [100]:

```
!conda install -c conda-forge lxml --yes
```

```
Collecting package metadata (current_repodata.json): done  
Solving environment: done
```

```
# All requested packages already installed.
```

In [101]:

```
!pip install et_xmlfile
```

```
Requirement already satisfied: et_xmlfile in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (1.0.1)
```

Installing and importing libraries

In [102]:

```
import requests # library to handle requests
import urllib.request # library to extensive URLs
import time # to measure cell execution
from bs4 import BeautifulSoup # to extract data from HTML fines
from lxml import html # html parser

import numpy as np # core library for scientific computing
import pandas as pd # core library for data manipulation and analysis
from urllib.request import urlopen
from html.parser import HTMLParser

# To see full dataframe...
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
pd.set_option('display.width', None)
```

Scraping section to get data from wikipedia page

In [103]:

```
#URL/web page to be scraped
url = "https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M"
url
```

Out[103]:

```
'https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M'
```

In [104]:

```
# open the url using urllib.request and put the HTML into the page variable
page = urllib.request.urlopen(url)
```

In [105]:

```
# status_code returned (200) meaning request was successful and the server responded
response=requests.get(url)
print(response.status_code)
```

```
200
```

Parsing section. Taking data from HTML and converting to text, tabular format

In [106]:

```
# parsing through HTML with BeautifulSoup
soup = BeautifulSoup(response.text, 'html.parser')
soup
```

Out[106]:

```
<!DOCTYPE html>

<html class="client-nojs" dir="ltr" lang="en">
<head>
<meta charset="utf-8"/>
<title>List of postal codes of Canada: M - Wikipedia</title>
<script>document.documentElement.className="client-js";RLCONF={"wgBreakFrames":!1,"wgSeparatorTransformTable":[],"wgDigitTransformTable":[],"wgDefaultDateFormat":"dmy","wgMonthNames":["","January","February","March","April","May","June","July","August","September","October","November","December"],"wgRequestId":"YBy01QzC50E83AVzcjoszgAAJJE","wgCSPNonce":!1,"wgCanonicalNamespace":"","wgCanonicalSpecialPageName":!1,"wgNamespaceNumber":0,"wgPageName":"List_of_postal_codes_of_Canada:_M","wgTitle":"List of postal codes of Canada: M","wgCurRevisionId":1004901187,"wgRevisionId":1004901187,"wgArticleId":539066,"wgIsArticle":!0,"wgIsRedirect":!1,"wgAction":"view","wgUserName":null,"wgUserGroups":["*"],"wgCategories":["Articles with short description","Short description is different from Wikidata","Communications in Ontario","Postal codes in Canada","Toronto","Ontario-related lists"],"wgPageContentLanguage":"en","wgPageContentModel":"wikitext","wgRelevantPageName":
"List_of_postal_codes_of_Canada:_M","wgRelevantArticleId":539066,"wgIsProbablyEditable":!0,"wgRelevantPageIsProbablyEditable":!0,"wgRestrictionEdit":[],"wgRestrictionMove":[],"wgMediaViewerOnClick":!0,"wgMediaViewerEnabledByDefault":!0,"wgPopupsReferencePreviews":!1,"wgPopupsConflictsWithNavPopupGadget":!1,"wgPopupsConflictsWithRefTooltipsGadget":!0,"wgVisualEditor":{"pageLanguageCode":"en","pageLanguageDir":"ltr","pageVariantFallbacks":"en"},"wgMFDisplayWikibaseDescriptions":{"search":!0,"nearby":!0,"watchlist":!0,"tagline":!1}, "wgMESchemaEditAttemptStepOversample":!1,"wgULSCurrentAutonym":"English","wgNoticeProject":"wikipedia","wgCentralAuthMobileDomain":!1,"wgEditSubmitButtonLabelPublish":!0,"wgULSPosition":"interlanguage","wgWikibaseItemId":Q3248240"};RLSTATE={"ext.globalCssJs.user.styles":"ready","site.styles":"ready","noscript":"ready","user.styles":"ready","ext.globalCssJs.user":"ready","user":"ready","user.options":"loading","ext.cite.styles":"ready","skins.vector.styles.legacy":"ready","jquery.tablesorter.styles":"ready","ext.visualEditor.desktopArticleTarget.noscript":"ready","ext.uls.interlanguage":"ready","ext.wikimediaBadges":"ready","wikibase.client.init": "ready"};RLPAGEMODULES=["ext.cite.ux-enhancements","site","mediawiki.page.ready","jquery.tablesorter","skins.vector.legacy.js","ext.gadget.ReferenceToolips","ext.gadget.charinsert","ext.gadget.extra-toolbar-buttons","ext.gadget.referenceToolbar","ext.gadget.switcher","ext.centralauth.centralautologin","mmv.head","mmv.bootstrap.autostart","ext.popups","ext.visualEditor.desktopArticleTarget.init","ext.visualEditor.targetLoader","ext.eventLogging","ext.wikimediaEvents","ext.navigationTiming","ext.uls.compactlinks","ext.uls.interface","ext.cx.eventlogging.campaigns","ext.centralNotice.geoIP","ext.centralNotice.startUp"];</script>
<script>(RLQ=window.RLQ||[]).push(function(){mw.loader.implement("user.options@hzgi",function($,jQuery,require,module){/*@nomin*/mw.user.tokens.set({"patrolToken":"+\\","watchToken":"+\\","csrfToken":"+\\"});});};</script>
<link href="/w/load.php?lang=en&modules=ext.cite.styles%7Cext.uls.interlanguage%7Cext.visualEditor.desktopArticleTarget.noscript%7Cext.wikimediaBadges%7Cjquery.tablesorter.styles%7Cskins.vector.styles.legacy%7Cwikibase.client.init&only=styles&skin=vector" rel="stylesheet"/>
<script async="" src="/w/load.php?lang=en&modules=startup&only=scripts&raw=1&skin=vector"></script>
<meta content="" name="ResourceLoaderDynamicStyles"/>
```

```
<link href="/w/load.php?lang=en&modules=site.styles&only=styles&s
kin=vector" rel="stylesheet"/>
<meta content="MediaWiki 1.36.0-wmf.27" name="generator"/>
<meta content="origin" name="referrer"/>
<meta content="origin-when-crossorigin" name="referrer"/>
<meta content="origin-when-cross-origin" name="referrer"/>
<link href="//upload.wikimedia.org" rel="preconnect"/>
<link href="//en.m.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M" medi
a="only screen and (max-width: 720px)" rel="alternate"/>
<link href="/w/index.php?title=List_of_postal_codes_of_Canada:_M&action=e
dit" rel="alternate" title="Edit this page" type="application/x-wiki"/>
<link href="/w/index.php?title=List_of_postal_codes_of_Canada:_M&action=e
dit" rel="edit" title="Edit this page"/>
<link href="/static/apple-touch/wikipedia.png" rel="apple-touch-icon"/>
<link href="/static/favicon/wikipedia.ico" rel="shortcut icon"/>
<link href="/w/opensearch_desc.php" rel="search" title="Wikipedia (en)" type
="application/opensearchdescription+xml"/>
<link href="//en.wikipedia.org/w/api.php?action=rsd" rel="EditURI" type="appl
ication/rsd+xml"/>
<link href="//creativecommons.org/licenses/by-sa/3.0/" rel="license"/>
<link href="https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M"
rel="canonical"/>
<link href="//login.wikimedia.org" rel="dns-prefetch"/>
<link href="//meta.wikimedia.org" rel="dns-prefetch"/>
</head>
<body class="mediawiki ltr sitedir-ltr mw-hide-empty-elt ns-0 ns-subject mw-e
ditable page-List_of_postal_codes_of_Canada_M rootpage-List_of_postal_codes_o
f_Canada_M skin-vector action-view skin-vector-legacy"><div class="noprint" i
d="mw-page-base"></div>
<div class="noprint" id="mw-head-base"></div>
<div class="mw-body" id="content" role="main">
<a id="top"></a>
<div class="mw-body-content" id="siteNotice"><!-- CentralNotice --></div>
<div class="mw-indicators mw-body-content">
</div>
<h1 class="firstHeading" id="firstHeading" lang="en">List of postal codes of
Canada: M</h1>
<div class="mw-body-content" id="bodyContent">
<div class="noprint" id="siteSub">From Wikipedia, the free encyclopedia</div>
<div id="contentSub"></div>
<div id="contentSub2"></div>
<div id="jump-to-nav"></div>
<a class="mw-jump-link" href="#mw-head">Jump to navigation</a>
<a class="mw-jump-link" href="#searchInput">Jump to search</a>
<div class="mw-content-ltr" dir="ltr" id="mw-content-text" lang="en"><div cla
ss="mw-parser-output"><div class="shortdescription nomobile noexcerpt nowrap
searchaux" style="display:none">Wikipedia list article</div>
<p>This is a list of <a href="/wiki/Postal_codes_in_Canada" title="Postal cod
es in Canada">postal codes in Canada</a> where the first letter is M. Postal
codes beginning with M are located within the city of <a href="/wiki/Toronto"
title="Toronto">Toronto</a> in the province of <a href="/wiki/Ontario" title
="Ontario">Ontario</a>. Only the first three characters are listed, correspon
ding to the Forward Sortation Area.</p><p><a href="/wiki/Canada_Post" title="Canada Post">Canada Post</a> provid
es a free postal code look-up tool on its website,<sup class="reference" id
="cite_ref-1"><a href="#cite_note-1">[1]</a></sup> via its <a href="/wiki/Mob
ile_app" title="Mobile app">applications</a> for such <a class="mw-redirect"

```

[smartphones](/wiki/Smartphones "Smartphones") as the [iPhone](/wiki/IPhone "iPhone") and [BlackBerry](/wiki/BlackBerry "BlackBerry"),^{[\[2\]](#cite_note-2)} and sells hard-copy directories and [CD-ROMs](/wiki/CD-ROM "CD-ROM"). Many vendors also sell validation tools, which allow customers to properly match addresses and postal codes. Hard-copy directories can also be consulted in all post offices, and some libraries.

</p>

<h2>Toronto - 103 FSAs[edit]</h2>

<p>Note: There are no rural FSAs in Toronto, hence no postal codes should start with M0. However, the postal code M0R 8T0 is assigned to an [Amazon](/wiki/Amazon_(company) "Amazon (company)") warehouse in Mississauga, and the postal code M0R 2A2 is used for the Gateway postal facility in Mississauga, suggesting that Canada Post may have reserved the M0 FSA for high volume addresses.

</p>

<table class="wikitable sortable">

<tbody><tr>

<th>Postal Code

</th>

<th>Borough

</th>

<th>Neighbourhood

</th></tr>

<tr>

<td>M1A

</td>

<td>Not assigned

</td>

<td>Not assigned

</td></tr>

<tr>

<td>M2A

</td>

<td>Not assigned

</td>

<td>Not assigned

</td></tr>

<tr>

<td>M3A

</td>

<td>North York

</td>

<td>Parkwoods

</td></tr>

<tr>

<td>M4A

</td>

<td>North York

</td>

```
<td>Victoria Village
</td></tr>
<tr>
<td>M5A
</td>
<td>Downtown Toronto
</td>
<td>Regent Park, Harbourfront
</td></tr>
<tr>
<td>M6A
</td>
<td>North York
</td>
<td>Lawrence Manor, Lawrence Heights
</td></tr>
<tr>
<td>M7A
</td>
<td>Downtown Toronto
</td>
<td>Queen's Park, Ontario Provincial Government
</td></tr>
<tr>
<td>M8A
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9A
</td>
<td>Etobicoke
</td>
<td>Islington Avenue, Humber Valley Village
</td></tr>
<tr>
<td>M1B
</td>
<td>Scarborough
</td>
<td>Malvern, Rouge
</td></tr>
<tr>
<td>M2B
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3B
</td>
<td>North York
</td>
<td>Don Mills
```

```
</td></tr>
<tr>
<td>M4B
</td>
<td>East York
</td>
<td>Parkview Hill, Woodbine Gardens
</td></tr>
<tr>
<td>M5B
</td>
<td>Downtown Toronto
</td>
<td>Garden District, Ryerson
</td></tr>
<tr>
<td>M6B
</td>
<td>North York
</td>
<td>Glencairn
</td></tr>
<tr>
<td>M7B
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8B
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9B
</td>
<td>Etobicoke
</td>
<td>West Deane Park, Princess Gardens, Martin Grove, Islington, Cloverdale
</td></tr>
<tr>
<td>M1C
</td>
<td>Scarborough
</td>
<td>Rouge Hill, Port Union, Highland Creek
</td></tr>
<tr>
<td>M2C
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
```

```
<tr>
<td>M3C
</td>
<td>North York
</td>
<td>Don Mills
</td></tr>
<tr>
<td>M4C
</td>
<td>East York
</td>
<td>Woodbine Heights
</td></tr>
<tr>
<td>M5C
</td>
<td>Downtown Toronto
</td>
<td>St. James Town
</td></tr>
<tr>
<td>M6C
</td>
<td>York
</td>
<td>Humewood-Cedarvale
</td></tr>
<tr>
<td>M7C
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8C
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9C
</td>
<td>Etobicoke
</td>
<td>Eringate, Bloordale Gardens, Old Burnhamthorpe, Markland Wood
</td></tr>
<tr>
<td>M1E
</td>
<td>Scarborough
</td>
<td>Guildwood, Morningside, West Hill
</td></tr>
<tr>
```

```
<td>M2E
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3E
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4E
</td>
<td>East Toronto
</td>
<td>The Beaches
</td></tr>
<tr>
<td>M5E
</td>
<td>Downtown Toronto
</td>
<td>Berczy Park
</td></tr>
<tr>
<td>M6E
</td>
<td>York
</td>
<td>Caledonia-Fairbanks
</td></tr>
<tr>
<td>M7E
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8E
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9E
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1G
```

```
</td>
<td>Scarborough
</td>
<td>Woburn
</td></tr>
<tr>
<td>M2G
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3G
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4G
</td>
<td>East York
</td>
<td>Leaside
</td></tr>
<tr>
<td>M5G
</td>
<td>Downtown Toronto
</td>
<td>Central Bay Street
</td></tr>
<tr>
<td>M6G
</td>
<td>Downtown Toronto
</td>
<td>Christie
</td></tr>
<tr>
<td>M7G
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8G
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9G
</td>
```

```
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1H
</td>
<td>Scarborough
</td>
<td>Cedarbrae
</td></tr>
<tr>
<td>M2H
</td>
<td>North York
</td>
<td>Hillcrest Village
</td></tr>
<tr>
<td>M3H
</td>
<td>North York
</td>
<td>Bathurst Manor, Wilson Heights, Downsview North
</td></tr>
<tr>
<td>M4H
</td>
<td>East York
</td>
<td>Thorncliffe Park
</td></tr>
<tr>
<td>M5H
</td>
<td>Downtown Toronto
</td>
<td>Richmond, Adelaide, King
</td></tr>
<tr>
<td>M6H
</td>
<td>West Toronto
</td>
<td>Dufferin, Dovercourt Village
</td></tr>
<tr>
<td>M7H
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8H
</td>
<td>Not assigned
```

```
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9J
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1J
</td>
<td>Scarborough
</td>
<td>Scarborough Village
</td></tr>
<tr>
<td>M2J
</td>
<td>North York
</td>
<td>Fairview, Henry Farm, Oriole
</td></tr>
<tr>
<td>M3J
</td>
<td>North York
</td>
<td>Northwood Park, York University
</td></tr>
<tr>
<td>M4J
</td>
<td>East York
</td>
<td>East Toronto, Broadview North (Old East York)
</td></tr>
<tr>
<td>M5J
</td>
<td>Downtown Toronto
</td>
<td>Harbourfront East, Union Station, Toronto Islands
</td></tr>
<tr>
<td>M6J
</td>
<td>West Toronto
</td>
<td>Little Portugal, Trinity
</td></tr>
<tr>
<td>M7J
</td>
<td>Not assigned
</td>
```

```
<td>Not assigned
</td></tr>
<tr>
<td>M8J
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9J
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1K
</td>
<td>Scarborough
</td>
<td>Kennedy Park, Ionview, East Birchmount Park
</td></tr>
<tr>
<td>M2K
</td>
<td>North York
</td>
<td>Bayview Village
</td></tr>
<tr>
<td>M3K
</td>
<td>North York
</td>
<td>Downsview
</td></tr>
<tr>
<td>M4K
</td>
<td>East Toronto
</td>
<td>The Danforth West, Riverdale
</td></tr>
<tr>
<td>M5K
</td>
<td>Downtown Toronto
</td>
<td>Toronto Dominion Centre, Design Exchange
</td></tr>
<tr>
<td>M6K
</td>
<td>West Toronto
</td>
<td>Brockton, Parkdale Village, Exhibition Place
```

```
</td></tr>
<tr>
<td>M7K
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8K
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9K
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1L
</td>
<td>Scarborough
</td>
<td>Golden Mile, Clairlea, Oakridge
</td></tr>
<tr>
<td>M2L
</td>
<td>North York
</td>
<td>York Mills, Silver Hills
</td></tr>
<tr>
<td>M3L
</td>
<td>North York
</td>
<td>Downsview
</td></tr>
<tr>
<td>M4L
</td>
<td>East Toronto
</td>
<td>India Bazaar, The Beaches West
</td></tr>
<tr>
<td>M5L
</td>
<td>Downtown Toronto
</td>
<td>Commerce Court, Victoria Hotel
</td></tr>
```

```
<tr>
<td>M6L
</td>
<td>North York
</td>
<td>North Park, Maple Leaf Park, Upwood Park
</td></tr>
<tr>
<td>M7L
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8L
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9L
</td>
<td>North York
</td>
<td>Humber Summit
</td></tr>
<tr>
<td>M1M
</td>
<td>Scarborough
</td>
<td>Cliffside, Cliffcrest, Scarborough Village West
</td></tr>
<tr>
<td>M2M
</td>
<td>North York
</td>
<td>Willowdale, Newtonbrook
</td></tr>
<tr>
<td>M3M
</td>
<td>North York
</td>
<td>Downsview
</td></tr>
<tr>
<td>M4M
</td>
<td>East Toronto
</td>
<td>Studio District
</td></tr>
<tr>
```

```
<td>M5M
</td>
<td>North York
</td>
<td>Bedford Park, Lawrence Manor East
</td></tr>
<tr>
<td>M6M
</td>
<td>York
</td>
<td>Del Ray, Mount Dennis, Keelsdale and Silverthorn
</td></tr>
<tr>
<td>M7M
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8M
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9M
</td>
<td>North York
</td>
<td>Humberlea, Emery
</td></tr>
<tr>
<td>M1N
</td>
<td>Scarborough
</td>
<td>Birch Cliff, Cliffside West
</td></tr>
<tr>
<td>M2N
</td>
<td>North York
</td>
<td>Willowdale, Willowdale East
</td></tr>
<tr>
<td>M3N
</td>
<td>North York
</td>
<td>Downsview
</td></tr>
<tr>
<td>M4N
```

```
</td>
<td>Central Toronto
</td>
<td>Lawrence Park
</td></tr>
<tr>
<td>M5N
</td>
<td>Central Toronto
</td>
<td>Roselawn
</td></tr>
<tr>
<td>M6N
</td>
<td>York
</td>
<td>Runnymede, The Junction North
</td></tr>
<tr>
<td>M7N
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8N
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9N
</td>
<td>York
</td>
<td>Weston
</td></tr>
<tr>
<td>M1P
</td>
<td>Scarborough
</td>
<td>Dorset Park, Wexford Heights, Scarborough Town Centre
</td></tr>
<tr>
<td>M2P
</td>
<td>North York
</td>
<td>York Mills West
</td></tr>
<tr>
<td>M3P
</td>
```

```
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4P
</td>
<td>Central Toronto
</td>
<td>Davisville North
</td></tr>
<tr>
<td>M5P
</td>
<td>Central Toronto
</td>
<td>Forest Hill North & West, Forest Hill Road Park
</td></tr>
<tr>
<td>M6P
</td>
<td>West Toronto
</td>
<td>High Park, The Junction South
</td></tr>
<tr>
<td>M7P
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8P
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9P
</td>
<td>Etobicoke
</td>
<td>Westmount
</td></tr>
<tr>
<td>M1R
</td>
<td>Scarborough
</td>
<td>Wexford, Maryvale
</td></tr>
<tr>
<td>M2R
</td>
<td>North York
```

```
</td>
<td>Willowdale, Willowdale West
</td></tr>
<tr>
<td>M3R
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4R
</td>
<td>Central Toronto
</td>
<td>North Toronto West, Lawrence Park
</td></tr>
<tr>
<td>M5R
</td>
<td>Central Toronto
</td>
<td>The Annex, North Midtown, Yorkville
</td></tr>
<tr>
<td>M6R
</td>
<td>West Toronto
</td>
<td>Parkdale, Roncesvalles
</td></tr>
<tr>
<td>M7R
</td>
<td>Mississauga
</td>
<td>Canada Post Gateway Processing Centre
</td></tr>
<tr>
<td>M8R
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9R
</td>
<td>Etobicoke
</td>
<td>Kingsview Village, St. Phillips, Martin Grove Gardens, Richview Gardens
</td></tr>
<tr>
<td>M1S
</td>
<td>Scarborough
</td>
```

```
<td>Agincourt
</td></tr>
<tr>
<td>M2S
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3S
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4S
</td>
<td>Central Toronto
</td>
<td>Davisville
</td></tr>
<tr>
<td>M5S
</td>
<td>Downtown Toronto
</td>
<td>University of Toronto, Harbord
</td></tr>
<tr>
<td>M6S
</td>
<td>West Toronto
</td>
<td>Runnymede, Swansea
</td></tr>
<tr>
<td>M7S
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8S
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M9S
</td>
<td>Not assigned
</td>
<td>Not assigned
```

```
</td></tr>
<tr>
<td>M1T
</td>
<td>Scarborough
</td>
<td>Clarks Corners, Tam O'Shanter, Sullivan
</td></tr>
<tr>
<td>M2T
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3T
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4T
</td>
<td>Central Toronto
</td>
<td>Moore Park, Summerhill East
</td></tr>
<tr>
<td>M5T
</td>
<td>Downtown Toronto
</td>
<td>Kensington Market, Chinatown, Grange Park
</td></tr>
<tr>
<td>M6T
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M7T
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8T
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
```

```
<tr>
<td>M9T
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1V
</td>
<td>Scarborough
</td>
<td>Milliken, Agincourt North, Steeles East, L'Amoreaux East
</td></tr>
<tr>
<td>M2V
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3V
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4V
</td>
<td>Central Toronto
</td>
<td>Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park
</td></tr>
<tr>
<td>M5V
</td>
<td>Downtown Toronto
</td>
<td>CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport
</td></tr>
<tr>
<td>M6V
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M7V
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
```

```
<tr>
<td>M8V
</td>
<td>Etobicoke
</td>
<td>New Toronto, Mimico South, Humber Bay Shores
</td></tr>
<tr>
<td>M9V
</td>
<td>Etobicoke
</td>
<td>South Steeles, Silverstone, Humbergate, Jamestown, Mount Olive, Beaumont
Heights, Thistletown, Albion Gardens
</td></tr>
<tr>
<td>M1W
</td>
<td>Scarborough
</td>
<td>Steeles West, L'Amoreaux West
</td></tr>
<tr>
<td>M2W
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3W
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4W
</td>
<td>Downtown Toronto
</td>
<td>Rosedale
</td></tr>
<tr>
<td>M5W
</td>
<td>Downtown Toronto
</td>
<td>Stn A PO Boxes
</td></tr>
<tr>
<td>M6W
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
```

```
<tr>
<td>M7W
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8W
</td>
<td>Etobicoke
</td>
<td>Alderwood, Long Branch
</td></tr>
<tr>
<td>M9W
</td>
<td>Etobicoke
</td>
<td>Northwest, West Humber - Clairville
</td></tr>
<tr>
<td>M1X
</td>
<td>Scarborough
</td>
<td>Upper Rouge
</td></tr>
<tr>
<td>M2X
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3X
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4X
</td>
<td>Downtown Toronto
</td>
<td>St. James Town, Cabbagetown
</td></tr>
<tr>
<td>M5X
</td>
<td>Downtown Toronto
</td>
<td>First Canadian Place, Underground city
</td></tr>
<tr>
```

```
<td>M6X
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M7X
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M8X
</td>
<td>Etobicoke
</td>
<td>The Kingsway, Montgomery Road, Old Mill North
</td></tr>
<tr>
<td>M9X
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1Y
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M2Y
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3Y
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M4Y
</td>
<td>Downtown Toronto
</td>
<td>Church and Wellesley
</td></tr>
<tr>
<td>M5Y
```

```
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M6Y
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M7Y
</td>
<td>East Toronto
</td>
<td>Business reply mail Processing Centre, South Central Letter Processing Pl
ant Toronto
</td></tr>
<tr>
<td>M8Y
</td>
<td>Etobicoke
</td>
<td>Old Mill South, King's Mill Park, Sunnylea, Humber Bay, Mimico NE, The Qu
eensway East, Royal York South East, Kingsway Park South East
</td></tr>
<tr>
<td>M9Y
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M1Z
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M2Z
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
<td>M3Z
</td>
<td>Not assigned
</td>
<td>Not assigned
</td></tr>
<tr>
```

```
<td>M4Z  
</td>  
<td>Not assigned  
</td>  
<td>Not assigned  
</td></tr>  
<tr>  
<td>M5Z  
</td>  
<td>Not assigned  
</td>  
<td>Not assigned  
</td></tr>  
<tr>  
<td>M6Z  
</td>  
<td>Not assigned  
</td>  
<td>Not assigned  
</td></tr>  
<tr>  
<td>M7Z  
</td>  
<td>Not assigned  
</td>  
<td>Not assigned  
</td></tr>  
<tr>  
<td>M8Z  
</td>  
<td>Etobicoke  
</td>  
<td>Mimico NW, The Queensway West, South of Bloor, Kingsway Park South West,  
Royal York South West  
</td></tr>  
<tr>  
<td>M9Z  
</td>  
<td>Not assigned  
</td>  
<td>Not assigned  
</td></tr></tbody></table>  
<h2><span id="Most_populated_FSAs.5B3.5D"></span><span class="mw-headline" id="Most_populated_FSAs[3]">Most populated FSAs<sup class="reference" id="cite_ref-statcan_3-0"><a href="#cite_note-statcan-3">[3]</a></sup></span><span class="mw-editsection"><span class="mw-editsection-bracket">[</span><a href="/w/index.php?title=List_of_postal_codes_of_Canada:_M&action=edit&section=2" title="Edit section: Most populated FSAs[3]">edit</a><span class="mw-editsection-bracket">]</span></span></h2>  
<ol><li>M1B, 65,129</li>  
<li>M2N, 60,124</li>  
<li>M1V, 55,250</li>  
<li>M9V, 55,159</li>  
<li>M2J, 54,391</li></ol>  
<h2><span id="Least_populated_FSAs.5B3.5D"></span><span class="mw-headline" id="Least_populated_FSAs[3]">Least populated FSAs<sup class="reference" id="cite_ref-statcan_3-1"><a href="#cite_note-statcan-3">[3]</a></sup></span><span
```

```
class="mw-editsection"><span class="mw-editsection-bracket">[</span><a href
="/w/index.php?title=List_of_postal_codes_of_Canada:_M&action=edit&se
ction=3" title="Edit section: Least populated FSAs[3]">edit</a><span class="m
w-editsection-bracket">]</span></span></h2>
<ol><li>M5K, 5</li>
<li>M5L, 5</li>
<li>M5W, 5</li>
<li>M5X, 5</li>
<li>M7A, 5</li></ol>
<h2><span class="mw-headline" id="References">References</span><span class="m
w-editsection"><span class="mw-editsection-bracket">[</span><a href="/w/inde
x.php?title=List_of_postal_codes_of_Canada:_M&action=edit&section=4"
title="Edit section: References">edit</a><span class="mw-editsection-bracke
t">]</span></span></h2>
<div class="mw-references-wrap"><ol class="references">
<li id="cite_note-1"><span class="mw-cite-backlink"><b><a href="#cite_ref-1">
^</a></b></span> <span class="reference-text">
<style data-mw-deduplicate="TemplateStyles:r999302996">.mw-parser-output cit
e.citation{font-style:inherit}.mw-parser-output .citation q{quote
s:""""""}.mw-parser-output .id-lock-free a,.mw-parser-output .citatio
n .cs1-lock-free a{background:linear-gradient(transparent,transparent),url
("//upload.wikimedia.org/wikipedia/commons/6/65/Lock-green.svg")right 0.1em c
enter/9px no-repeat}.mw-parser-output .id-lock-limited a,.mw-parser-output .i
d-lock-registration a,.mw-parser-output .citation .cs1-lock-limited a,.mw-par
ser-output .citation .cs1-lock-registration a{background:linear-gradient(tran
sparent,transparent),url("//upload.wikimedia.org/wikipedia/commons/d/d6/Lock-
gray-alt-2.svg")right 0.1em center/9px no-repeat}.mw-parser-output .id-lock-s
ubscription a,.mw-parser-output .citation .cs1-lock-subscription a{backgroun
d:linear-gradient(transparent,transparent),url("//upload.wikimedia.org/wikipe
dia/commons/a/aa/Lock-red-alt-2.svg")right 0.1em center/9px no-repeat}.mw-par
ser-output .cs1-subscription,.mw-parser-output .cs1-registration{color:#555}.
mw-parser-output .cs1-subscription span,.mw-parser-output .cs1-registration s
pan{border-bottom:1px dotted;cursor:help}.mw-parser-output .cs1-ws-icon a{bac
kground:linear-gradient(transparent,transparent),url("//upload.wikimedia.org/
wikipedia/commons/4/4c/Wikisource-logo.svg")right 0.1em center/12px no-repe
t}.mw-parser-output code.cs1-code{color:inherit;background:inherit;border:none
;padding:inherit}.mw-parser-output .cs1-hidden-error{display:none;font-size:
100%}.mw-parser-output .cs1-visible-error{font-size:100%}.mw-parser-output .c
s1-maint{display:none;color:#33aa33;margin-left:0.3em}.mw-parser-output .cs1-
format{font-size:95%}.mw-parser-output .cs1-kern-left,.mw-parser-output .cs1-
kern-wl-left{padding-left:0.2em}.mw-parser-output .cs1-kern-right,.mw-parser-
output .cs1-kern-wl-right{padding-right:0.2em}.mw-parser-output .citation .mw
-selflink{font-weight:inherit}</style><cite class="citation web cs1" id="CITE
REFCanada_Post">Canada Post. <a class="external text" href="https://www.canad
apost.ca/cpotools/apps/fpc/personal/findByCity?execution=e2s1" rel="nofollo
w">Canada Post - Find a Postal Code</a><span class="reference-accessdate">.
Retrieved <span class="nowrap">9 November</span> 2008</span>.</cite><span cla
ss="Z3988" title="ctx_ver=Z39.88-2004&rft_val_fmt=info%3Aofi%2Ffmt%3Akev%
3Amix%3Abook&genre=unknown&rft_bttitle=Canada+Post+-+Find+a+Postal
+Code&rft.au=Canada+Post&rft_id=https%3A%2F%2Fwww.canadapost.ca%2Fcpo
tools%2Fapps%2Ffpc%2Fpersonal%2FfindByCity%3Fexecution%3De2s1&rfr_id=
o%3Asid%2Fen.wikipedia.org%3AList+of+postal+codes+of+Canada%3A+M"></span></sp
an>
</li>
<li id="cite_note-2"><span class="mw-cite-backlink"><b><a href="#cite_ref-2">
^</a></b></span> <span class="reference-text"><link href="mw-data:TemplateSty
les:r999302996" rel="mw-duplicated-inline-style"/><cite class="citation web
```

```
cs1"><a class="external text" href="https://web.archive.org/web/20110519093024/http://www.canadapost.ca/cpo/mc/personal/tools/mobileapp/default.jsf" rel="nofollow">Mobile Apps</a>. Canada Post. Archived from <a class="external text" href="http://www.canadapost.ca/cpo/mc/personal/tools/mobileapp/default.jsf" rel="nofollow">the original</a> on 2011-05-19.</cite><span class="Z3988" title="ctx_ver=Z39.88-2004& rft_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amx%3Abok& rft.genre=unknown& rft.btitle=Mobile+Apps& rft.pub=Canada+Post& am; rft_id=http%3A%2F%2Fwww.canadapost.ca%2Fcpo%2Fmc%2Fpersonal%2Ftools%2Fmobil eapp%2Fdefault.jsf& rfr_id=info%3Asid%2Fen.wikipedia.org%3AList+of+postal+ codes+of+Canada%3A+M"></span></span>
</li>
<li id="cite_note-statcan-3"><span class="mw-cite-backlink">^ <a href="#cite_ref-statcan_3-0"><sup><i><b>a</b></i></sup></a> <a href="#cite_ref-statcan_3-1"><sup><i><b>b</b></i></sup></a></span> <span class="reference-text"><link href="mw-data:TemplateStyles:r999302996" rel="mw-deduplicated-inline-style"/>< cite class="citation web cs1"><a class="external text" href="http://www12.statcan.ca/english/census06/data/popdwel/Table.cfm?T=1201& SR=1& S=0& O=A& RPP=9999& PR=0& CMA=0" rel="nofollow">2006 Census of Population</a>. 15 October 2008.</cite><span class="Z3988" title="ctx_ver=Z39.88-2004& rft_val_fmt=info%3Aofi%2Ffmt%3Akev%3Amx%3Abok& rft.genre=unknown& am; rft.btitle=2006+Census+of+Population& rft.date=2008-10-15& rft_id=htt p%3A%2F%2Fwww12.statcan.ca%2Fenglish%2Fcensus06%2Fdata%2Fpopdwel1%2FTable.cf m%3FT%3D1201%26SR%3D1%26S%3D0%260%3DA%26RPP%3D9999%26PR%3D0%26CMA%3D0& rfr _id=info%3Asid%2Fen.wikipedia.org%3AList+of+postal+codes+of+Canada%3A+M"></sp an></span>
</li>
</ol></div>
<table class="navbox">
<tbody><tr>
<td style="width:36px; text-align:center"><a class="image" href="/wiki/File:Flag_of_Canada.svg" title="Flag of Canada">
</a>
</td>
<th class="navbox-title" style="font-size:110%"><a href="/wiki/Postal_codes_in_Canada" title="Postal codes in Canada">Canadian postal codes</a>
</th>
<td style="width:36px; text-align:center"><a class="image" href="/wiki/File:Canadian_postal_district_map_(without_legends).svg">
</a>
</td></tr>
<tr>
<td colspan="3" style="text-align:center; font-size: 100%;">
<table cellspacing="0" style="background-color: #F8F8F8;" width="100%">
<tbody><tr>
```

```
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Newfoundland_and_Labrador" title="Newfoundland and Labrador">NL</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Nova_Scotia" title="Nova Scotia">NS</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Prince_Edward_Island" title="Prince Edward Island">PE</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/New_Brunswick" title="New Brunswick">NB</a>
</td>
<td colspan="3" style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Quebec" title="Quebec">QC</a>
</td>
<td colspan="5" style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Ontario" title="Ontario">ON</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Manitoba" title="Manitoba">MB</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Saskatchewan" title="Saskatchewan">SK</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Alberta" title="Alberta">AB</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/British_Columbia" title="British Columbia">BC</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Nunavut" title="Nunavut">NU</a><a href="/wiki/Northwest_Territories" title="Northwest Territories">NT</a>
</td>
<td style="text-align:center; border:1px solid #aaa;"><a href="/wiki/Yukon" title="Yukon">YT</a>
</td></tr>
<tr>
<td align="center" style="border: 1px solid #FF0000; background-color: #FFE0E0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canada:_A" title="List of postal codes of Canada: A">A</a>
</td>
<td align="center" style="border: 1px solid #FF4000; background-color: #FFE8E0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canada:_B" title="List of postal codes of Canada: B">B</a>
</td>
<td align="center" style="border: 1px solid #FF8000; background-color: #FFF0E0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canada:_C" title="List of postal codes of Canada: C">C</a>
</td>
<td align="center" style="border: 1px solid #FFC000; background-color: #FFF8E0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canada:_E" title="List of postal codes of Canada: E">E</a>
</td>
<td align="center" style="border: 1px solid #FFFF00; background-color: #FFFFFF; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canada:_G" title="List of postal codes of Canada: G">G</a>
```

```
</td>
<td align="center" style="border: 1px solid #C0FF00; background-color: #F8FFE
0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_H" title="List of postal codes of Canada: H">H</a>
</td>
<td align="center" style="border: 1px solid #80FF00; background-color: #F0FFE
0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_J" title="List of postal codes of Canada: J">J</a>
</td>
<td align="center" style="border: 1px solid #00FF00; background-color: #E0FFE
0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_K" title="List of postal codes of Canada: K">K</a>
</td>
<td align="center" style="border: 1px solid #00FF80; background-color: #E0FFF
0; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_L" title="List of postal codes of Canada: L">L</a>
</td>
<td align="center" style="border: 1px solid #E0FFF8; background-color: #00FFC
0; font-size: 135%; color: black;" width="5%"><a class="mw-selflink selflin
k">M</a>
</td>
<td align="center" style="border: 1px solid #00FFE0; background-color: #E0FFF
C; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_N" title="List of postal codes of Canada: N">N</a>
</td>
<td align="center" style="border: 1px solid #00FFFF; background-color: #E0FFF
F; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_P" title="List of postal codes of Canada: P">P</a>
</td>
<td align="center" style="border: 1px solid #00C0FF; background-color: #E0F8F
F; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_R" title="List of postal codes of Canada: R">R</a>
</td>
<td align="center" style="border: 1px solid #0080FF; background-color: #E0F0F
F; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_S" title="List of postal codes of Canada: S">S</a>
</td>
<td align="center" style="border: 1px solid #0040FF; background-color: #E0E8F
F; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_T" title="List of postal codes of Canada: T">T</a>
</td>
<td align="center" style="border: 1px solid #0000FF; background-color: #E0E0F
F; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_V" title="List of postal codes of Canada: V">V</a>
</td>
<td align="center" style="border: 1px solid #A000FF; background-color: #E8E0F
F; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_X" title="List of postal codes of Canada: X">X</a>
</td>
<td align="center" style="border: 1px solid #FF00FF; background-color: #FFE0F
F; font-size: 135%;" width="5%"><a href="/wiki/List_of_postal_codes_of_Canad
a:_Y" title="List of postal codes of Canada: Y">Y</a>
</td></tr></tbody></table>
</td></tr></tbody></table>
<!--
NewPP limit report
Parsed by mw1358
```

```
Cached time: 20210205001814
Cache expiry: 2592000
Dynamic content: false
Complications: [vary-revision-sha1]
CPU time usage: 0.200 seconds
Real time usage: 0.243 seconds
Preprocessor visited node count: 279/1000000
Post-expand include size: 10828/2097152 bytes
Template argument size: 275/2097152 bytes
Highest expansion depth: 8/40
Expensive parser function count: 0/500
Unstrip recursion depth: 1/20
Unstrip post-expand size: 9406/5000000 bytes
Lua time usage: 0.101/10.000 seconds
Lua memory usage: 3013916/52428800 bytes
Number of Wikibase entities loaded: 0/400
-->
<!--
Transclusion expansion time report (%,ms,calls,template)
100.00% 202.632      1 -total
 54.62% 110.679      3 Template:Cite_web
 38.51% 78.040       1 Template:Short_description
 22.61% 45.825       1 Template:Pagetype
  9.48% 19.206       2 Template:Main_other
   7.30% 14.787       1 Template:SDcat
   1.78%  3.607       1 Template:Canadian_postal_codes
-->
<!-- Saved in parser cache with key enwiki:pcache:idhash:539066-0!canonical a
nd timestamp 20210205001814 and revision id 1004901187. Serialized with JSON.
-->
</div><noscript></noscript>
<div class="printfooter">Retrieved from <a dir="ltr" href="https://en.wikipedia.org/w/index.php?title=List_of_postal_codes_of_Canada:_M&oldid=1004901187">https://en.wikipedia.org/w/index.php?title=List_of_postal_codes_of_Canada:_M&oldid=1004901187</a></div></div>
<div class="catlinks" data-mw="interface" id="catlinks"><div class="mw-normal-catlinks" id="mw-normal-catlinks"><a href="/wiki/Help:Category" title="Help:Category">Categories</a>: <ul><li><a href="/wiki/Category:Communications_in_Ontario" title="Category:Communications in Ontario">Communications in Ontario</a></li><li><a href="/wiki/Category:Postal_codes_in_Canada" title="Category:Postal codes in Canada">Postal codes in Canada</a></li><li><a href="/wiki/Category:Toronto" title="Category:Toronto">Toronto</a></li><li><a href="/wiki/Category:Ontario-related_lists" title="Category:Ontario-related lists">Ontario-related lists</a></li></ul></div><div class="mw-hidden-catlinks mw-hidden-cats-hidden" id="mw-hidden-catlinks">Hidden categories: <ul><li><a href="/wiki/Category:Articles_with_short_description" title="Category:Articles with short description">Articles with short description</a></li><li><a href="/wiki/Category:Short_description_is_different_from_Wikidata" title="Category:Short description is different from Wikidata">Short description is different from Wikidata</a></li></ul></div></div>
<div id="mw-data-after-content">
<div class="read-more-container"></div>
</div>
```

```
<div id="mw-navigation">
<h2>Navigation menu</h2>
<div id="mw-head">
<!-- Please do not use role attribute as CSS selector, it is deprecated. --&gt;
&lt;nav aria-labelledby="p-personal-label" class="mw-portlet mw-portlet-personal
vector-menu" id="p-personal" role="navigation"&gt;
&lt;h3 id="p-personal-label"&gt;
&lt;span&gt;Personal tools&lt;/span&gt;
&lt;/h3&gt;
&lt;div class="vector-menu-content"&gt;
&lt;ul class="vector-menu-content-list"&gt;&lt;li id="pt-anonuserpage"&gt;Not logged in&lt;/li&gt;&lt;li id="pt-anontalk"&gt;&lt;a accesskey="n" href="/wiki/Special:MyTalk" title="D
iscussion about edits from this IP address [n]"&gt;Talk&lt;/a&gt;&lt;/li&gt;&lt;li id="pt-anonc
ontribs"&gt;&lt;a accesskey="y" href="/wiki/Special:MyContributions" title="A list
of edits made from this IP address [y]"&gt;Contributions&lt;/a&gt;&lt;/li&gt;&lt;li id="pt-crea
teaccount"&gt;&lt;a href="/w/index.php?title=Special>CreateAccount&returnto=Lis
t+of+postal+codes+of+Canada%3A+M" title="You are encouraged to create an acco
unt and log in; however, it is not mandatory">Create account</a></li><li id
="pt-login"><a accesskey="o" href="/w/index.php?title=Special:UserLogin&r
eturnto=List+of+postal+codes+of+Canada%3A+M" title="You're encouraged to log
in; however, it's not mandatory. [o]">Log in</a></li></ul>
</div>
</nav>
<div id="left-navigation">
<!-- Please do not use role attribute as CSS selector, it is deprecated. --&gt;
&lt;nav aria-labelledby="p-namespaces-label" class="mw-portlet mw-portlet-namesp
aces vector-menu vector-menu-tabs" id="p-namespaces" role="navigation"&gt;
&lt;h3 id="p-namespaces-label"&gt;
&lt;span&gt;Namespaces&lt;/span&gt;
&lt;/h3&gt;
&lt;div class="vector-menu-content"&gt;
&lt;ul class="vector-menu-content-list"&gt;&lt;li class="selected" id="ca-nstab-main"&gt;
&lt;a accesskey="c" href="/wiki/List_of_postal_codes_of_Canada:_M" title="View t
he content page [c]"&gt;Article&lt;/a&gt;&lt;/li&gt;&lt;li id="ca-talk"&gt;&lt;a accesskey="t" href
="/wiki/Talk>List_of_postal_codes_of_Canada:_M" rel="discussion" title="Discu
ss improvements to the content page [t]">Talk</a></li></ul>
</div>
</nav>
<!-- Please do not use role attribute as CSS selector, it is deprecated. --&gt;
&lt;nav aria-labelledby="p-variants-label" class="mw-portlet mw-portlet-variants
emptyPortlet vector-menu vector-menu-dropdown" id="p-variants" role="navigati
on"&gt;
&lt;input aria-labelledby="p-variants-label" class="vector-menu-checkbox" type
="checkbox"/&gt;
&lt;h3 id="p-variants-label"&gt;
&lt;span&gt;Variants&lt;/span&gt;
&lt;/h3&gt;
&lt;div class="vector-menu-content"&gt;
&lt;ul class="vector-menu-content-list"&gt;&lt;/ul&gt;
&lt;/div&gt;
&lt;/nav&gt;
&lt;/div&gt;
&lt;div id="right-navigation"&gt;
<!-- Please do not use role attribute as CSS selector, it is deprecated. --&gt;
&lt;nav aria-labelledby="p-views-label" class="mw-portlet mw-portlet-views vecto
r-menu vector-menu-tabs" id="p-views" role="navigation"&gt;
&lt;h3 id="p-views-label"&gt;</pre>
```

```
<span>Views</span>
</h3>
<div class="vector-menu-content">
<ul class="vector-menu-content-list"><li class="selected" id="ca-view"><a href="/wiki/List_of_postal_codes_of_Canada:_M">Read</a></li><li id="ca-edit"><a accesskey="e" href="/w/index.php?title=List_of_postal_codes_of_Canada:_M&action=edit" title="Edit this page [e]">Edit</a></li><li id="ca-history"><a accesskey="h" href="/w/index.php?title=List_of_postal_codes_of_Canada:_M&action=history" title="Past revisions of this page [h]">View history</a></li>
</ul>
</div>
</nav>
<!-- Please do not use role attribute as CSS selector, it is deprecated. --&gt;
&lt;nav aria-labelledby="p-cactions-label" class="mw-portlet mw-portlet-cactions emptyPortlet vector-menu vector-menu-dropdown" id="p-cactions" role="navigation"&gt;
&lt;input aria-labelledby="p-cactions-label" class="vector-menu-checkbox" type="checkbox"/&gt;
&lt;h3 id="p-cactions-label"&gt;
&lt;span&gt;More&lt;/span&gt;
&lt;/h3&gt;
&lt;div class="vector-menu-content"&gt;
&lt;ul class="vector-menu-content-list"&gt;&lt;/ul&gt;
&lt;/div&gt;
&lt;/nav&gt;
&lt;div id="p-search" role="search"&gt;
&lt;h3&gt;
&lt;label for="searchInput"&gt;Search&lt;/label&gt;
&lt;/h3&gt;
&lt;form action="/w/index.php" id="searchform"&gt;
&lt;div data-search-loc="header-navigation" id="simpleSearch"&gt;
&lt;input accesskey="f" autocapitalize="sentences" id="searchInput" name="search" placeholder="Search Wikipedia" title="Search Wikipedia [f]" type="search" /&gt;
&lt;input name="title" type="hidden" value="Special:Search"/&gt;
&lt;input class="searchButton mw-fallbackSearchButton" id="mw-searchButton" name="fulltext" title="Search Wikipedia for this text" type="submit" value="Search" /&gt;
&lt;input class="searchButton" id="searchButton" name="go" title="Go to a page with this exact name if it exists" type="submit" value="Go" /&gt;
&lt;/input&gt;&lt;/div&gt;
&lt;/form&gt;
&lt;/div&gt;
&lt;/div&gt;
&lt;/div&gt;
&lt;div id="mw-panel"&gt;
&lt;div id="p-logo" role="banner"&gt;
&lt;a class="mw-wiki-logo" href="/wiki/Main_Page" title="Visit the main page"&gt;&lt;/a&gt;
&lt;/div&gt;
<!-- Please do not use role attribute as CSS selector, it is deprecated. --&gt;
&lt;nav aria-labelledby="p-navigation-label" class="mw-portlet mw-portlet-navigation vector-menu vector-menu-portal portal" id="p-navigation" role="navigation"&gt;
&lt;h3 id="p-navigation-label"&gt;
&lt;span&gt;Navigation&lt;/span&gt;
&lt;/h3&gt;</pre>
```

```
<div class="vector-menu-content">
<ul class="vector-menu-content-list"><li id="n-mainpage-description"><a accesskey="z" href="/wiki/Main_Page" title="Visit the main page [z]">Main page</a>
</li><li id="n-contents"><a href="/wiki/Wikipedia:Contents" title="Guides to browsing Wikipedia">Contents</a></li><li id="n-currentevents"><a href="/wiki/Portal:Current_events" title="Articles related to current events">Current events</a></li><li id="n-randompage"><a accesskey="x" href="/wiki/Special:Random" title="Visit a randomly selected article [x]">Random article</a></li><li id="n-aboutsite"><a href="/wiki/Wikipedia:About" title="Learn about Wikipedia and how it works">About Wikipedia</a></li><li id="n-contactpage"><a href="https://en.wikipedia.org/wiki/Wikipedia:Contact_us" title="How to contact Wikipedia">Contact us</a></li><li id="n-sitesupport"><a href="https://donate.wikimedia.org/w/index.php?title=Special:FundraiserRedirector&utm_source=donate&utm_medium=sidebar&utm_campaign=C13_en.wikipedia.org&useLang=en" title="Support us by donating to the Wikimedia Foundation">Donate</a></li></ul>
</div>
</nav>
<!-- Please do not use role attribute as CSS selector, it is deprecated. -->
<nav aria-labelledby="p-interaction-label" class="mw-portlet mw-portlet-interaction vector-menu vector-menu-portal portal" id="p-interaction" role="navigation">
<h3 id="p-interaction-label">
<span>Contribute</span>
</h3>
<div class="vector-menu-content">
<ul class="vector-menu-content-list"><li id="n-help"><a href="/wiki/Help:Contents" title="Guidance on how to use and edit Wikipedia">Help</a></li><li id="n-introduction"><a href="/wiki/Help:Introduction" title="Learn how to edit Wikipedia">Learn to edit</a></li><li id="n-portal"><a href="/wiki/Wikipedia:Community_portal" title="The hub for editors">Community portal</a></li><li id="n-recentchanges"><a accesskey="r" href="/wiki/Special:RecentChanges" title="A list of recent changes to Wikipedia [r]">Recent changes</a></li><li id="n-upload"><a href="/wiki/Wikipedia:File_Upload_Wizard" title="Add images or other media for use on Wikipedia">Upload file</a></li></ul>
</div>
</nav>
<!-- Please do not use role attribute as CSS selector, it is deprecated. -->
<nav aria-labelledby="p-tb-label" class="mw-portlet mw-portlet-tb vector-menu vector-menu-portal portal" id="p-tb" role="navigation">
<h3 id="p-tb-label">
<span>Tools</span>
</h3>
<div class="vector-menu-content">
<ul class="vector-menu-content-list"><li id="t-whatlinkshere"><a accesskey="j" href="/wiki/Special:WhatLinksHere>List_of_postal_codes_of_Canada:_M" title="List of all English Wikipedia pages containing links to this page [j]">What links here</a></li><li id="t-recentchangeslinked"><a accesskey="k" href="/wiki/Special:RecentChangesLinked>List_of_postal_codes_of_Canada:_M" rel="nofollow" title="Recent changes in pages linked from this page [k]">Related changes</a></li><li id="t-upload"><a accesskey="u" href="/wiki/Wikipedia:File_Upload_Wizard" title="Upload files [u]">Upload file</a></li><li id="t-specialpages"><a accesskey="q" href="/wiki/Special:SpecialPages" title="A list of all special pages [q]">Special pages</a></li><li id="t-permalink"><a href="/index.php?title=List_of_postal_codes_of_Canada:_M&oldid=1004901187" title="Permanent link to this revision of this page">Permanent link</a></li><li id="t-info"><a href="/index.php?title=List_of_postal_codes_of_Canada:_M&action=info" title="More information about this page">Page information</a></li></ul>
</div>
</nav>
```

```
id="t-cite"><a href="/w/index.php?title=Special:CiteThisPage&page=List_of
_postal_codes_of_Canada%3A_M&id=1004901187&wpFormIdentifier=titlefor
m" title="Information on how to cite this page">Cite this page</a></li><li id
="t-wikibase"><a accesskey="g" href="https://www.wikidata.org/wiki/Special:En
tityPage/Q3248240" title="Structured data on this page hosted by Wikidata
[g]">Wikidata item</a></li></ul>
</div>
</nav>
<!-- Please do not use role attribute as CSS selector, it is deprecated. -->
<nav aria-labelledby="p-coll-print_export-label" class="mw-portlet mw-portlet
-coll-print_export vector-menu vector-menu-portal portal" id="p-coll-print_ex
port" role="navigation">
<h3 id="p-coll-print_export-label">
<span>Print/export</span>
</h3>
<div class="vector-menu-content">
<ul class="vector-menu-content-list"><li id="coll-download-as-rl"><a href="/
w/index.php?title=Special:DownloadAsPdf&page=List_of_postal_codes_of_Cana
da%3A_M&action=show-download-screen" title="Download this page as a PDF f
ile">Download as PDF</a></li><li id="t-print"><a accesskey="p" href="/inde
x.php?title=List_of_postal_codes_of_Canada:_M&printable=yes" title="Print
able version of this page [p]">Printable version</a></li></ul>
</div>
</nav>
<!-- Please do not use role attribute as CSS selector, it is deprecated. -->
<nav aria-labelledby="p-lang-label" class="mw-portlet mw-portlet-lang vector-
menu vector-menu-portal portal" id="p-lang" role="navigation">
<h3 id="p-lang-label">
<span>Languages</span>
</h3>
<div class="vector-menu-content">
<ul class="vector-menu-content-list"><li class="interlanguage-link interwiki-
fr"><a class="interlanguage-link-target" href="https://fr.wikipedia.org/wiki/
Liste_des_codes_postaux_canadiens_d%C3%A9butant_par_M" hreflang="fr" lang="f
r" title="Liste des codes postaux canadiens débutant par M - French">Français
</a></li></ul>
<div class="after-portlet after-portlet-lang"><span class="wb-langlinks-edit
wb-langlinks-link"><a class="wbc-editpage" href="https://www.wikidata.org/wik
i/Special:EntityPage/Q3248240#sitelinks-wikipedia" title="Edit interlanguage
links">Edit links</a></span></div>
</div>
</nav>
</div>
</div>
<footer class="mw-footer" id="footer" role="contentinfo">
<ul id="footer-info">
<li id="footer-info-lastmod"> This page was last edited on 5 February 2021, a
t 00:18<span class="anonymous-show"> (UTC)</span>. </li>
<li id="footer-info-copyright">Text is available under the <a href="//en.wiki
pedia.org/wiki/Wikipedia:Text_of_Creative_Commons_Attribution-ShareAlike_3.0_
Unported_License" rel="license">Creative Commons Attribution-ShareAlike Licens
e</a><a href="//creativecommons.org/licenses/by-sa/3.0/" rel="license" style
="display:none;"></a>;
additional terms may apply. By using this site, you agree to the <a href="//
foundation.wikimedia.org/wiki/Terms_of_Use">Terms of Use</a> and <a href="//f
oundation.wikimedia.org/wiki/Privacy_policy">Privacy Policy</a>. Wikipedia® i
s a registered trademark of the <a href="//www.wikimediafoundation.org/">Wiki

https://labs.cognitiveclass.ai/tools/jupyterlab/lab/tree/labs/DV0101EN/DV0101EN-3-5-1-Generating-Maps-in-Python-py-v2.0.ipynb?lti=true



37/337


```

```
media Foundation, Inc.</a>, a non-profit organization.</li>
</ul>
<ul id="footer-places">
<li id="footer-places-privacy"><a class="extiw" href="https://foundation.wiki
media.org/wiki/Privacy_policy" title="wmf:Privacy policy">Privacy policy</a>
</li>
<li id="footer-places-about"><a href="/wiki/Wikipedia:About" title="Wikipedi
a:About">About Wikipedia</a></li>
<li id="footer-places-disclaimer"><a href="/wiki/Wikipedia:General_disclaime
r" title="Wikipedia:General disclaimer">Disclaimers</a></li>
<li id="footer-places-contact"><a href="//en.wikipedia.org/wiki/Wikipedia:Con
tact_us">Contact Wikipedia</a></li>
<li id="footer-places-mobileview"><a class="noprint stopMobileRedirectToggle"
href="//en.m.wikipedia.org/w/index.php?title=List_of_postal_codes_of_Canada:_M&mobileaction=toggle_view_mobile">Mobile view</a></li>
<li id="footer-places-developers"><a href="https://www.mediawiki.org/wiki/Spe
cial:MyLanguage/How_to_contribute">Developers</a></li>
<li id="footer-places-statslink"><a href="https://stats.wikimedia.org/#/en.wi
kipedia.org">Statistics</a></li>
<li id="footer-places-cookiestatement"><a href="https://foundation.wikimedia.
org/wiki/Cookie_statement">Cookie statement</a></li>
</ul>
<ul class="noprint" id="footer-icons">
<li id="footer-copyrightico"><a href="https://wikimediafoundation.org/"></a>
</li>
<li id="footer-poweredbyico"><a href="https://www.mediawiki.org/"></a></li>
</ul>
<div style="clear: both;"></div>
</footer>
<script>(RLQ=window.RLQ||[]).push(function(){mw.config.set({"wgPageParseReport": {"limitreport": {"cputime": "0.200", "walltime": "0.243", "ppvisitednodes": {"value": 279, "limit": 1000000}, "postexpandincludesize": {"value": 10828, "limit": 2097152}, "templateargumentsize": {"value": 275, "limit": 2097152}, "expansiondepth": {"value": 8, "limit": 40}, "expensivefunctioncount": {"value": 0, "limit": 500}, "unstrip-depth": {"value": 1, "limit": 20}, "unstrip-size": {"value": 9406, "limit": 5000000}, "entityaccesscount": {"value": 0, "limit": 400}, "timingprofile": ["100.00% 20 2.632 1 -total", " 54.62% 110.679 3 Template:Cite_web", " 38.51% 7 8.040 1 Template:Short_description", " 22.61% 45.825 1 Template:Pa getype", " 9.48% 19.206 2 Template:Main_other", " 7.30% 14.787 1 Template:SDcat", " 1.78% 3.607 1 Template:Canadian_postal_code s"]}, "scribunto": {"limitreport-timeusage": {"value": "0.101", "limit": "10.000"}, "limitreport-memusage": {"value": 3013916, "limit": 52428800}, "cachereport": {"origin": "mw1358", "timestamp": "20210205001814", "ttl": 2592000, "transientconte nt": false}}});</script>
<script type="application/ld+json">{@context": "https://schema.org", "@typ e": "Article", "name": "List of postal codes of Canada: M", "url": "https://en.w ikipedia.org/w/index.php?title=List_of_postal_codes_of_Canada:_M", "sameAs": "http://www.wikidata.org/entity/Q3248240", "mainEntity": "http://www.wikidata.org/entity/Q3248240", "author": {"@type": "Organization", "name": "Contributors to Wiki media projects"}, "publisher": {"@type": "Organization", "name": "Wikimedia Founda
```

```
tion, Inc.", "logo": {"@type": "ImageObject", "url": "https://www.wikimedia.org\n/static/images/wmf-hor-googpub.png"}, "datePublished": "2004-03-20T10:02:13\nZ", "dateModified": "2021-02-05T00:18:07Z", "headline": "Wikimedia list article"}\n</script>\n<script>(RLQ=window.RLQ||[]).push(function(){mw.config.set({"wgBackendRespon\neTime":139,"wgHostname":"mw1267"});});</script>\n</body></html>
```

~~#Notebook_part1~~

Conversion of HTML data, text data to tabular format

In [107]:

```
Toronto_table=soup.find('table',{'class':'wikitable'})
```

In [108]:

```
toronto_df=pd.read_html(str(Toronto_table))
# convert list to dataframe
toronto_df=pd.DataFrame(toronto_df[0])
toronto_df
```

Out[108]:

Postal Code	Borough	Neighbourhood
0	M1A	Not assigned
1	M2A	Not assigned
2	M3A	North York
3	M4A	North York
4	M5A	Downtown Toronto
5	M6A	North York
6	M7A	Downtown Toronto
7	M8A	Not assigned
8	M9A	Etobicoke
9	M1B	Scarborough
10	M2B	Not assigned
11	M3B	North York
12	M4B	East York
13	M5B	Downtown Toronto
14	M6B	North York
15	M7B	Not assigned
16	M8B	Not assigned
17	M9B	Etobicoke
18	M1C	Scarborough
19	M2C	Not assigned
20	M3C	North York
21	M4C	East York
22	M5C	Downtown Toronto
23	M6C	York
24	M7C	Not assigned
25	M8C	Not assigned
26	M9C	Etobicoke
27	M1E	Scarborough
28	M2E	Not assigned
29	M3E	Not assigned
30	M4E	East Toronto
31	M5E	Downtown Toronto
32	M6E	York
33	M7E	Not assigned

Postal Code	Borough	Neighbourhood
34	M8E	Not assigned
35	M9E	Not assigned
36	M1G	Scarborough
37	M2G	Not assigned
38	M3G	Not assigned
39	M4G	East York
40	M5G	Downtown Toronto
41	M6G	Downtown Toronto
42	M7G	Not assigned
43	M8G	Not assigned
44	M9G	Not assigned
45	M1H	Scarborough
46	M2H	North York
47	M3H	North York
		Bathurst Manor, Wilson Heights, Downsview North
48	M4H	East York
49	M5H	Downtown Toronto
		Richmond, Adelaide, King
50	M6H	West Toronto
		Dufferin, Dovercourt Village
51	M7H	Not assigned
52	M8H	Not assigned
53	M9H	Not assigned
54	M1J	Scarborough
		Scarborough Village
55	M2J	North York
		Fairview, Henry Farm, Oriole
56	M3J	North York
		Northwood Park, York University
57	M4J	East York
		East Toronto, Broadview North (Old East York)
58	M5J	Downtown Toronto
		Harbourfront East, Union Station, Toronto Islands
59	M6J	West Toronto
		Little Portugal, Trinity
60	M7J	Not assigned
61	M8J	Not assigned
62	M9J	Not assigned
63	M1K	Scarborough
		Kennedy Park, Ionview, East Birchmount Park
64	M2K	North York
		Bayview Village
65	M3K	North York
		Downsview
66	M4K	East Toronto
		The Danforth West, Riverdale
67	M5K	Downtown Toronto
		Toronto Dominion Centre, Design Exchange
68	M6K	West Toronto
		Brockton, Parkdale Village, Exhibition Place
69	M7K	Not assigned

Postal Code	Borough	Neighbourhood
70	M8K	Not assigned
71	M9K	Not assigned
72	M1L	Scarborough
73	M2L	North York
74	M3L	North York
75	M4L	East Toronto
76	M5L	Downtown Toronto
77	M6L	North York
78	M7L	Not assigned
79	M8L	Not assigned
80	M9L	North York
81	M1M	Scarborough
82	M2M	North York
83	M3M	North York
84	M4M	East Toronto
85	M5M	North York
86	M6M	York
87	M7M	Not assigned
88	M8M	Not assigned
89	M9M	North York
90	M1N	Scarborough
91	M2N	North York
92	M3N	North York
93	M4N	Central Toronto
94	M5N	Central Toronto
95	M6N	York
96	M7N	Not assigned
97	M8N	Not assigned
98	M9N	York
99	M1P	Scarborough
100	M2P	North York
101	M3P	Not assigned
102	M4P	Central Toronto
103	M5P	Central Toronto
104	M6P	West Toronto
105	M7P	Not assigned

Postal Code	Borough	Neighbourhood
106	M8P	Not assigned
107	M9P	Etobicoke
108	M1R	Scarborough
109	M2R	North York
110	M3R	Not assigned
111	M4R	Central Toronto
112	M5R	Central Toronto
113	M6R	West Toronto
114	M7R	Mississauga
115	M8R	Not assigned
116	M9R	Etobicoke
		Kingsview Village, St. Phillips, Martin Grove ...
117	M1S	Scarborough
118	M2S	Not assigned
119	M3S	Not assigned
120	M4S	Central Toronto
		Davisville
121	M5S	Downtown Toronto
		University of Toronto, Harbord
122	M6S	West Toronto
		Runnymede, Swansea
123	M7S	Not assigned
124	M8S	Not assigned
125	M9S	Not assigned
126	M1T	Scarborough
		Clarks Corners, Tam O'Shanter, Sullivan
127	M2T	Not assigned
128	M3T	Not assigned
129	M4T	Central Toronto
		Moore Park, Summerhill East
130	M5T	Downtown Toronto
		Kensington Market, Chinatown, Grange Park
131	M6T	Not assigned
132	M7T	Not assigned
133	M8T	Not assigned
134	M9T	Not assigned
135	M1V	Scarborough
		Milliken, Agincourt North, Steeles East, L'Amo...
136	M2V	Not assigned
137	M3V	Not assigned
138	M4V	Central Toronto
		Summerhill West, Rathnelly, South Hill, Forest...
139	M5V	Downtown Toronto
		CN Tower, King and Spadina, Railway Lands, Har...
140	M6V	Not assigned
141	M7V	Not assigned

Postal Code	Borough	Neighbourhood
142	M8V	Etobicoke New Toronto, Mimico South, Humber Bay Shores
143	M9V	Etobicoke South Steeles, Silverstone, Humbergate, Jamest...
144	M1W	Scarborough Steeles West, L'Amoreaux West
145	M2W	Not assigned Not assigned
146	M3W	Not assigned Not assigned
147	M4W	Downtown Toronto Rosedale
148	M5W	Downtown Toronto Stn A PO Boxes
149	M6W	Not assigned Not assigned
150	M7W	Not assigned Not assigned
151	M8W	Etobicoke Alderwood, Long Branch
152	M9W	Etobicoke Northwest, West Humber - Clairville
153	M1X	Scarborough Upper Rouge
154	M2X	Not assigned Not assigned
155	M3X	Not assigned Not assigned
156	M4X	Downtown Toronto St. James Town, Cabbagetown
157	M5X	Downtown Toronto First Canadian Place, Underground city
158	M6X	Not assigned Not assigned
159	M7X	Not assigned Not assigned
160	M8X	Etobicoke The Kingsway, Montgomery Road, Old Mill North
161	M9X	Not assigned Not assigned
162	M1Y	Not assigned Not assigned
163	M2Y	Not assigned Not assigned
164	M3Y	Not assigned Not assigned
165	M4Y	Downtown Toronto Church and Wellesley
166	M5Y	Not assigned Not assigned
167	M6Y	Not assigned Not assigned
168	M7Y	East Toronto Business reply mail Processing Centre, South C...
169	M8Y	Etobicoke Old Mill South, King's Mill Park, Sunnylea, Hu...
170	M9Y	Not assigned Not assigned
171	M1Z	Not assigned Not assigned
172	M2Z	Not assigned Not assigned
173	M3Z	Not assigned Not assigned
174	M4Z	Not assigned Not assigned
175	M5Z	Not assigned Not assigned
176	M6Z	Not assigned Not assigned
177	M7Z	Not assigned Not assigned

Postal Code	Borough	Neighbourhood
178	M8Z	Etobicoke Mimico NW, The Queensway West, South of Bloor,...
179	M9Z	Not assigned

In [109]:

```
print(toronto_df)
```

	Postal Code	Borough
0	M1A	Not assigned
1	M2A	Not assigned
2	M3A	North York
3	M4A	North York
4	M5A	Downtown Toronto
5	M6A	North York
6	M7A	Downtown Toronto
7	M8A	Not assigned
8	M9A	Etobicoke
9	M1B	Scarborough
10	M2B	Not assigned
11	M3B	North York
12	M4B	East York
13	M5B	Downtown Toronto
14	M6B	North York
15	M7B	Not assigned
16	M8B	Not assigned
17	M9B	Etobicoke
18	M1C	Scarborough
19	M2C	Not assigned
20	M3C	North York
21	M4C	East York
22	M5C	Downtown Toronto
23	M6C	York
24	M7C	Not assigned
25	M8C	Not assigned
26	M9C	Etobicoke
27	M1E	Scarborough
28	M2E	Not assigned
29	M3E	Not assigned
30	M4E	East Toronto
31	M5E	Downtown Toronto
32	M6E	York
33	M7E	Not assigned
34	M8E	Not assigned
35	M9E	Not assigned
36	M1G	Scarborough
37	M2G	Not assigned
38	M3G	Not assigned
39	M4G	East York
40	M5G	Downtown Toronto
41	M6G	Downtown Toronto
42	M7G	Not assigned
43	M8G	Not assigned
44	M9G	Not assigned
45	M1H	Scarborough
46	M2H	North York
47	M3H	North York
48	M4H	East York
49	M5H	Downtown Toronto
50	M6H	West Toronto
51	M7H	Not assigned
52	M8H	Not assigned
53	M9H	Not assigned
54	M1J	Scarborough
55	M2J	North York

56 M3J North York
57 M4J East York
58 M5J Downtown Toronto
59 M6J West Toronto
60 M7J Not assigned
61 M8J Not assigned
62 M9J Not assigned
63 M1K Scarborough
64 M2K North York
65 M3K North York
66 M4K East Toronto
67 M5K Downtown Toronto
68 M6K West Toronto
69 M7K Not assigned
70 M8K Not assigned
71 M9K Not assigned
72 M1L Scarborough
73 M2L North York
74 M3L North York
75 M4L East Toronto
76 M5L Downtown Toronto
77 M6L North York
78 M7L Not assigned
79 M8L Not assigned
80 M9L North York
81 M1M Scarborough
82 M2M North York
83 M3M North York
84 M4M East Toronto
85 M5M North York
86 M6M York
87 M7M Not assigned
88 M8M Not assigned
89 M9M North York
90 M1N Scarborough
91 M2N North York
92 M3N North York
93 M4N Central Toronto
94 M5N Central Toronto
95 M6N York
96 M7N Not assigned
97 M8N Not assigned
98 M9N York
99 M1P Scarborough
100 M2P North York
101 M3P Not assigned
102 M4P Central Toronto
103 M5P Central Toronto
104 M6P West Toronto
105 M7P Not assigned
106 M8P Not assigned
107 M9P Etobicoke
108 M1R Scarborough
109 M2R North York
110 M3R Not assigned
111 M4R Central Toronto
112 M5R Central Toronto

113	M6R	West Toronto
114	M7R	Mississauga
115	M8R	Not assigned
116	M9R	Etobicoke
117	M1S	Scarborough
118	M2S	Not assigned
119	M3S	Not assigned
120	M4S	Central Toronto
121	M5S	Downtown Toronto
122	M6S	West Toronto
123	M7S	Not assigned
124	M8S	Not assigned
125	M9S	Not assigned
126	M1T	Scarborough
127	M2T	Not assigned
128	M3T	Not assigned
129	M4T	Central Toronto
130	M5T	Downtown Toronto
131	M6T	Not assigned
132	M7T	Not assigned
133	M8T	Not assigned
134	M9T	Not assigned
135	M1V	Scarborough
136	M2V	Not assigned
137	M3V	Not assigned
138	M4V	Central Toronto
139	M5V	Downtown Toronto
140	M6V	Not assigned
141	M7V	Not assigned
142	M8V	Etobicoke
143	M9V	Etobicoke
144	M1W	Scarborough
145	M2W	Not assigned
146	M3W	Not assigned
147	M4W	Downtown Toronto
148	M5W	Downtown Toronto
149	M6W	Not assigned
150	M7W	Not assigned
151	M8W	Etobicoke
152	M9W	Etobicoke
153	M1X	Scarborough
154	M2X	Not assigned
155	M3X	Not assigned
156	M4X	Downtown Toronto
157	M5X	Downtown Toronto
158	M6X	Not assigned
159	M7X	Not assigned
160	M8X	Etobicoke
161	M9X	Not assigned
162	M1Y	Not assigned
163	M2Y	Not assigned
164	M3Y	Not assigned
165	M4Y	Downtown Toronto
166	M5Y	Not assigned
167	M6Y	Not assigned
168	M7Y	East Toronto
169	M8Y	Etobicoke

170	M9Y	Not assigned
171	M1Z	Not assigned
172	M2Z	Not assigned
173	M3Z	Not assigned
174	M4Z	Not assigned
175	M5Z	Not assigned
176	M6Z	Not assigned
177	M7Z	Not assigned
178	M8Z	Etobicoke
179	M9Z	Not assigned

		Neighbourhood
0		Not assigned
1		Not assigned
2		Parkwoods
3		Victoria Village
4		Regent Park, Harbourfront
5		Lawrence Manor, Lawrence Heights
6		Queen's Park, Ontario Provincial Government
7		Not assigned
8		Islington Avenue, Humber Valley Village
9		Malvern, Rouge
10		Not assigned
11		Don Mills
12		Parkview Hill, Woodbine Gardens
13		Garden District, Ryerson
14		Glencairn
15		Not assigned
16		Not assigned
17	West Deane Park, Princess Gardens, Martin Grov...	
18	Rouge Hill, Port Union, Highland Creek	
19		Not assigned
20		Don Mills
21		Woodbine Heights
22		St. James Town
23		Humewood-Cedarvale
24		Not assigned
25		Not assigned
26	Eringate, Bloordale Gardens, Old Burnhamthorpe...	
27	Guildwood, Morningside, West Hill	
28		Not assigned
29		Not assigned
30		The Beaches
31		Berczy Park
32		Caledonia-Fairbanks
33		Not assigned
34		Not assigned
35		Not assigned
36		Woburn
37		Not assigned
38		Not assigned
39		Leaside
40		Central Bay Street
41		Christie
42		Not assigned
43		Not assigned
44		Not assigned

45 Cedarbrae
46 Hillcrest Village
47 Bathurst Manor, Wilson Heights, Downsview North
48 Thorncliffe Park
49 Richmond, Adelaide, King
50 Dufferin, Dovercourt Village
51 Not assigned
52 Not assigned
53 Not assigned
54 Scarborough Village
55 Fairview, Henry Farm, Oriole
56 Northwood Park, York University
57 East Toronto, Broadview North (Old East York)
58 Harbourfront East, Union Station, Toronto Islands
59 Little Portugal, Trinity
60 Not assigned
61 Not assigned
62 Not assigned
63 Kennedy Park, Ionview, East Birchmount Park
64 Bayview Village
65 Downsview
66 The Danforth West, Riverdale
67 Toronto Dominion Centre, Design Exchange
68 Brockton, Parkdale Village, Exhibition Place
69 Not assigned
70 Not assigned
71 Not assigned
72 Golden Mile, Clairlea, Oakridge
73 York Mills, Silver Hills
74 Downsview
75 India Bazaar, The Beaches West
76 Commerce Court, Victoria Hotel
77 North Park, Maple Leaf Park, Upwood Park
78 Not assigned
79 Not assigned
80 Humber Summit
81 Cliffside, Cliffcrest, Scarborough Village West
82 Willowdale, Newtonbrook
83 Downsview
84 Studio District
85 Bedford Park, Lawrence Manor East
86 Del Ray, Mount Dennis, Keelsdale and Silverthorn
87 Not assigned
88 Not assigned
89 Humberlea, Emery
90 Birch Cliff, Cliffside West
91 Willowdale, Willowdale East
92 Downsview
93 Lawrence Park
94 Roselawn
95 Runnymede, The Junction North
96 Not assigned
97 Not assigned
98 Weston
99 Dorset Park, Wexford Heights, Scarborough Town...
100 York Mills West
101 Not assigned

102 Davisville North
103 Forest Hill North & West, Forest Hill Road Park
104 High Park, The Junction South
105 Not assigned
106 Not assigned
107 Westmount
108 Wexford, Maryvale
109 Willowdale, Willowdale West
110 Not assigned
111 North Toronto West, Lawrence Park
112 The Annex, North Midtown, Yorkville
113 Parkdale, Roncesvalles
114 Canada Post Gateway Processing Centre
115 Not assigned
116 Kingsview Village, St. Phillips, Martin Grove ...
117 Agincourt
118 Not assigned
119 Not assigned
120 Davisville
121 University of Toronto, Harbord
122 Runnymede, Swansea
123 Not assigned
124 Not assigned
125 Not assigned
126 Clarks Corners, Tam O'Shanter, Sullivan
127 Not assigned
128 Not assigned
129 Moore Park, Summerhill East
130 Kensington Market, Chinatown, Grange Park
131 Not assigned
132 Not assigned
133 Not assigned
134 Not assigned
135 Milliken, Agincourt North, Steeles East, L'Amo...
136 Not assigned
137 Not assigned
138 Summerhill West, Rathnelly, South Hill, Forest...
139 CN Tower, King and Spadina, Railway Lands, Har...
140 Not assigned
141 Not assigned
142 New Toronto, Mimico South, Humber Bay Shores
143 South Steeles, Silverstone, Humbergate, Jamest...
144 Steeles West, L'Amoreaux West
145 Not assigned
146 Not assigned
147 Rosedale
148 Stn A PO Boxes
149 Not assigned
150 Not assigned
151 Alderwood, Long Branch
152 Northwest, West Humber - Clairville
153 Upper Rouge
154 Not assigned
155 Not assigned
156 St. James Town, Cabbagetown
157 First Canadian Place, Underground city
158 Not assigned

159		Not assigned
160	The Kingsway, Montgomery Road, Old Mill North	
161		Not assigned
162		Not assigned
163		Not assigned
164		Not assigned
165		Church and Wellesley
166		Not assigned
167		Not assigned
168	Business reply mail Processing Centre, South C...	
169	Old Mill South, King's Mill Park, Sunnylea, Hu...	
170		Not assigned
171		Not assigned
172		Not assigned
173		Not assigned
174		Not assigned
175		Not assigned
176		Not assigned
177		Not assigned
178	Mimico NW, The Queensway West, South of Bloor,....	
179		Not assigned

In [110]:

```
toronto_df.value_counts()
```

Out[110]:

Postal Code	Borough	Neighbourhood
M9Z	Not assigned	Not assigned
1		
M4J	East York	East Toronto, Broadview North (Old East York)
1		
M4G	East York	Leaside
1		
M4E	East Toronto	The Beaches
1		
M4C	East York	Woodbine Heights
1		
M4B	East York	Parkview Hill, Woodbine Gardens
1		
M4A	North York	Victoria Village
1		
M3Z	Not assigned	Not assigned
1		
M3Y	Not assigned	Not assigned
1		
M3X	Not assigned	Not assigned
1		
M3W	Not assigned	Not assigned
1		
M3V	Not assigned	Not assigned
1		
M3T	Not assigned	Not assigned
1		
M3S	Not assigned	Not assigned
1		
M3R	Not assigned	Not assigned
1		
M3P	Not assigned	Not assigned
1		
M3N	North York	Downsvie
1		
M3M	North York	Downsvie
1		
M3L	North York	Downsvie
1		
M3K	North York	Downsvie
1		
M3J	North York	Northwood Park, York University
1		
M4H	East York	Thorncliffe Park
1		
M4K	East Toronto	The Danforth West, Riverdale
1		
M9Y	Not assigned	Not assigned
1		
M4L	East Toronto	India Bazaar, The Beaches West
1		
M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange
1		
M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islan
nds		

1
M5H Downtown Toronto Richmond, Adelaide, King
1
M5G Downtown Toronto Central Bay Street
1
M5E Downtown Toronto Berczy Park
1
M5C Downtown Toronto St. James Town
1
M5B Downtown Toronto Garden District, Ryerson
1
M5A Downtown Toronto Regent Park, Harbourfront
1
M4Z Not assigned Not assigned
1
M4Y Downtown Toronto Church and Wellesley
1
M4X Downtown Toronto St. James Town, Cabbagetown
1
M4W Downtown Toronto Rosedale
1
M4V Central Toronto Summerhill West, Rathnelly, South Hill, Forest
Hill SE, Deer Park
1
M4T Central Toronto Moore Park, Summerhill East
1
M4S Central Toronto Davisville
1
M4R Central Toronto North Toronto West, Lawrence Park
1
M4P Central Toronto Davisville North
1
M4N Central Toronto Lawrence Park
1
M4M East Toronto Studio District
1
M3H North York Bathurst Manor, Wilson Heights, Downsview Nort
h
1
M3G Not assigned Not assigned
1
M3E Not assigned Not assigned
1
M3C North York Don Mills
1
M1Z Not assigned Not assigned
1
M1Y Not assigned Not assigned
1
M1X Scarborough Upper Rouge
1
M1W Scarborough Steeles West, L'Amoreaux West
1
M1V Scarborough Milliken, Agincourt North, Steeles East, L'Amo
reaux East
1
M1T Scarborough Clarks Corners, Tam O'Shanter, Sullivan

1		
M1S	Scarborough	Agincourt
1		
M1R	Scarborough	Wexford, Maryvale
1		
M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town
Centre		
1		
M1N	Scarborough	Birch Cliff, Cliffside West
1		
M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village Wes
t		
1		
M1L	Scarborough	Golden Mile, Clairlea, Oakridge
1		
M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park
1		
M1J	Scarborough	Scarborough Village
1		
M1H	Scarborough	Cedarbrae
1		
M1G	Scarborough	Woburn
1		
M1E	Scarborough	Guildwood, Morningside, West Hill
1		
M1C	Scarborough	Rouge Hill, Port Union, Highland Creek
1		
M1B	Scarborough	Malvern, Rouge
1		
M2A	Not assigned	Not assigned
1		
M2B	Not assigned	Not assigned
1		
M2C	Not assigned	Not assigned
1		
M2S	Not assigned	Not assigned
1		
M3B	North York	Don Mills
1		
M3A	North York	Parkwoods
1		
M2Z	Not assigned	Not assigned
1		
M2Y	Not assigned	Not assigned
1		
M2X	Not assigned	Not assigned
1		
M2W	Not assigned	Not assigned
1		
M2V	Not assigned	Not assigned
1		
M2T	Not assigned	Not assigned
1		
M2R	North York	Willowdale, Willowdale West
1		
M2E	Not assigned	Not assigned
1		

M2P 1	North York	York Mills West
M2N 1	North York	Willowdale, Willowdale East
M2M 1	North York	Willowdale, Newtonbrook
M2L 1	North York	York Mills, Silver Hills
M2K 1	North York	Bayview Village
M2J 1	North York	Fairview, Henry Farm, Oriole
M2H 1	North York	Hillcrest Village
M2G 1	Not assigned	Not assigned
M5L 1	Downtown Toronto	Commerce Court, Victoria Hotel
M5M 1	North York	Bedford Park, Lawrence Manor East
M5N 1	Central Toronto	Roselawn
M7V 1	Not assigned	Not assigned
M8V 1	Etobicoke	New Toronto, Mimico South, Humber Bay Shores
M8T 1	Not assigned	Not assigned
M8S 1	Not assigned	Not assigned
M8R 1	Not assigned	Not assigned
M8P 1	Not assigned	Not assigned
M8N 1	Not assigned	Not assigned
M8M 1	Not assigned	Not assigned
M8L 1	Not assigned	Not assigned
M8K 1	Not assigned	Not assigned
M8J 1	Not assigned	Not assigned
M8H 1	Not assigned	Not assigned
M8G 1	Not assigned	Not assigned
M8E 1	Not assigned	Not assigned
M8C 1	Not assigned	Not assigned
M8B 1	Not assigned	Not assigned
M8A 1	Not assigned	Not assigned
M7Z	Not assigned	Not assigned

1
M7Y East Toronto Business reply mail Processing Centre, South C
entral Letter Processing Plant Toronto

1
M7X Not assigned Not assigned

1
M8W Etobicoke Alderwood, Long Branch

1
M8X Etobicoke The Kingsway, Montgomery Road, Old Mill North

1
M8Y Etobicoke Old Mill South, King's Mill Park, Sunnylea, Hu
mber Bay, Mimico NE, The Queensway East, Royal York South East, Kingsway Park
South East 1

M9M North York Humberlea, Emery

1
M9X Not assigned Not assigned

1
M9W Etobicoke Northwest, West Humber - Clairville

1
M9V Etobicoke South Steeles, Silverstone, Humbergate, Jamest
own, Mount Olive, Beaumont Heights, Thistletown, Albion Gardens

1
M9T Not assigned Not assigned

1
M9S Not assigned Not assigned

1
M9R Etobicoke Kingsview Village, St. Phillips, Martin Grove
Gardens, Richview Gardens

1
M9P Etobicoke Westmount

1
M9N York Weston

1
M9L North York Humber Summit

1
M8Z Etobicoke Mimico NW, The Queensway West, South of Bloor,
Kingsway Park South West, Royal York South West

1
M9K Not assigned Not assigned

1
M9J Not assigned Not assigned

1
M9H Not assigned Not assigned

1
M9G Not assigned Not assigned

1
M9E Not assigned Not assigned

1
M9C Etobicoke Eringate, Bloordale Gardens, Old Burnhamthorp
e, Markland Wood

1
M9B Etobicoke West Deane Park, Princess Gardens, Martin Grov
e, Islington, Cloverdale

1
M9A Etobicoke Islington Avenue, Humber Valley Village

1
M7W Not assigned Not assigned

1		
M7T	Not assigned	Not assigned
1		
M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Par
k		
1		
M7S	Not assigned	Not assigned
1		
M6N	York	Runnymede, The Junction North
1		
M6M	York	Del Ray, Mount Dennis, Keelsdale and Silvertho
rn		
1		
M6L	North York	North Park, Maple Leaf Park, Upwood Park
1		
M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place
1		
M6J	West Toronto	Little Portugal, Trinity
1		
M6H	West Toronto	Dufferin, Dovercourt Village
1		
M6G	Downtown Toronto	Christie
1		
M6E	York	Caledonia-Fairbanks
1		
M6C	York	Humewood-Cedarvale
1		
M6B	North York	Glencairn
1		
M6A	North York	Lawrence Manor, Lawrence Heights
1		
M5Z	Not assigned	Not assigned
1		
M5Y	Not assigned	Not assigned
1		
M5X	Downtown Toronto	First Canadian Place, Underground city
1		
M5W	Downtown Toronto	Stn A PO Boxes
1		
M5V	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har
bourfront West, Bathurst Quay, South Niagara, Island airport		
1		
M5T	Downtown Toronto	Kensington Market, Chinatown, Grange Park
1		
M5S	Downtown Toronto	University of Toronto, Harbord
1		
M5R	Central Toronto	The Annex, North Midtown, Yorkville
1		
M6P	West Toronto	High Park, The Junction South
1		
M6R	West Toronto	Parkdale, Roncesvalles
1		
M6S	West Toronto	Runnymede, Swansea
1		
M7G	Not assigned	Not assigned
1		
M7R	Mississauga	Canada Post Gateway Processing Centre

```
1  
M7P      Not assigned    Not assigned  
1  
M7N      Not assigned    Not assigned  
1  
M7M      Not assigned    Not assigned  
1  
M7L      Not assigned    Not assigned  
1  
M7K      Not assigned    Not assigned  
1  
M7J      Not assigned    Not assigned  
1  
M7H      Not assigned    Not assigned  
1  
M7E      Not assigned    Not assigned  
1  
M6T      Not assigned    Not assigned  
1  
M7C      Not assigned    Not assigned  
1  
M7B      Not assigned    Not assigned  
1  
M7A      Downtown Toronto Queen's Park, Ontario Provincial Government  
1  
M6Z      Not assigned    Not assigned  
1  
M6Y      Not assigned    Not assigned  
1  
M6X      Not assigned    Not assigned  
1  
M6W      Not assigned    Not assigned  
1  
M6V      Not assigned    Not assigned  
1  
M1A      Not assigned    Not assigned  
1  
dtype: int64
```

Removing 'Not Assigned' data to create clean tabular data

In [111]:

```
#dropping rows of 'Not Assigned' in 'Borough' and 'Neighbourhood' column
toronto_clean = toronto_df[toronto_df.Borough != 'Not assigned'].reset_index(drop=True)
toronto_clean
```

Out[111]:

Postal Code	Borough	Neighbourhood
0	M3A	North York
1	M4A	North York
2	M5A	Downtown Toronto
3	M6A	North York
4	M7A	Downtown Toronto
5	M9A	Etobicoke
6	M1B	Scarborough
7	M3B	North York
8	M4B	East York
9	M5B	Downtown Toronto
10	M6B	North York
11	M9B	Etobicoke
12	M1C	Scarborough
13	M3C	North York
14	M4C	East York
15	M5C	Downtown Toronto
16	M6C	York
17	M9C	Etobicoke
18	M1E	Scarborough
19	M4E	East Toronto
20	M5E	Downtown Toronto
21	M6E	York
22	M1G	Scarborough
23	M4G	East York
24	M5G	Downtown Toronto
25	M6G	Downtown Toronto
26	M1H	Scarborough
27	M2H	North York
28	M3H	North York
29	M4H	East York
30	M5H	Downtown Toronto
31	M6H	West Toronto
32	M1J	Scarborough
33	M2J	North York
		Parkwoods
		Victoria Village
		Regent Park, Harbourfront
		Lawrence Manor, Lawrence Heights
		Queen's Park, Ontario Provincial Government
		Islington Avenue, Humber Valley Village
		Malvern, Rouge
		Don Mills
		Parkview Hill, Woodbine Gardens
		Garden District, Ryerson
		Glencairn
		West Deane Park, Princess Gardens, Martin Grov...
		Rouge Hill, Port Union, Highland Creek
		Don Mills
		Woodbine Heights
		St. James Town
		Humewood-Cedarvale
		Eringate, Bloordale Gardens, Old Burnhamthorpe...
		Guildwood, Morningside, West Hill
		The Beaches
		Berczy Park
		Caledonia-Fairbanks
		Woburn
		Leaside
		Central Bay Street
		Christie
		Cedarbrae
		Hillcrest Village
		Bathurst Manor, Wilson Heights, Downsview North
		Thorncliffe Park
		Richmond, Adelaide, King
		Dufferin, Dovercourt Village
		Scarborough Village
		Fairview, Henry Farm, Oriole

Postal Code	Borough	Neighbourhood	
34	M3J	North York	Northwood Park, York University
35	M4J	East York	East Toronto, Broadview North (Old East York)
36	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands
37	M6J	West Toronto	Little Portugal, Trinity
38	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park
39	M2K	North York	Bayview Village
40	M3K	North York	Downsview
41	M4K	East Toronto	The Danforth West, Riverdale
42	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange
43	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place
44	M1L	Scarborough	Golden Mile, Clairlea, Oakridge
45	M2L	North York	York Mills, Silver Hills
46	M3L	North York	Downsview
47	M4L	East Toronto	India Bazaar, The Beaches West
48	M5L	Downtown Toronto	Commerce Court, Victoria Hotel
49	M6L	North York	North Park, Maple Leaf Park, Upwood Park
50	M9L	North York	Humber Summit
51	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West
52	M2M	North York	Willowdale, Newtonbrook
53	M3M	North York	Downsview
54	M4M	East Toronto	Studio District
55	M5M	North York	Bedford Park, Lawrence Manor East
56	M6M	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn
57	M9M	North York	Humberlea, Emery
58	M1N	Scarborough	Birch Cliff, Cliffside West
59	M2N	North York	Willowdale, Willowdale East
60	M3N	North York	Downsview
61	M4N	Central Toronto	Lawrence Park
62	M5N	Central Toronto	Roselawn
63	M6N	York	Runnymede, The Junction North
64	M9N	York	Weston
65	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...
66	M2P	North York	York Mills West
67	M4P	Central Toronto	Davisville North
68	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park
69	M6P	West Toronto	High Park, The Junction South

Postal Code	Borough	Neighbourhood
70	M9P	Etobicoke
71	M1R	Scarborough
72	M2R	North York
73	M4R	Central Toronto
74	M5R	Central Toronto
75	M6R	West Toronto
76	M7R	Mississauga
77	M9R	Etobicoke
78	M1S	Scarborough
79	M4S	Central Toronto
80	M5S	Downtown Toronto
81	M6S	West Toronto
82	M1T	Scarborough
83	M4T	Central Toronto
84	M5T	Downtown Toronto
85	M1V	Scarborough
86	M4V	Central Toronto
87	M5V	Downtown Toronto
88	M8V	Etobicoke
89	M9V	Etobicoke
90	M1W	Scarborough
91	M4W	Downtown Toronto
92	M5W	Downtown Toronto
93	M8W	Etobicoke
94	M9W	Etobicoke
95	M1X	Scarborough
96	M4X	Downtown Toronto
97	M5X	Downtown Toronto
98	M8X	Etobicoke
99	M4Y	Downtown Toronto
100	M7Y	East Toronto
101	M8Y	Etobicoke
102	M8Z	Etobicoke
		Mimico NW, The Queensway West, South of Bloor,...
		Westmount
		Wexford, Maryvale
		Willowdale, Willowdale West
		North Toronto West, Lawrence Park
		The Annex, North Midtown, Yorkville
		Parkdale, Roncesvalles
		Canada Post Gateway Processing Centre
		Kingsview Village, St. Phillips, Martin Grove ...
		Agincourt
		Davisville
		University of Toronto, Harbord
		Runnymede, Swansea
		Clarks Corners, Tam O'Shanter, Sullivan
		Moore Park, Summerhill East
		Kensington Market, Chinatown, Grange Park
		Milliken, Agincourt North, Steeles East, L'Amo...
		Summerhill West, Rathnelly, South Hill, Forest...
		CN Tower, King and Spadina, Railway Lands, Har...
		New Toronto, Mimico South, Humber Bay Shores
		South Steeles, Silverstone, Humbergate, Jamest...
		Steeles West, L'Amoreaux West
		Rosedale
		Stn A PO Boxes
		Alderwood, Long Branch
		Northwest, West Humber - Clairville
		Upper Rouge
		St. James Town, Cabbagetown
		First Canadian Place, Underground city
		The Kingsway, Montgomery Road, Old Mill North
		Church and Wellesley
		Business reply mail Processing Centre, South C...
		Old Mill South, King's Mill Park, Sunnylea, Hu...

In [112]:

```
toronto_clean = toronto_df[toronto_df.Neighbourhood != 'Not assigned'].reset_index(drop=True)
toronto_clean
```

Out[112]:

Postal Code	Borough	Neighbourhood
0	M3A	North York
1	M4A	North York
2	M5A	Downtown Toronto
3	M6A	North York
4	M7A	Downtown Toronto
5	M9A	Etobicoke
6	M1B	Scarborough
7	M3B	North York
8	M4B	East York
9	M5B	Downtown Toronto
10	M6B	North York
11	M9B	Etobicoke
12	M1C	Scarborough
13	M3C	North York
14	M4C	East York
15	M5C	Downtown Toronto
16	M6C	York
17	M9C	Etobicoke
18	M1E	Scarborough
19	M4E	East Toronto
20	M5E	Downtown Toronto
21	M6E	York
22	M1G	Scarborough
23	M4G	East York
24	M5G	Downtown Toronto
25	M6G	Downtown Toronto
26	M1H	Scarborough
27	M2H	North York
28	M3H	North York
29	M4H	East York
30	M5H	Downtown Toronto
31	M6H	West Toronto
32	M1J	Scarborough
33	M2J	North York
		Parkwoods
		Victoria Village
		Regent Park, Harbourfront
		Lawrence Manor, Lawrence Heights
		Queen's Park, Ontario Provincial Government
		Islington Avenue, Humber Valley Village
		Malvern, Rouge
		Don Mills
		Parkview Hill, Woodbine Gardens
		Garden District, Ryerson
		Glencairn
		West Deane Park, Princess Gardens, Martin Grov...
		Rouge Hill, Port Union, Highland Creek
		Don Mills
		Woodbine Heights
		St. James Town
		Humewood-Cedarvale
		Eringate, Bloordale Gardens, Old Burnhamthorpe...
		Guildwood, Morningside, West Hill
		The Beaches
		Berczy Park
		Caledonia-Fairbanks
		Woburn
		Leaside
		Central Bay Street
		Christie
		Cedarbrae
		Hillcrest Village
		Bathurst Manor, Wilson Heights, Downsview North
		Thorncliffe Park
		Richmond, Adelaide, King
		Dufferin, Dovercourt Village
		Scarborough Village
		Fairview, Henry Farm, Oriole

Postal Code	Borough	Neighbourhood	
34	M3J	North York	Northwood Park, York University
35	M4J	East York	East Toronto, Broadview North (Old East York)
36	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands
37	M6J	West Toronto	Little Portugal, Trinity
38	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park
39	M2K	North York	Bayview Village
40	M3K	North York	Downsview
41	M4K	East Toronto	The Danforth West, Riverdale
42	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange
43	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place
44	M1L	Scarborough	Golden Mile, Clairlea, Oakridge
45	M2L	North York	York Mills, Silver Hills
46	M3L	North York	Downsview
47	M4L	East Toronto	India Bazaar, The Beaches West
48	M5L	Downtown Toronto	Commerce Court, Victoria Hotel
49	M6L	North York	North Park, Maple Leaf Park, Upwood Park
50	M9L	North York	Humber Summit
51	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West
52	M2M	North York	Willowdale, Newtonbrook
53	M3M	North York	Downsview
54	M4M	East Toronto	Studio District
55	M5M	North York	Bedford Park, Lawrence Manor East
56	M6M	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn
57	M9M	North York	Humberlea, Emery
58	M1N	Scarborough	Birch Cliff, Cliffside West
59	M2N	North York	Willowdale, Willowdale East
60	M3N	North York	Downsview
61	M4N	Central Toronto	Lawrence Park
62	M5N	Central Toronto	Roselawn
63	M6N	York	Runnymede, The Junction North
64	M9N	York	Weston
65	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...
66	M2P	North York	York Mills West
67	M4P	Central Toronto	Davisville North
68	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park
69	M6P	West Toronto	High Park, The Junction South

Postal Code	Borough	Neighbourhood
70	M9P	Etobicoke Westmount
71	M1R	Scarborough Wexford, Maryvale
72	M2R	North York Willowdale, Willowdale West
73	M4R	Central Toronto North Toronto West, Lawrence Park
74	M5R	Central Toronto The Annex, North Midtown, Yorkville
75	M6R	West Toronto Parkdale, Roncesvalles
76	M7R	Mississauga Canada Post Gateway Processing Centre
77	M9R	Etobicoke Kingsview Village, St. Phillips, Martin Grove ...
78	M1S	Scarborough Agincourt
79	M4S	Central Toronto Davisville
80	M5S	Downtown Toronto University of Toronto, Harbord
81	M6S	West Toronto Runnymede, Swansea
82	M1T	Scarborough Clarks Corners, Tam O'Shanter, Sullivan
83	M4T	Central Toronto Moore Park, Summerhill East
84	M5T	Downtown Toronto Kensington Market, Chinatown, Grange Park
85	M1V	Scarborough Milliken, Agincourt North, Steeles East, L'Amo...
86	M4V	Central Toronto Summerhill West, Rathnelly, South Hill, Forest...
87	M5V	Downtown Toronto CN Tower, King and Spadina, Railway Lands, Har...
88	M8V	Etobicoke New Toronto, Mimico South, Humber Bay Shores
89	M9V	Etobicoke South Steeles, Silverstone, Humbergate, Jamest...
90	M1W	Scarborough Steeles West, L'Amoreaux West
91	M4W	Downtown Toronto Rosedale
92	M5W	Downtown Toronto Stn A PO Boxes
93	M8W	Etobicoke Alderwood, Long Branch
94	M9W	Etobicoke Northwest, West Humber - Clairville
95	M1X	Scarborough Upper Rouge
96	M4X	Downtown Toronto St. James Town, Cabbagetown
97	M5X	Downtown Toronto First Canadian Place, Underground city
98	M8X	Etobicoke The Kingsway, Montgomery Road, Old Mill North
99	M4Y	Downtown Toronto Church and Wellesley
100	M7Y	East Toronto Business reply mail Processing Centre, South C...
101	M8Y	Etobicoke Old Mill South, King's Mill Park, Sunnylea, Hu...
102	M8Z	Etobicoke Mimico NW, The Queensway West, South of Bloor,...

In [113]:

```
#Combining the neighbourhoods that have the same postal code
toronto_clean = toronto_clean.groupby(['Postal Code', 'Borough'], sort=False).agg(', '.join
)
toronto_clean.reset_index(inplace=True)
toronto_clean
```

Out[113]:

Postal Code	Borough	Neighbourhood
0	M3A	North York
1	M4A	North York
2	M5A	Downtown Toronto
3	M6A	North York
4	M7A	Downtown Toronto
5	M9A	Etobicoke
6	M1B	Scarborough
7	M3B	North York
8	M4B	East York
9	M5B	Downtown Toronto
10	M6B	North York
11	M9B	Etobicoke
12	M1C	Scarborough
13	M3C	North York
14	M4C	East York
15	M5C	Downtown Toronto
16	M6C	York
17	M9C	Etobicoke
18	M1E	Scarborough
19	M4E	East Toronto
20	M5E	Downtown Toronto
21	M6E	York
22	M1G	Scarborough
23	M4G	East York
24	M5G	Downtown Toronto
25	M6G	Downtown Toronto
26	M1H	Scarborough
27	M2H	North York
28	M3H	North York
29	M4H	East York
30	M5H	Downtown Toronto
31	M6H	West Toronto
32	M1J	Scarborough
33	M2J	North York
		Parkwoods
		Victoria Village
		Regent Park, Harbourfront
		Lawrence Manor, Lawrence Heights
		Queen's Park, Ontario Provincial Government
		Islington Avenue, Humber Valley Village
		Malvern, Rouge
		Don Mills
		Parkview Hill, Woodbine Gardens
		Garden District, Ryerson
		Glencairn
		West Deane Park, Princess Gardens, Martin Grov...
		Rouge Hill, Port Union, Highland Creek
		Don Mills
		Woodbine Heights
		St. James Town
		Humewood-Cedarvale
		Eringate, Bloordale Gardens, Old Burnhamthorpe...
		Guildwood, Morningside, West Hill
		The Beaches
		Berczy Park
		Caledonia-Fairbanks
		Woburn
		Leaside
		Central Bay Street
		Christie
		Cedarbrae
		Hillcrest Village
		Bathurst Manor, Wilson Heights, Downsview North
		Thorncliffe Park
		Richmond, Adelaide, King
		Dufferin, Dovercourt Village
		Scarborough Village
		Fairview, Henry Farm, Oriole

Postal Code	Borough	Neighbourhood	
34	M3J	North York	Northwood Park, York University
35	M4J	East York	East Toronto, Broadview North (Old East York)
36	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands
37	M6J	West Toronto	Little Portugal, Trinity
38	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park
39	M2K	North York	Bayview Village
40	M3K	North York	Downsview
41	M4K	East Toronto	The Danforth West, Riverdale
42	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange
43	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place
44	M1L	Scarborough	Golden Mile, Clairlea, Oakridge
45	M2L	North York	York Mills, Silver Hills
46	M3L	North York	Downsview
47	M4L	East Toronto	India Bazaar, The Beaches West
48	M5L	Downtown Toronto	Commerce Court, Victoria Hotel
49	M6L	North York	North Park, Maple Leaf Park, Upwood Park
50	M9L	North York	Humber Summit
51	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West
52	M2M	North York	Willowdale, Newtonbrook
53	M3M	North York	Downsview
54	M4M	East Toronto	Studio District
55	M5M	North York	Bedford Park, Lawrence Manor East
56	M6M	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn
57	M9M	North York	Humberlea, Emery
58	M1N	Scarborough	Birch Cliff, Cliffside West
59	M2N	North York	Willowdale, Willowdale East
60	M3N	North York	Downsview
61	M4N	Central Toronto	Lawrence Park
62	M5N	Central Toronto	Roselawn
63	M6N	York	Runnymede, The Junction North
64	M9N	York	Weston
65	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...
66	M2P	North York	York Mills West
67	M4P	Central Toronto	Davisville North
68	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park
69	M6P	West Toronto	High Park, The Junction South

Postal Code	Borough	Neighbourhood
70	M9P	Etobicoke Westmount
71	M1R	Scarborough Wexford, Maryvale
72	M2R	North York Willowdale, Willowdale West
73	M4R	Central Toronto North Toronto West, Lawrence Park
74	M5R	Central Toronto The Annex, North Midtown, Yorkville
75	M6R	West Toronto Parkdale, Roncesvalles
76	M7R	Mississauga Canada Post Gateway Processing Centre
77	M9R	Etobicoke Kingsview Village, St. Phillips, Martin Grove ...
78	M1S	Scarborough Agincourt
79	M4S	Central Toronto Davisville
80	M5S	Downtown Toronto University of Toronto, Harbord
81	M6S	West Toronto Runnymede, Swansea
82	M1T	Scarborough Clarks Corners, Tam O'Shanter, Sullivan
83	M4T	Central Toronto Moore Park, Summerhill East
84	M5T	Downtown Toronto Kensington Market, Chinatown, Grange Park
85	M1V	Scarborough Milliken, Agincourt North, Steeles East, L'Amo...
86	M4V	Central Toronto Summerhill West, Rathnelly, South Hill, Forest...
87	M5V	Downtown Toronto CN Tower, King and Spadina, Railway Lands, Har...
88	M8V	Etobicoke New Toronto, Mimico South, Humber Bay Shores
89	M9V	Etobicoke South Steeles, Silverstone, Humbergate, Jamest...
90	M1W	Scarborough Steeles West, L'Amoreaux West
91	M4W	Downtown Toronto Rosedale
92	M5W	Downtown Toronto Stn A PO Boxes
93	M8W	Etobicoke Alderwood, Long Branch
94	M9W	Etobicoke Northwest, West Humber - Clairville
95	M1X	Scarborough Upper Rouge
96	M4X	Downtown Toronto St. James Town, Cabbagetown
97	M5X	Downtown Toronto First Canadian Place, Underground city
98	M8X	Etobicoke The Kingsway, Montgomery Road, Old Mill North
99	M4Y	Downtown Toronto Church and Wellesley
100	M7Y	East Toronto Business reply mail Processing Centre, South C...
101	M8Y	Etobicoke Old Mill South, King's Mill Park, Sunnylea, Hu...
102	M8Z	Etobicoke Mimico NW, The Queensway West, South of Bloor,...

In [114]:

```
# Replacing the name of the neighbourhoods which are 'Not assigned' with names of Borough
toronto_clean['Neighbourhood'] = np.where(toronto_clean['Neighbourhood'] == 'Not assigned',
, toronto_clean['Borough'], toronto_clean ['Neighbourhood'])
toronto_clean
```

Out[114]:

Postal Code	Borough	Neighbourhood
0	M3A	North York
1	M4A	North York
2	M5A	Downtown Toronto
3	M6A	North York
4	M7A	Downtown Toronto
5	M9A	Etobicoke
6	M1B	Scarborough
7	M3B	North York
8	M4B	East York
9	M5B	Downtown Toronto
10	M6B	North York
11	M9B	Etobicoke
12	M1C	Scarborough
13	M3C	North York
14	M4C	East York
15	M5C	Downtown Toronto
16	M6C	York
17	M9C	Etobicoke
18	M1E	Scarborough
19	M4E	East Toronto
20	M5E	Downtown Toronto
21	M6E	York
22	M1G	Scarborough
23	M4G	East York
24	M5G	Downtown Toronto
25	M6G	Downtown Toronto
26	M1H	Scarborough
27	M2H	North York
28	M3H	North York
29	M4H	East York
30	M5H	Downtown Toronto
31	M6H	West Toronto
32	M1J	Scarborough
33	M2J	North York
		Parkwoods
		Victoria Village
		Regent Park, Harbourfront
		Lawrence Manor, Lawrence Heights
		Queen's Park, Ontario Provincial Government
		Islington Avenue, Humber Valley Village
		Malvern, Rouge
		Don Mills
		Parkview Hill, Woodbine Gardens
		Garden District, Ryerson
		Glencairn
		West Deane Park, Princess Gardens, Martin Grov...
		Rouge Hill, Port Union, Highland Creek
		Don Mills
		Woodbine Heights
		St. James Town
		Humewood-Cedarvale
		Eringate, Bloordale Gardens, Old Burnhamthorpe...
		Guildwood, Morningside, West Hill
		The Beaches
		Berczy Park
		Caledonia-Fairbanks
		Woburn
		Leaside
		Central Bay Street
		Christie
		Cedarbrae
		Hillcrest Village
		Bathurst Manor, Wilson Heights, Downsview North
		Thorncliffe Park
		Richmond, Adelaide, King
		Dufferin, Dovercourt Village
		Scarborough Village
		Fairview, Henry Farm, Oriole

Postal Code	Borough	Neighbourhood	
34	M3J	North York	Northwood Park, York University
35	M4J	East York	East Toronto, Broadview North (Old East York)
36	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands
37	M6J	West Toronto	Little Portugal, Trinity
38	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park
39	M2K	North York	Bayview Village
40	M3K	North York	Downsview
41	M4K	East Toronto	The Danforth West, Riverdale
42	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange
43	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place
44	M1L	Scarborough	Golden Mile, Clairlea, Oakridge
45	M2L	North York	York Mills, Silver Hills
46	M3L	North York	Downsview
47	M4L	East Toronto	India Bazaar, The Beaches West
48	M5L	Downtown Toronto	Commerce Court, Victoria Hotel
49	M6L	North York	North Park, Maple Leaf Park, Upwood Park
50	M9L	North York	Humber Summit
51	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West
52	M2M	North York	Willowdale, Newtonbrook
53	M3M	North York	Downsview
54	M4M	East Toronto	Studio District
55	M5M	North York	Bedford Park, Lawrence Manor East
56	M6M	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn
57	M9M	North York	Humberlea, Emery
58	M1N	Scarborough	Birch Cliff, Cliffside West
59	M2N	North York	Willowdale, Willowdale East
60	M3N	North York	Downsview
61	M4N	Central Toronto	Lawrence Park
62	M5N	Central Toronto	Roselawn
63	M6N	York	Runnymede, The Junction North
64	M9N	York	Weston
65	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...
66	M2P	North York	York Mills West
67	M4P	Central Toronto	Davisville North
68	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park
69	M6P	West Toronto	High Park, The Junction South

Postal Code	Borough	Neighbourhood
70	M9P	Etobicoke
71	M1R	Scarborough
72	M2R	North York
73	M4R	Central Toronto
74	M5R	Central Toronto
75	M6R	West Toronto
76	M7R	Mississauga
77	M9R	Etobicoke
78	M1S	Scarborough
79	M4S	Central Toronto
80	M5S	Downtown Toronto
81	M6S	West Toronto
82	M1T	Scarborough
83	M4T	Central Toronto
84	M5T	Downtown Toronto
85	M1V	Scarborough
86	M4V	Central Toronto
87	M5V	Downtown Toronto
88	M8V	Etobicoke
89	M9V	Etobicoke
90	M1W	Scarborough
91	M4W	Downtown Toronto
92	M5W	Downtown Toronto
93	M8W	Etobicoke
94	M9W	Etobicoke
95	M1X	Scarborough
96	M4X	Downtown Toronto
97	M5X	Downtown Toronto
98	M8X	Etobicoke
99	M4Y	Downtown Toronto
100	M7Y	East Toronto
101	M8Y	Etobicoke
102	M8Z	Etobicoke
		Mimico NW, The Queensway West, South of Bloor,...
		Westmount
		Wexford, Maryvale
		Willowdale, Willowdale West
		North Toronto West, Lawrence Park
		The Annex, North Midtown, Yorkville
		Parkdale, Roncesvalles
		Canada Post Gateway Processing Centre
		Kingsview Village, St. Phillips, Martin Grove ...
		Agincourt
		Davisville
		University of Toronto, Harbord
		Runnymede, Swansea
		Clarks Corners, Tam O'Shanter, Sullivan
		Moore Park, Summerhill East
		Kensington Market, Chinatown, Grange Park
		Milliken, Agincourt North, Steeles East, L'Amo...
		Summerhill West, Rathnelly, South Hill, Forest...
		CN Tower, King and Spadina, Railway Lands, Har...
		New Toronto, Mimico South, Humber Bay Shores
		South Steeles, Silverstone, Humbergate, Jamest...
		Steeles West, L'Amoreaux West
		Rosedale
		Stn A PO Boxes
		Alderwood, Long Branch
		Northwest, West Humber - Clairville
		Upper Rouge
		St. James Town, Cabbagetown
		First Canadian Place, Underground city
		The Kingsway, Montgomery Road, Old Mill North
		Church and Wellesley
		Business reply mail Processing Centre, South C...
		Old Mill South, King's Mill Park, Sunnylea, Hu...

In [115]:

```
print('The dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(toronto_clean['Borough'].unique()),
    toronto_clean.shape[0]
))
```

The dataframe has 10 boroughs and 103 neighbourhoods.

In [116]:

```
# Shape of data frame
toronto_clean.shape
```

Out[116]:

(103, 3)

In [117]:

```
toronto_clean.describe()
```

Out[117]:

	Postal Code	Borough	Neighbourhood
count	103	103	103
unique	103	10	99
top	M4G	North York	Downsvie
freq	1	24	4

#Notebook_part2

Obtaining Latitude Longitude for each of the Coordinates using Foursquare data

In [118]:

```
!conda install -c conda-forge geopy --yes # Foursquare API latitude longitude
```

Collecting package metadata (current_repodata.json): done
Solving environment: done

All requested packages already installed.

In [119]:

```
!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haven't completed the Foursquare API lab
import folium # map rendering library
```

Collecting package metadata (current_repodata.json): done
Solving environment: done

All requested packages already installed.

In [120]:

```
!pip install geopy
```

Requirement already satisfied: geopy in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (2.1.0)
Requirement already satisfied: geographiclib<2,>=1.49 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geopy) (1.50)

In [121]:

```
!pip install geocoder
```

Requirement already satisfied: geocoder in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (1.38.1)
Requirement already satisfied: click in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geocoder) (7.1.2)
Requirement already satisfied: requests in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geocoder) (2.25.0)
Requirement already satisfied: ratelim in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geocoder) (0.1.6)
Requirement already satisfied: future in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geocoder) (0.18.2)
Requirement already satisfied: six in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geocoder) (1.15.0)
Requirement already satisfied: chardet<4,>=3.0.2 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests->geocoder) (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->geocoder) (1.25.11)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests->geocoder) (2020.12.5)
Requirement already satisfied: idna<3,>=2.5 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests->geocoder) (2.10)
Requirement already satisfied: decorator in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from ratelim->geocoder) (4.4.2)

In [122]:

```
from geopy.geocoders import Nominatim # convert an address into latitude and longitude values

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors

import random # Library for random number generation

# Libraries for displaying images
from IPython.display import Image
from IPython.core.display import HTML

! pip install folium==0.5.0
import folium # plotting library

# import k-means from clustering stage
from sklearn.cluster import KMeans
```

```
Requirement already satisfied: folium==0.5.0 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (0.5.0)
Requirement already satisfied: branca in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium==0.5.0) (0.4.2)
Requirement already satisfied: jinja2 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium==0.5.0) (2.11.2)
Requirement already satisfied: requests in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium==0.5.0) (2.25.0)
Requirement already satisfied: six in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium==0.5.0) (1.15.0)
Requirement already satisfied: MarkupSafe>=0.23 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from jinja2->folium==0.5.0) (1.1.1)
Requirement already satisfied: chardet<4,>=3.0.2 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests->folium==0.5.0) (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->folium==0.5.0) (1.25.11)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests->folium==0.5.0) (2020.12.5)
Requirement already satisfied: idna<3,>=2.5 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests->folium==0.5.0) (2.10)
```

Importing csv data containing the latitudes and longitudes for various neighbourhoods in Canada

In [123]:

```
#conversion to dataframe based on geospatial data in csv format
latlong = pd.read_csv('http://cocl.us/Geospatial_data')
latlong.head(103)
```

Out[123]:

	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
5	M1J	43.744734	-79.239476
6	M1K	43.727929	-79.262029
7	M1L	43.711112	-79.284577
8	M1M	43.716316	-79.239476
9	M1N	43.692657	-79.264848
10	M1P	43.757410	-79.273304
11	M1R	43.750072	-79.295849
12	M1S	43.794200	-79.262029
13	M1T	43.781638	-79.304302
14	M1V	43.815252	-79.284577
15	M1W	43.799525	-79.318389
16	M1X	43.836125	-79.205636
17	M2H	43.803762	-79.363452
18	M2J	43.778517	-79.346556
19	M2K	43.786947	-79.385975
20	M2L	43.757490	-79.374714
21	M2M	43.789053	-79.408493
22	M2N	43.770120	-79.408493
23	M2P	43.752758	-79.400049
24	M2R	43.782736	-79.442259
25	M3A	43.753259	-79.329656
26	M3B	43.745906	-79.352188
27	M3C	43.725900	-79.340923
28	M3H	43.754328	-79.442259
29	M3J	43.767980	-79.487262
30	M3K	43.737473	-79.464763
31	M3L	43.739015	-79.506944
32	M3M	43.728496	-79.495697
33	M3N	43.761631	-79.520999

	Postal Code	Latitude	Longitude
34	M4A	43.725882	-79.315572
35	M4B	43.706397	-79.309937
36	M4C	43.695344	-79.318389
37	M4E	43.676357	-79.293031
38	M4G	43.709060	-79.363452
39	M4H	43.705369	-79.349372
40	M4J	43.685347	-79.338106
41	M4K	43.679557	-79.352188
42	M4L	43.668999	-79.315572
43	M4M	43.659526	-79.340923
44	M4N	43.728020	-79.388790
45	M4P	43.712751	-79.390197
46	M4R	43.715383	-79.405678
47	M4S	43.704324	-79.388790
48	M4T	43.689574	-79.383160
49	M4V	43.686412	-79.400049
50	M4W	43.679563	-79.377529
51	M4X	43.667967	-79.367675
52	M4Y	43.665860	-79.383160
53	M5A	43.654260	-79.360636
54	M5B	43.657162	-79.378937
55	M5C	43.651494	-79.375418
56	M5E	43.644771	-79.373306
57	M5G	43.657952	-79.387383
58	M5H	43.650571	-79.384568
59	M5J	43.640816	-79.381752
60	M5K	43.647177	-79.381576
61	M5L	43.648198	-79.379817
62	M5M	43.733283	-79.419750
63	M5N	43.711695	-79.416936
64	M5P	43.696948	-79.411307
65	M5R	43.672710	-79.405678
66	M5S	43.662696	-79.400049
67	M5T	43.653206	-79.400049
68	M5V	43.628947	-79.394420
69	M5W	43.646435	-79.374846

	Postal Code	Latitude	Longitude
70	M5X	43.648429	-79.382280
71	M6A	43.718518	-79.464763
72	M6B	43.709577	-79.445073
73	M6C	43.693781	-79.428191
74	M6E	43.689026	-79.453512
75	M6G	43.669542	-79.422564
76	M6H	43.669005	-79.442259
77	M6J	43.647927	-79.419750
78	M6K	43.636847	-79.428191
79	M6L	43.713756	-79.490074
80	M6M	43.691116	-79.476013
81	M6N	43.673185	-79.487262
82	M6P	43.661608	-79.464763
83	M6R	43.648960	-79.456325
84	M6S	43.651571	-79.484450
85	M7A	43.662301	-79.389494
86	M7R	43.636966	-79.615819
87	M7Y	43.662744	-79.321558
88	M8V	43.605647	-79.501321
89	M8W	43.602414	-79.543484
90	M8X	43.653654	-79.506944
91	M8Y	43.636258	-79.498509
92	M8Z	43.628841	-79.520999
93	M9A	43.667856	-79.532242
94	M9B	43.650943	-79.554724
95	M9C	43.643515	-79.577201
96	M9L	43.756303	-79.565963
97	M9M	43.724766	-79.532242
98	M9N	43.706876	-79.518188
99	M9P	43.696319	-79.532242
100	M9R	43.688905	-79.554724
101	M9V	43.739416	-79.588437
102	M9W	43.706748	-79.594054

Merging the 'Postal Code' column from both dataframe to create a single dataframe that includes (Postal Code, Borough, Neighbourhood, Latitude, Longitude)

In [124]:

```
latlng_code = pd.merge(toronto_clean,latlng,on='Postal Code')  
latlng_code
```

Out[124]:

	Postal Code	Borough	Neighbourhood	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
5	M9A	Etobicoke	Islington Avenue, Humber Valley Village	43.667856	-79.532242
6	M1B	Scarborough	Malvern, Rouge	43.806686	-79.194353
7	M3B	North York	Don Mills	43.745906	-79.352188
8	M4B	East York	Parkview Hill, Woodbine Gardens	43.706397	-79.309937
9	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
10	M6B	North York	Glencairn	43.709577	-79.445073
11	M9B	Etobicoke	West Deane Park, Princess Gardens, Martin Grov...	43.650943	-79.554724
12	M1C	Scarborough	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497
13	M3C	North York	Don Mills	43.725900	-79.340923
14	M4C	East York	Woodbine Heights	43.695344	-79.318389
15	M5C	Downtown Toronto	St. James Town	43.651494	-79.375418
16	M6C	York	Humewood-Cedarvale	43.693781	-79.428191
17	M9C	Etobicoke	Eringate, Bloordale Gardens, Old Burnhamthorpe...	43.643515	-79.577201
18	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711
19	M4E	East Toronto	The Beaches	43.676357	-79.293031
20	M5E	Downtown Toronto	Berczy Park	43.644771	-79.373306
21	M6E	York	Caledonia-Fairbanks	43.689026	-79.453512
22	M1G	Scarborough	Woburn	43.770992	-79.216917
23	M4G	East York	Leaside	43.709060	-79.363452
24	M5G	Downtown Toronto	Central Bay Street	43.657952	-79.387383
25	M6G	Downtown Toronto	Christie	43.669542	-79.422564
26	M1H	Scarborough	Cedarbrae	43.773136	-79.239476
27	M2H	North York	Hillcrest Village	43.803762	-79.363452

Postal Code	Borough	Neighbourhood	Latitude	Longitude
28	M3H	North York	Bathurst Manor, Wilson Heights, Downsview North	43.754328 -79.442259
29	M4H	East York	Thorncliffe Park	43.705369 -79.349372
30	M5H	Downtown Toronto	Richmond, Adelaide, King	43.650571 -79.384568
31	M6H	West Toronto	Dufferin, Dovercourt Village	43.669005 -79.442259
32	M1J	Scarborough	Scarborough Village	43.744734 -79.239476
33	M2J	North York	Fairview, Henry Farm, Oriole	43.778517 -79.346556
34	M3J	North York	Northwood Park, York University	43.767980 -79.487262
35	M4J	East York	East Toronto, Broadview North (Old East York)	43.685347 -79.338106
36	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands	43.640816 -79.381752
37	M6J	West Toronto	Little Portugal, Trinity	43.647927 -79.419750
38	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park	43.727929 -79.262029
39	M2K	North York	Bayview Village	43.786947 -79.385975
40	M3K	North York	Downsview	43.737473 -79.464763
41	M4K	East Toronto	The Danforth West, Riverdale	43.679557 -79.352188
42	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange	43.647177 -79.381576
43	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place	43.636847 -79.428191
44	M1L	Scarborough	Golden Mile, Clairlea, Oakridge	43.711112 -79.284577
45	M2L	North York	York Mills, Silver Hills	43.757490 -79.374714
46	M3L	North York	Downsview	43.739015 -79.506944
47	M4L	East Toronto	India Bazaar, The Beaches West	43.668999 -79.315572
48	M5L	Downtown Toronto	Commerce Court, Victoria Hotel	43.648198 -79.379817
49	M6L	North York	North Park, Maple Leaf Park, Upwood Park	43.713756 -79.490074
50	M9L	North York	Humber Summit	43.756303 -79.565963
51	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West	43.716316 -79.239476
52	M2M	North York	Willowdale, Newtonbrook	43.789053 -79.408493
53	M3M	North York	Downsview	43.728496 -79.495697
54	M4M	East Toronto	Studio District	43.659526 -79.340923
55	M5M	North York	Bedford Park, Lawrence Manor East	43.733283 -79.419750
56	M6M	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn	43.691116 -79.476013
57	M9M	North York	Humberlea, Emery	43.724766 -79.532242
58	M1N	Scarborough	Birch Cliff, Cliffside West	43.692657 -79.264848

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
59	M2N	North York	Willowdale, Willowdale East	43.770120	-79.408493
60	M3N	North York	Downsview	43.761631	-79.520999
61	M4N	Central Toronto	Lawrence Park	43.728020	-79.388790
62	M5N	Central Toronto	Roselawn	43.711695	-79.416936
63	M6N	York	Runnymede, The Junction North	43.673185	-79.487262
64	M9N	York	Weston	43.706876	-79.518188
65	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...	43.757410	-79.273304
66	M2P	North York	York Mills West	43.752758	-79.400049
67	M4P	Central Toronto	Davisville North	43.712751	-79.390197
68	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park	43.696948	-79.411307
69	M6P	West Toronto	High Park, The Junction South	43.661608	-79.464763
70	M9P	Etobicoke	Westmount	43.696319	-79.532242
71	M1R	Scarborough	Wexford, Maryvale	43.750072	-79.295849
72	M2R	North York	Willowdale, Willowdale West	43.782736	-79.442259
73	M4R	Central Toronto	North Toronto West, Lawrence Park	43.715383	-79.405678
74	M5R	Central Toronto	The Annex, North Midtown, Yorkville	43.672710	-79.405678
75	M6R	West Toronto	Parkdale, Roncesvalles	43.648960	-79.456325
76	M7R	Mississauga	Canada Post Gateway Processing Centre	43.636966	-79.615819
77	M9R	Etobicoke	Kingsview Village, St. Phillips, Martin Grove ...	43.688905	-79.554724
78	M1S	Scarborough	Agincourt	43.794200	-79.262029
79	M4S	Central Toronto	Davisville	43.704324	-79.388790
80	M5S	Downtown Toronto	University of Toronto, Harbord	43.662696	-79.400049
81	M6S	West Toronto	Runnymede, Swansea	43.651571	-79.484450
82	M1T	Scarborough	Clarks Corners, Tam O'Shanter, Sullivan	43.781638	-79.304302
83	M4T	Central Toronto	Moore Park, Summerhill East	43.689574	-79.383160
84	M5T	Downtown Toronto	Kensington Market, Chinatown, Grange Park	43.653206	-79.400049
85	M1V	Scarborough	Milliken, Agincourt North, Steeles East, L'Amo...	43.815252	-79.284577
86	M4V	Central Toronto	Summerhill West, Rathnelly, South Hill, Forest...	43.686412	-79.400049
87	M5V	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...	43.628947	-79.394420
88	M8V	Etobicoke	New Toronto, Mimico South, Humber Bay Shores	43.605647	-79.501321

Postal Code	Borough	Neighbourhood	Latitude	Longitude
89	M9V	Etobicoke South Steeles, Silverstone, Humbergate, Jamest...	43.739416	-79.588437
90	M1W	Scarborough Steeles West, L'Amoreaux West	43.799525	-79.318389
91	M4W	Downtown Toronto Rosedale	43.679563	-79.377529
92	M5W	Downtown Toronto Stn A PO Boxes	43.646435	-79.374846
93	M8W	Etobicoke Alderwood, Long Branch	43.602414	-79.543484
94	M9W	Etobicoke Northwest, West Humber - Clairville	43.706748	-79.594054
95	M1X	Scarborough Upper Rouge	43.836125	-79.205636
96	M4X	Downtown Toronto St. James Town, Cabbagetown	43.667967	-79.367675
97	M5X	Downtown Toronto First Canadian Place, Underground city	43.648429	-79.382280
98	M8X	Etobicoke The Kingsway, Montgomery Road, Old Mill North	43.653654	-79.506944
99	M4Y	Downtown Toronto Church and Wellesley	43.665860	-79.383160
100	M7Y	East Toronto Business reply mail Processing Centre, South C...	43.662744	-79.321558
101	M8Y	Etobicoke Old Mill South, King's Mill Park, Sunnylea, Hu...	43.636258	-79.498509
102	M8Z	Etobicoke Mimico NW, The Queensway West, South of Bloor,...	43.628841	-79.520999

#Notebook_part3

Exploring and clustering the neighborhoods in Toronto

In [125]:

```
from pandas.io.json import json_normalize # transform JSON file into a pandas dataframe
```

In [126]:

```
latlng_toronto=latlng_code[latlng_code['Borough'].str.contains("Toronto")]
latlng_toronto
```

Out[126]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
2	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
4	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494
9	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
15	M5C	Downtown Toronto	St. James Town	43.651494	-79.375418
19	M4E	East Toronto	The Beaches	43.676357	-79.293031
20	M5E	Downtown Toronto	Berczy Park	43.644771	-79.373306
24	M5G	Downtown Toronto	Central Bay Street	43.657952	-79.387383
25	M6G	Downtown Toronto	Christie	43.669542	-79.422564
30	M5H	Downtown Toronto	Richmond, Adelaide, King	43.650571	-79.384568
31	M6H	West Toronto	Dufferin, Dovercourt Village	43.669005	-79.442259
36	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands	43.640816	-79.381752
37	M6J	West Toronto	Little Portugal, Trinity	43.647927	-79.419750
41	M4K	East Toronto	The Danforth West, Riverdale	43.679557	-79.352188
42	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange	43.647177	-79.381576
43	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place	43.636847	-79.428191
47	M4L	East Toronto	India Bazaar, The Beaches West	43.668999	-79.315572
48	M5L	Downtown Toronto	Commerce Court, Victoria Hotel	43.648198	-79.379817
54	M4M	East Toronto	Studio District	43.659526	-79.340923
61	M4N	Central Toronto	Lawrence Park	43.728020	-79.388790
62	M5N	Central Toronto	Roselawn	43.711695	-79.416936
67	M4P	Central Toronto	Davisville North	43.712751	-79.390197
68	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park	43.696948	-79.411307
69	M6P	West Toronto	High Park, The Junction South	43.661608	-79.464763
73	M4R	Central Toronto	North Toronto West, Lawrence Park	43.715383	-79.405678
74	M5R	Central Toronto	The Annex, North Midtown, Yorkville	43.672710	-79.405678
75	M6R	West Toronto	Parkdale, Roncesvalles	43.648960	-79.456325
79	M4S	Central Toronto	Davisville	43.704324	-79.388790

Postal Code	Borough	Neighbourhood	Latitude	Longitude
80	M5S	Downtown Toronto	University of Toronto, Harbord	43.662696 -79.400049
81	M6S	West Toronto	Runnymede, Swansea	43.651571 -79.484450
83	M4T	Central Toronto	Moore Park, Summerhill East	43.689574 -79.383160
84	M5T	Downtown Toronto	Kensington Market, Chinatown, Grange Park	43.653206 -79.400049
86	M4V	Central Toronto	Summerhill West, Rathnelly, South Hill, Forest...	43.686412 -79.400049
87	M5V	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...	43.628947 -79.394420
91	M4W	Downtown Toronto	Rosedale	43.679563 -79.377529
92	M5W	Downtown Toronto	Stn A PO Boxes	43.646435 -79.374846
96	M4X	Downtown Toronto	St. James Town, Cabbagetown	43.667967 -79.367675
97	M5X	Downtown Toronto	First Canadian Place, Underground city	43.648429 -79.382280
99	M4Y	Downtown Toronto	Church and Wellesley	43.665860 -79.383160
100	M7Y	East Toronto	Business reply mail Processing Centre, South C...	43.662744 -79.321558

In [127]:

```
latlng_toronto.shape
```

Out[127]:

```
(39, 5)
```

In [128]:

```
print('Toronto dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(latlng_toronto['Borough'].unique()),
    latlng_toronto.shape[0]
)
)
```

Toronto dataframe has 4 boroughs and 39 neighbourhoods.

In [129]:

```
address = 'Toronto, Ontario'
geolocator = Nominatim(user_agent="tor_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geographical coordinate of Toronto are {}, {}'.format(latitude, longitude))
```

The geographical coordinate of Toronto are 43.6534817, -79.3839347.

In [130]:

```
# create map of Toronto using Latitude and Longitude values
map_toronto = folium.Map(location=[43.6534817, -79.3839347], zoom_start=11)
map_toronto
```

Out[130]:

Make this Notebook Trusted to load map: File -> Trust Notebook

In [131]:

```
# add markers to map (Borough)
for lat, lng, label in zip(latlng_toronto['Latitude'], latlng_toronto['Longitude'], latlng_toronto['Borough']):
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_toronto)

map_toronto
```

Out[131]:

Make this Notebook Trusted to load map: File -> Trust Notebook

In [132]:

```
# add markers to map (Neighbourhood)

for lat, lng, label in zip(latlng_toronto['Latitude'], latlng_toronto['Longitude'], latlng_toronto['Neighbourhood']):
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_toronto)

map_toronto
```

Out[132]:

Make this Notebook Trusted to load map: File -> Trust Notebook

In [133]:

```
DowntownToronto_data = latlng_toronto[latlng_toronto['Borough'] == 'Downtown Toronto'].reset_index(drop=True)
DowntownToronto_data.head(25)
```

Out[133]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
1	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494
2	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
3	M5C	Downtown Toronto	St. James Town	43.651494	-79.375418
4	M5E	Downtown Toronto	Berczy Park	43.644771	-79.373306
5	M5G	Downtown Toronto	Central Bay Street	43.657952	-79.387383
6	M6G	Downtown Toronto	Christie	43.669542	-79.422564
7	M5H	Downtown Toronto	Richmond, Adelaide, King	43.650571	-79.384568
8	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands	43.640816	-79.381752
9	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange	43.647177	-79.381576
10	M5L	Downtown Toronto	Commerce Court, Victoria Hotel	43.648198	-79.379817
11	M5S	Downtown Toronto	University of Toronto, Harbord	43.662696	-79.400049
12	M5T	Downtown Toronto	Kensington Market, Chinatown, Grange Park	43.653206	-79.400049
13	M5V	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...	43.628947	-79.394420
14	M4W	Downtown Toronto	Rosedale	43.679563	-79.377529
15	M5W	Downtown Toronto	Stn A PO Boxes	43.646435	-79.374846
16	M4X	Downtown Toronto	St. James Town, Cabbagetown	43.667967	-79.367675
17	M5X	Downtown Toronto	First Canadian Place, Underground city	43.648429	-79.382280
18	M4Y	Downtown Toronto	Church and Wellesley	43.665860	-79.383160

In [134]:

```
EastToronto_data = latlng_toronto[latlng_toronto['Borough'] == 'East Toronto'].reset_index
(drop=True)
EastToronto_data.head(25)
```

Out[134]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M4E	East Toronto	The Beaches	43.676357	-79.293031
1	M4K	East Toronto	The Danforth West, Riverdale	43.679557	-79.352188
2	M4L	East Toronto	India Bazaar, The Beaches West	43.668999	-79.315572
3	M4M	East Toronto	Studio District	43.659526	-79.340923
4	M7Y	East Toronto	Business reply mail Processing Centre, South C...	43.662744	-79.321558

In [135]:

```
WestToronto_data = latlng_toronto[latlng_toronto['Borough'] == 'West Toronto'].reset_index
(drop=True)
WestToronto_data.head(25)
```

Out[135]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M6H	West Toronto	Dufferin, Dovercourt Village	43.669005	-79.442259
1	M6J	West Toronto	Little Portugal, Trinity	43.647927	-79.419750
2	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place	43.636847	-79.428191
3	M6P	West Toronto	High Park, The Junction South	43.661608	-79.464763
4	M6R	West Toronto	Parkdale, Roncesvalles	43.648960	-79.456325
5	M6S	West Toronto	Runnymede, Swansea	43.651571	-79.484450

In [136]:

```
CentralToronto_data = latlng_toronto[latlng_toronto['Borough'] == 'Central Toronto'].reset_index(drop=True)
CentralToronto_data.head(25)
```

Out[136]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M4N	Central Toronto	Lawrence Park	43.728020	-79.388790
1	M5N	Central Toronto	Roselawn	43.711695	-79.416936
2	M4P	Central Toronto	Davisville North	43.712751	-79.390197
3	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park	43.696948	-79.411307
4	M4R	Central Toronto	North Toronto West, Lawrence Park	43.715383	-79.405678
5	M5R	Central Toronto	The Annex, North Midtown, Yorkville	43.672710	-79.405678
6	M4S	Central Toronto	Davisville	43.704324	-79.388790
7	M4T	Central Toronto	Moore Park, Summerhill East	43.689574	-79.383160
8	M4V	Central Toronto	Summerhill West, Rathnelly, South Hill, Forest...	43.686412	-79.400049

In [137]:

```
print('Downtown Toronto dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(DowntownToronto_data['Borough'].unique()),
    DowntownToronto_data.shape[0]
)
)
```

Downtown Toronto dataframe has 1 boroughs and 19 neighbourhoods.

In [138]:

```
print('East Toronto dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(EastToronto_data['Borough'].unique()),
    EastToronto_data.shape[0]
)
)
```

East Toronto dataframe has 1 boroughs and 5 neighbourhoods.

In [139]:

```
print('West Toronto dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(WestToronto_data['Borough'].unique()),
    WestToronto_data.shape[0]
)
)
```

West Toronto dataframe has 1 boroughs and 6 neighbourhoods.

In [140]:

```
print('Central Toronto dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(CentralToronto_data['Borough'].unique()),
    CentralToronto_data.shape[0]
)
)
```

Central Toronto dataframe has 1 boroughs and 9 neighbourhoods.

We will explore Downtown Toronto and its neighbourhoods

Define Foursquare credentials and version

In [141]:

```
CLIENT_ID = 'KW0JV4RJ1HGQL2P4RVDTXMOC4HBRFMQA3GLD0CWFMB4GTEMC' # your Foursquare ID
CLIENT_SECRET = 'QMYKCS1PKC4WAINWPUAUQ1ALE25NPDKSQVV0Y025PEETYX2' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version
LIMIT = 100 # A default Foursquare API limit value
print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET: ' + CLIENT_SECRET)
```

Your credentails:

```
CLIENT_ID: KW0JV4RJ1HGQL2P4RVDTXMOC4HBRFMQA3GLD0CWFMB4GTEMC
CLIENT_SECRET: QMYKCS1PKC4WAINWPUAUQ1ALE25NPDKSQVV0Y025PEETYX2
```

In [142]:

#Let's explore the first neighborhood in our dataframe

```
DowntownToronto_data.loc[0, 'Neighbourhood']
```

Out[142]:

```
'Regent Park, Harbourfront'
```

In [143]:

```
neighbourhood_latitude = DowntownToronto_data.loc[0, 'Latitude'] # neighborhood latitude value
neighbourhood_longitude = DowntownToronto_data.loc[0, 'Longitude'] # neighborhood longitude value
neighbourhood_name = DowntownToronto_data.loc[0, 'Neighbourhood'] # neighborhood name
print('Latitude and longitude values of {} are {}, {}.'.format(neighbourhood_name,
                                                               neighbourhood_latitude,
                                                               neighbourhood_longitude))
```

Latitude and longitude values of Regent Park, Harbourfront are 43.6542599, -79.3606359.

Now, let's get the top 100 venues that are in Regent Park, Harbourfront within a radius of 500 meters

In [144]:

```
# type your answer here
LIMIT = 100
radius = 500
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={} ,{}&radius={}&limit={}'.format(CLIENT_ID,
                                                CLIENT_SECRET,
                                                VERSION,
                                                neighbourhood_latitude,
                                                neighbourhood_longitude,
                                                radius,
                                                LIMIT)
url
```

Out[144]:

```
'https://api.foursquare.com/v2/venues/explore?&client_id=KW0JV4RJ1HGQL2P4RVDT
XMOOC4HBRFMQA3GLD0CWFMB4GTEMC&client_secret=QMYKCS1PKC4WAINWPUAUQ1ALE25NPDKSQ
VV0Y025PEETYX2&v=20180605&ll=43.6542599,-79.3606359&radius=500&limit=100'
```

In [145]:

```
results = requests.get(url).json()  
results
```

Out[145]:

```
{'meta': {'code': 200, 'requestId': '601e622583076b1f23e61e32'},  
 'response': {'headerLocation': 'Corktown',  
 'headerFullLocation': 'Corktown, Toronto',  
 'headerLocationGranularity': 'neighborhood',  
 'totalResults': 45,  
 'suggestedBounds': {'ne': {'lat': 43.6587599045, 'lng': -79.3544279001486},  
 'sw': {'lat': 43.6497598955, 'lng': -79.36684389985142}},  
 'groups': [{{'type': 'Recommended Places',  
 'name': 'recommended',  
 'items': [{{'reasons': {'count': 0,  
 'items': [{{'summary': 'This spot is popular',  
 'type': 'general',  
 'reasonName': 'globalInteractionReason'}}]}},  
 'venue': {'id': '53b8466a498e83df908c3f21',  
 'name': 'Tandem Coffee',  
 'location': {'address': '368 King St E',  
 'crossStreet': 'at Trinity St',  
 'lat': 43.65355870959944,  
 'lng': -79.36180945913513,  
 'labeledLatLngs': [{{'label': 'display',  
 'lat': 43.65355870959944,  
 'lng': -79.36180945913513}],  
 'distance': 122,  
 'cc': 'CA',  
 'city': 'Toronto',  
 'state': 'ON',  
 'country': 'Canada',  
 'formattedAddress': ['368 King St E (at Trinity St)',  
 'Toronto ON',  
 'Canada']},  
 'categories': [{{'id': '4bf58dd8d48988d1e0931735',  
 'name': 'Coffee Shop',  
 'pluralName': 'Coffee Shops',  
 'shortName': 'Coffee Shop',  
 'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coff  
 eeshop_',  
 'suffix': '.png'},  
 'primary': True}],  
 'photos': {'count': 0, 'groups': []}},  
 'referralId': 'e-0-53b8466a498e83df908c3f21-0'}},  
 {''reasons': {'count': 0,  
 'items': [{{'summary': 'This spot is popular',  
 'type': 'general',  
 'reasonName': 'globalInteractionReason'}}]}},  
 'venue': {'id': '54ea41ad498e9a11e9e13308',  
 'name': 'Roselle Desserts',  
 'location': {'address': '362 King St E',  
 'crossStreet': 'Trinity St',  
 'lat': 43.653446723052674,  
 'lng': -79.3620167174383,  
 'labeledLatLngs': [{{'label': 'display',  
 'lat': 43.653446723052674,  
 'lng': -79.3620167174383}],  
 'distance': 143,  
 'postalCode': 'M5A 1K9',
```

```
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['362 King St E (Trinity St)',
'Toronto ON M5A 1K9',
'Canada']],
'categories': [{}{'id': '4bf58dd8d48988d16a941735',
'name': 'Bakery',
'pluralName': 'Bakeries',
'shortName': 'Bakery',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/bake
ry_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-54ea41ad498e9a11e9e13308-1'},
{'reasons': {'count': 0,
'items': [{}{'summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]},
'venue': {'id': '574c229e498ebb5c6b257902',
'name': 'Cooper Koo Family YMCA',
'location': {'address': '461 Cherry St',
'lat': 43.65324910177244,
'lng': -79.35800826343677,
'labeledLatLngs': [{}{'label': 'display',
'lat': 43.65324910177244,
'lng': -79.35800826343677}],
'distance': 239,
'postalCode': 'M5A 0H7',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['461 Cherry St', 'Toronto ON M5A 0H7', 'Canad
a']},
'categories': [{}{'id': '52e81612bcfc57f1066b7a37',
'name': 'Distribution Center',
'pluralName': 'Distribution Centers',
'shortName': 'Distributor',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/
default_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-574c229e498ebb5c6b257902-2'},
{'reasons': {'count': 0,
'items': [{}{'summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]},
'venue': {'id': '50760559e4b0e8c7babe2497',
'name': 'Body Blitz Spa East',
'location': {'address': '497 King Street East',
'crossStreet': 'btwn Sackville St and Sumach St',
'lat': 43.65473505045365,
'lng': -79.35987433132891,
```

```
'labeledLatLngs': [{label: 'display',
  'lat': 43.65473505045365,
  'lng': -79.35987433132891}],
'distance': 80,
'postalCode': 'M5A 1L9',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['497 King Street East (btwn Sackville St and Sum
ach St)'],
'Toronto ON M5A 1L9',
'Canada']},
'categories': [{id: '4bf58dd8d48988d1ed941735',
  'name': 'Spa',
  'pluralName': 'Spas',
  'shortName': 'Spa',
  'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/spa
-',
    'suffix': '.png'},
  'primary': True}],
'photos': {'count': 0, 'groups': []}},
'reerralId': 'e-0-50760559e4b0e8c7babe2497-3'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '5612b1cc498e3dd742af0dc8',
  'name': 'Impact Kitchen',
  'location': {'address': '573 King St E',
    'crossStreet': 'at St Lawrence St',
    'lat': 43.65636850543279,
    'lng': -79.35697968750694,
    'labeledLatLngs': [{label: 'display',
      'lat': 43.65636850543279,
      'lng': -79.35697968750694}],
    'distance': 376,
    'postalCode': 'M5A 4L3',
    'cc': 'CA',
    'city': 'Toronto',
    'state': 'ON',
    'country': 'Canada',
    'formattedAddress': ['573 King St E (at St Lawrence St)',
      'Toronto ON M5A 4L3',
      'Canada']}},
  'categories': [{id: '4bf58dd8d48988d1c4941735',
    'name': 'Restaurant',
    'pluralName': 'Restaurants',
    'shortName': 'Restaurant',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/defa
ult_',
      'suffix': '.png'},
    'primary': True}],
    'photos': {'count': 0, 'groups': []}},
    'referralId': 'e-0-5612b1cc498e3dd742af0dc8-4'},
    {'reasons': {'count': 0,
      'items': [{summary: 'This spot is popular',
        'type': 'general',
        'reasonName': 'globalInteractionReason'}]}}]
```

```
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '566e1294498e3f6629006bc3',
'name': 'Dominion Pub and Kitchen',
'location': {'address': '500 Queen Street East',
'lat': 43.65691857501867,
'lng': -79.35896684476664,
'labeledLatLngs': [{"label": "display",
'lat': 43.65691857501867,
'lng': -79.35896684476664}],
'distance': 325,
'postalCode': 'M5A 1T9',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['500 Queen Street East',
'Toronto ON M5A 1T9',
'Canada']},
'categories': [{"id': '4bf58dd8d48988d11b941735',
'name': 'Pub',
'pluralName': 'Pubs',
'shortName': 'Pub',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/nightlife/pub_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-566e1294498e3f6629006bc3-5'},
{'reasons': {'count': 0,
'items': [{"summary": 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]},
'venue': {'id': '4ae5b91ff964a520a6a121e3',
'name': 'Morning Glory Cafe',
'location': {'address': '457 King St. E',
'crossStreet': 'Gilead Place',
'lat': 43.653946942635294,
'lng': -79.36114884214422,
'labeledLatLngs': [{"label": "display",
'lat': 43.653946942635294,
'lng': -79.36114884214422}],
'distance': 54,
'postalCode': 'M5A 1L6',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['457 King St. E (Gilead Place)',
'Toronto ON M5A 1L6',
'Canada']},
'categories': [{"id': '4bf58dd8d48988d143941735',
'name': 'Breakfast Spot',
'pluralName': 'Breakfast Spots',
'shortName': 'Breakfast',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/breakfast_',
'suffix': '.png'},
'primary': True}]}]
```

```
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []},
'venuePage': {'id': '39686393'}},
'referralId': 'e-0-4ae5b91ff964a520a6a121e3-6'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '51ccc048498ec7792efc955e',
'name': 'Corktown Common',
'location': {'lat': 43.655617799749734,
'lng': -79.3562113397429,
'labeledLatLngs': [{'label': 'display',
'lat': 43.655617799749734,
'lng': -79.3562113397429}],
'distance': 387,
'cc': 'CA',
'country': 'Canada',
'formattedAddress': ['Canada']},
'categories': [{'id': '4bf58dd8d48988d163941735',
'name': 'Park',
'pluralName': 'Parks',
'shortName': 'Park',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/parks_outdoors/park_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-51ccc048498ec7792efc955e-7'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4e8b7fa1cc2112f67517660a',
'name': 'The Extension Room',
'location': {'address': '30 Eastern Ave',
'crossStreet': 'Sackville St.',
'lat': 43.65331304337331,
'lng': -79.35972538072777,
'labeledLatLngs': [{'label': 'display',
'lat': 43.65331304337331,
'lng': -79.35972538072777}],
'distance': 128,
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['30 Eastern Ave (Sackville St.)',
'Toronto ON',
'Canada']},
'categories': [{'id': '4bf58dd8d48988d175941735',
'name': 'Gym / Fitness Center',
'pluralName': 'Gyms or Fitness Centers',
'shortName': 'Gym / Fitness',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/gym_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}}]
```

```
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4e8b7fa1cc2112f67517660a-8'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4ad4c05ef964a520bff620e3',
'name': 'The Distillery Historic District',
'location': {'address': 'btwn Front, Cherry, Gardiner & Parliament',
'lat': 43.65024435658077,
'lng': -79.35932278633118,
'labeledLatLngs': [{'label': 'display',
'lat': 43.65024435658077,
'lng': -79.35932278633118}],
'distance': 459,
'postalCode': 'M5A 3C4',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['btwn Front, Cherry, Gardiner & Parliament',
'Toronto ON M5A 3C4',
'Canada']}},
'categories': [{id: '4deefb944765f83613cdba6e',
'name': 'Historic Site',
'pluralName': 'Historic Sites',
'shortName': 'Historic Site',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/arts_entertainment/historicsite_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4ad4c05ef964a520bff620e3-9'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4c3e1eaa6faac9b66dc60d76',
'name': 'Distillery Sunday Market',
'location': {'address': '1 Trinity St',
'lat': 43.650074989330655,
'lng': -79.36183171318665,
'labeledLatLngs': [{'label': 'display',
'lat': 43.650074989330655,
'lng': -79.36183171318665}],
'distance': 475,
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['1 Trinity St', 'Toronto ON', 'Canada']}},
'categories': [{id: '4bf58dd8d48988d1fa941735',
'name': 'Farmers Market',
'pluralName': 'Farmers Markets',
'shortName': "Farmer's Market",
```

```
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/fo
d_farmersmarket_',
          'suffix': '.png'},
          'primary': True}],
          'photos': {'count': 0, 'groups': []}},
          'referralId': 'e-0-4c3e1eaa6faac9b66dc60d76-10'},
          {'reasons': {'count': 0,
                      'items': [{"summary": 'This spot is popular',
                                 'type': 'general',
                                 'reasonName': 'globalInteractionReason'}]}},
          'venue': {'id': '4b0978e1f964a520cd1723e3',
                    'name': 'SOMA chocolatemaker',
                    'location': {'address': '55 Mill Street, Unit #48',
                                 'crossStreet': 'The Distillery District',
                                 'lat': 43.65062222570758,
                                 'lng': -79.35812684032683,
                                 'labeledLatLngs': [{"label": 'display',
                                         'lat': 43.65062222570758,
                                         'lng': -79.35812684032683}],
                                 'distance': 452,
                                 'postalCode': 'M5A 3C4',
                                 'cc': 'CA',
                                 'city': 'Toronto',
                                 'state': 'ON',
                                 'country': 'Canada',
                                 'formattedAddress': ['55 Mill Street, Unit #48 (The Distillery Distri
ct)', 'Toronto ON M5A 3C4',
                                         'Canada']},
                    'categories': [{"id': '52f2ab2ebcbc57f1066b8b31',
                                    'name': 'Chocolate Shop',
                                    'pluralName': 'Chocolate Shops',
                                    'shortName': 'Chocolate Shop',
                                    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/can
dystore_',
                                             'suffix': '.png'},
                                             'primary': True}],
                    'photos': {'count': 0, 'groups': []}},
                    'referralId': 'e-0-4b0978e1f964a520cd1723e3-11'},
                    {'reasons': {'count': 0,
                                'items': [{"summary": 'This spot is popular',
                                           'type': 'general',
                                           'reasonName': 'globalInteractionReason'}]}},
                    'venue': {'id': '4af59046f964a520e0f921e3',
                              'name': 'Figs Breakfast & Lunch',
                              'location': {'address': '344 Queen St. E.',
                                           'crossStreet': 'at Parliament St.',
                                           'lat': 43.65567455427388,
                                           'lng': -79.3645032892494,
                                           'labeledLatLngs': [{"label": 'display',
                                                   'lat': 43.65567455427388,
                                                   'lng': -79.3645032892494}],
                                           'distance': 349,
                                           'postalCode': 'M5A 1S8',
                                           'cc': 'CA',
                                           'city': 'Toronto',
                                           'state': 'ON',
```

```
'country': 'Canada',
'formattedAddress': ['344 Queen St. E. (at Parliament St.)',
'Toronto ON M5A 1S8',
'Canada']],
'categories': [{id: '4bf58dd8d48988d143941735',
'name': 'Breakfast Spot',
'pluralName': 'Breakfast Spots',
'shortName': 'Breakfast',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/breakfast_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4af59046f964a520e0f921e3-12'},
{'reasons': {'count': 0,
'items': [{summary: 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {id: '5619551a498e9e35fce2256b',
'name': 'Sumach Espresso',
'location': {address: '118 Sumach St',
'crossStreet': 'at Shuter St',
'lat': 43.65813540553308,
'lng': -79.35951549011845,
'labeledLatLngs': [{label: 'display',
'lat': 43.65813540553308,
'lng': -79.35951549011845}],
'distance': 440,
'postalCode': 'M5A 3J9',
'cc': 'CA',
'neighborhood': 'Downtown Toronto',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['118 Sumach St (at Shuter St)',
'Toronto ON M5A 3J9',
'Canada']},
'categories': [{id: '4bf58dd8d48988d1e0931735',
'name': 'Coffee Shop',
'pluralName': 'Coffee Shops',
'shortName': 'Coffee Shop',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-5619551a498e9e35fce2256b-13'},
{'reasons': {'count': 0,
'items': [{summary: 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {id: '57cd9d20498e6ab8342980e2',
'name': 'Arvo',
'location': {address: '17 Gristmill Ln',
'crossStreet': 'at Parliament St',
'lat': 43.64996280366945,
'lng': -79.36144178325522,
```

```
'labeledLatLngs': [{label: 'display',
  'lat': 43.64996280366945,
  'lng': -79.36144178325522}],
'distance': 482,
'postalCode': 'M5A 3R6',
'cc': 'CA',
'neighborhood': 'Distillery',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['17 Gristmill Ln (at Parliament St)',
  'Toronto ON M5A 3R6',
  'Canada']],
'categories': [{id: '4bf58dd8d48988d1e0931735',
  'name': 'Coffee Shop',
  'pluralName': 'Coffee Shops',
  'shortName': 'Coffee Shop',
  'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
    'suffix': '.png'},
  'primary': True}],
'photos': {'count': 0, 'groups': []}},
'reerralId': 'e-0-57cd9d20498e6ab8342980e2-14'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '51853a73498e4d97a8b20831',
  'name': 'Rooster Coffee',
  'location': {'address': '343 King St E',
    'crossStreet': 'btwn Princess & Berkeley St',
    'lat': 43.65189965670432,
    'lng': -79.36560912104514,
    'labeledLatLngs': [{label: 'display',
      'lat': 43.65189965670432,
      'lng': -79.36560912104514}],
    'distance': 479,
    'postalCode': 'M5A 1L1',
    'cc': 'CA',
    'city': 'Toronto',
    'state': 'ON',
    'country': 'Canada',
    'formattedAddress': ['343 King St E (btwn Princess & Berkeley St)',
      'Toronto ON M5A 1L1',
      'Canada']},
    'categories': [{id: '4bf58dd8d48988d1e0931735',
      'name': 'Coffee Shop',
      'pluralName': 'Coffee Shops',
      'shortName': 'Coffee Shop',
      'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
        'suffix': '.png'},
      'primary': True}],
      'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-51853a73498e4d97a8b20831-15'},
      {'reasons': {'count': 0,
        'items': [{summary: 'This spot is popular',
```

```
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4b156a02f964a5207fac23e3',
'name': 'Brick Street Bakery',
'location': {'address': '27 Trinity St',
'crossStreet': 'in Distillery District',
'lat': 43.650574039683974,
'lng': -79.35953942981405,
'labeledLatLngs': [{'label': 'display',
'lat': 43.650574039683974,
'lng': -79.35953942981405}],
'distance': 419,
'postalCode': 'M5A 3C4',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['27 Trinity St (in Distillery District)',
'Toronto ON M5A 3C4',
'Canada']},
'categories': [{id: '4bf58dd8d48988d16a941735',
name: 'Bakery',
pluralName: 'Bakeries',
shortName: 'Bakery',
icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/bake
ry_',
'suffix': '.png'},
primary: True}],
'photos': {'count': 0, 'groups': []}},
'reerralId': 'e-0-4b156a02f964a5207fac23e3-16'},
{'reasons': {'count': 0,
'items': [{summary: 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '58c7fbf7424f9373e6427e99',
'name': 'Starbucks',
'location': {'address': '351 King St E, 60',
'lat': 43.651613,
'lng': -79.364917,
'labeledLatLngs': [{'label': 'display',
'lat': 43.651613,
'lng': -79.364917}],
'distance': 453,
'postalCode': 'M5A 1L1',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['351 King St E, 60',
'Toronto ON M5A 1L1',
'Canada']},
'categories': [{id: '4bf58dd8d48988d1e0931735',
name: 'Coffee Shop',
pluralName: 'Coffee Shops',
shortName: 'Coffee Shop',
icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coff
eshop_',
'suffix': '.png'},
primary: True}],
'photos': {'count': 0, 'groups': []}}]
```

```
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-58c7fbf7424f9373e6427e99-17'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4bb68165f562ef3b88483097',
'name': 'Young Centre for the Performing Arts',
'location': {'address': '50 Tank House Ln.',
'crossStreet': 'at Cherry St.',
'lat': 43.65082466432163,
'lng': -79.35759324240415,
'labeledLatLngs': [{'label': 'display',
'lat': 43.65082466432163,
'lng': -79.35759324240415}],
'distance': 454,
'postalCode': 'M5A 3C4',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['50 Tank House Ln. (at Cherry St.)',
'Toronto ON M5A 3C4',
'Canada']}},
'categories': [{id: '4bf58dd8d48988d1f2931735',
'name': 'Performing Arts Venue',
'pluralName': 'Performing Arts Venues',
'shortName': 'Performing Arts',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/arts_entertainment/performingarts_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4bb68165f562ef3b88483097-18'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4ddfbaca185035f3a44e8df6',
'name': 'Underpass Park',
'location': {'address': 'Eastern Ave.',
'crossStreet': 'Richmond St.',
'lat': 43.65576361726024,
'lng': -79.3548059463501,
'labeledLatLngs': [{'label': 'display',
'lat': 43.65576361726024,
'lng': -79.3548059463501}],
'distance': 498,
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['Eastern Ave. (Richmond St.)',
'Toronto ON',
'Canada']}},
```

```
'categories': [{id: '4bf58dd8d48988d163941735',
  name: 'Park',
  pluralName: 'Parks',
  shortName: 'Park',
  icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/parks_outdoors/park_',
    suffix: '.png'},
  primary: True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4ddfbaca185035f3a44e8df6-19'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    type: 'general',
    reasonName: 'globalInteractionReason'}]}},
'venue': {'id': '53a22c92498ec91fda7ce133',
  'name': 'Cluny Bistro & Boulangerie',
  'location': {'address': '35 Tank House Lane',
    'lat': 43.650565116074695,
    'lng': -79.35784287026658,
    'labeledLatLngs': [{'label': 'display',
      'lat': 43.650565116074695,
      'lng': -79.35784287026658}],
    'distance': 468,
    'postalCode': 'M5A 3C4',
    'cc': 'CA',
    'neighborhood': 'Distillery District, Toronto, ON',
    'city': 'Toronto',
    'state': 'ON',
    'country': 'Canada',
    'formattedAddress': ['35 Tank House Lane',
      'Toronto ON M5A 3C4',
      'Canada']}},
  'categories': [{id: '4bf58dd8d48988d10c941735',
    name: 'French Restaurant',
    pluralName: 'French Restaurants',
    shortName: 'French',
    icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/french_',
      suffix: '.png'},
    primary: True}],
  'photos': {'count': 0, 'groups': []},
  'venuePage': {'id': '199972479'},
  'referralId': 'e-0-53a22c92498ec91fda7ce133-20'},
  {'reasons': {'count': 0,
    'items': [{summary: 'This spot is popular',
      type: 'general',
      reasonName: 'globalInteractionReason'}]}},
  'venue': {'id': '5653a96f498e99c91027730b',
    'name': 'Cacao 70',
    'location': {'address': '28 Gristmill Lane',
      'lat': 43.650066694561666,
      'lng': -79.36072263183006,
      'labeledLatLngs': [{'label': 'display',
        'lat': 43.650066694561666,
        'lng': -79.36072263183006}],
      'distance': 466,
      'postalCode': 'M5A 3C4',
```

```
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['28 Gristmill Lane',
'Toronto ON M5A 3C4',
'Canada']],
'categories': [{id: '4bf58dd8d48988d1d0941735',
'name': 'Dessert Shop',
'pluralName': 'Dessert Shops',
'shortName': 'Desserts',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/dessert_',
'suffix': '.png'},
'primary': true}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-5653a96f498e99c91027730b-21'},
{'reasons': {'count': 0,
'items': [{summary: 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]},
'venue': {'id': '581258b738fa5bbef4c0857',
'name': 'Dark Horse Espresso Bar',
'location': {'address': '416 Front St E',
'crossStreet': 'Cooperage St',
'lat': 43.653080578844275,
'lng': -79.3570778621122,
'labeledLatLngs': [{'label': 'display',
'lat': 43.653080578844275,
'lng': -79.3570778621122}],
'distance': 315,
'cc': 'CA',
'neighborhood': 'Downtown Toronto',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['416 Front St E (Cooperage St)',
'Toronto ON',
'Canada']},
'categories': [{id: '4bf58dd8d48988d1e0931735',
'name': 'Coffee Shop',
'pluralName': 'Coffee Shops',
'shortName': 'Coffee Shop',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
'suffix': '.png'},
'primary': true}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-581258b738fa5bbef4c0857-22'},
{'reasons': {'count': 0,
'items': [{summary: 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]},
'venue': {'id': '4ad4c05df964a5204ef620e3',
'name': 'The Sweet Escape Patisserie',
'location': {'address': '55 Mill Street',
'lat': 43.65063217302609,
```

```
'lng': -79.35870913127346,
'labeledLatLngs': [{label': 'display',
  'lat': 43.65063217302609,
  'lng': -79.35870913127346}],
'distance': 432,
'postalCode': 'M5A 3C4',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['55 Mill Street',
  'Toronto ON M5A 3C4',
  'Canada']],
'categories': [{id': '4bf58dd8d48988d16a941735',
  'name': 'Bakery',
  'pluralName': 'Bakeries',
  'shortName': 'Bakery',
  'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/bake
ry_',
    'suffix': '.png'},
  'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4ad4c05df964a5204ef620e3-23'},
{'reasons': {'count': 0,
  'items': [{summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4c16a548955976b0cadea4f6',
  'name': 'Parliament Square Park',
  'location': {'address': '44 Parliament Street',
    'lat': 43.65026388338689,
    'lng': -79.36219509081177,
    'labeledLatLngs': [{label': 'display',
      'lat': 43.65026388338689,
      'lng': -79.36219509081177}],
    'distance': 462,
    'cc': 'CA',
    'city': 'Toronto',
    'state': 'ON',
    'country': 'Canada',
    'formattedAddress': ['44 Parliament Street', 'Toronto ON', 'Canad
a']},
  'categories': [{id': '4bf58dd8d48988d163941735',
    'name': 'Park',
    'pluralName': 'Parks',
    'shortName': 'Park',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/parks_out
doors/park_',
      'suffix': '.png'},
    'primary': True}],
    'photos': {'count': 0, 'groups': []}},
    'referralId': 'e-0-4c16a548955976b0cadea4f6-24'},
    {'reasons': {'count': 0,
      'items': [{summary': 'This spot is popular',
        'type': 'general',
        'reasonName': 'globalInteractionReason'}]}},
    'venue': {'id': '4b58dd55f964a5208f6f28e3',
```

```
'name': 'The Yoga Lounge',
'location': {'address': '106 Sherbourne St.',
'crossStreet': 'at Adelaide St. East',
'lat': 43.65551522261721,
'lng': -79.36495542526245,
'labeledLatLngs': [{"label": "display",
'lat': 43.65551522261721,
'lng': -79.36495542526245}],
'distance': 374,
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['106 Sherbourne St. (at Adelaide St. East)',
'Toronto ON',
'Canada']},
'categories': [{id: '4bf58dd8d48988d102941735',
'name': 'Yoga Studio',
'pluralName': 'Yoga Studios',
'shortName': 'Yoga Studio',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/gym
_yogastudio_',
'suffix': '.png'},
'primary': true}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4b58dd55f964a5208f6f28e3-25'},
{'reasons': {'count': 0,
'items': [{"summary": 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4d84d98181fdb1f7d4a704c0',
'name': 'Caffe Furbo',
'location': {'address': '12 case goods lane',
'lat': 43.649969882303814,
'lng': -79.35884946388191,
'labeledLatLngs': [{"label": "display",
'lat': 43.649969882303814,
'lng': -79.35884946388191}],
'distance': 498,
'postalCode': 'M5A 3C4',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['12 case goods lane',
'Toronto ON M5A 3C4',
'Canada']},
'categories': [{id: '4bf58dd8d48988d16d941735',
'name': 'Caf\u00e9',
'pluralName': 'Caf\u00e9s',
'shortName': 'Caf\u00e9',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/cafe
_',
'suffix': '.png'},
'primary': true}],
'photos': {'count': 0, 'groups': []},
'venuePage': {'id': '47611149'}}},
```

```
'referralId': 'e-0-4d84d98181fdb1f7d4a704c0-26'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4b8c46f3f964a5200cc832e3',
  'name': 'Alumnae Theatre',
  'location': {'address': '70 Berkley St.',
    'crossStreet': 'at Adelaide St.',
    'lat': 43.65275554626444,
    'lng': -79.36475283805089,
    'labeledLatLngs': [{"label": "display",
      'lat': 43.65275554626444,
      'lng': -79.36475283805089}],
    'distance': 371,
    'postalCode': 'M5A 2W6',
    'cc': 'CA',
    'city': 'Toronto',
    'state': 'ON',
    'country': 'Canada',
    'formattedAddress': ['70 Berkley St. (at Adelaide St.)',
      'Toronto ON M5A 2W6',
      'Canada']}},
  'categories': [{"id": "4bf58dd8d48988d137941735",
    'name': 'Theater',
    'pluralName': 'Theaters',
    'shortName': 'Theater',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/arts_entertainment/performingarts_theater_',
      'suffix': '.png'},
    'primary': True}],
    'photos': {'count': 0, 'groups': []}}},
'referralId': 'e-0-4b8c46f3f964a5200cc832e3-27'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4ade8ea8f964a5205a7621e3',
  'name': 'Berkeley Church',
  'location': {'address': '315 Queen St E',
    'crossStreet': 'at Berkeley St',
    'lat': 43.65512324174501,
    'lng': -79.36587330410705,
    'labeledLatLngs': [{"label": "display",
      'lat': 43.65512324174501,
      'lng': -79.36587330410705}],
    'distance': 432,
    'postalCode': 'M5A 1S7',
    'cc': 'CA',
    'city': 'Toronto',
    'state': 'ON',
    'country': 'Canada',
    'formattedAddress': ['315 Queen St E (at Berkeley St)',
      'Toronto ON M5A 1S7',
      'Canada']}},
  'categories': [{"id": "4bf58dd8d48988d171941735",
    'name': 'Event Space',
```

```
'pluralName': 'Event Spaces',
'shortName': 'Event Space',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/
eventspace_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4ade8ea8f964a5205a7621e3-28'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4bb52a8bf562ef3b7d992e97',
'name': 'Soulpepper Theatre',
'location': {'address': '55 Mill St',
'lat': 43.65078003430492,
'lng': -79.35761534068922,
'labeledLatLngs': [{'label': 'display',
'lat': 43.65078003430492,
'lng': -79.35761534068922}],
'distance': 457,
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['55 Mill St', 'Toronto ON', 'Canada']},
'categories': [{id: '4bf58dd8d48988d137941735',
'name': 'Theater',
'pluralName': 'Theaters',
'shortName': 'Theater',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/arts_ent
ertainment/performingarts_theater_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4bb52a8bf562ef3b7d992e97-29'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '54d37e23498e5e29bcb35362',
'name': 'ODIN Cafe + Bar',
'location': {'address': '514 King St E',
'crossStreet': 'at River St',
'lat': 43.656738544928054,
'lng': -79.35650305267754,
'labeledLatLngs': [{'label': 'display',
'lat': 43.656738544928054,
'lng': -79.35650305267754}],
'distance': 432,
'postalCode': 'M5A 1M1',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['514 King St E (at River St)',
'Toronto ON M5A 1M1',
```



```
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['Toronto ON M5A 1H7', 'Canada']},
'categories': [{id: '4bf58dd8d48988d11b941735',
  name: 'Pub',
  pluralName: 'Pubs',
  shortName: 'Pub',
  icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/nightlife/pub_',
    'suffix': '.png'},
  primary: True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-5b7cccd5e1f8ed6002c9bd7d0-32'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    type: 'general',
    reasonName: 'globalInteractionReason'}]}},
'venue': {id: '51ddece498e1ffd34185d2f',
  name: 'El Catrin',
  location: {address: '18 Tank House Lane',
    crossStreet: 'Distillery District',
    lat: 43.650600737116996,
    lng: -79.35892024942333,
    labeledLatLngs: [{'label': 'display',
      lat: 43.650600737116996,
      lng: -79.35892024942333}],
    distance: 430,
    postalCode: 'M5A 3C4',
    cc: 'CA',
    city: 'Toronto',
    state: 'ON',
    country: 'Canada',
    formattedAddress: ['18 Tank House Lane (Distillery District)',
      'Toronto ON M5A 3C4',
      'Canada']}},
  categories: [{id: '4bf58dd8d48988d1c1941735',
    name: 'Mexican Restaurant',
    pluralName: 'Mexican Restaurants',
    shortName: 'Mexican',
    icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/mexican_',
      'suffix': '.png'},
    primary: True}],
  photos: {'count': 0, 'groups': []},
  venuePage: {'id': '61086042'},
  referralId: 'e-0-51ddece498e1ffd34185d2f-33'},
  {'reasons': {'count': 0,
    'items': [{summary: 'This spot is popular',
      type: 'general',
      reasonName: 'globalInteractionReason'}]}},
  venue: {id: '4af9fc95f964a520ca1522e3',
    name: 'Mill St. Brew Pub',
    location: {address: '21 Tank House Ln',
      crossStreet: 'at Pure Spirits Mews',
      lat: 43.65035331843578,
      lng: -79.35848936650571,
```

```
'labeledLatLngs': [{label: 'display',
    'lat': 43.65035331843578,
    'lng': -79.35848936650571}],
'distance': 467,
'postalCode': 'M5A 3C4',
'cc': 'CA',
'neighborhood': 'Distillery District',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['21 Tank House Ln (at Pure Spirits Mews)',
    'Toronto ON M5A 3C4',
    'Canada']],
'categories': [{id: '4bf58dd8d48988d11b941735',
    'name': 'Pub',
    'pluralName': 'Pubs',
    'shortName': 'Pub',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/nightlife/pub_',
        'suffix': '.png'},
    'primary': true}],
'photos': {'count': 0, 'groups': []},
'referralId': 'e-0-4af9fc95f964a520ca1522e3-34'},
{'reasons': {'count': 0,
    'items': [{summary: 'This spot is popular',
        'type': 'general',
        'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4af21e78f964a520fae521e3',
    'name': 'Arta Gallery',
    'location': {'address': '14 Distillery Lane',
        'crossStreet': 'Distillery District',
        'lat': 43.650022181882065,
        'lng': -79.36122168476822,
        'labeledLatLngs': [{label: 'display',
            'lat': 43.650022181882065,
            'lng': -79.36122168476822}],
        'distance': 474,
        'postalCode': 'M5A 3C4',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['14 Distillery Lane (Distillery District)',
            'Toronto ON M5A 3C4',
            'Canada']},
        'categories': [{id: '4bf58dd8d48988d1e2931735',
            'name': 'Art Gallery',
            'pluralName': 'Art Galleries',
            'shortName': 'Art Gallery',
            'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/arts_entertainment/artgallery_',
                'suffix': '.png'},
            'primary': true}],
            'photos': {'count': 0, 'groups': []},
            'referralId': 'e-0-4af21e78f964a520fae521e3-35'},
            {'reasons': {'count': 0,
                'items': [{summary: 'This spot is popular',
```

```
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '59d00b871ffe9715e465071c',
'name': 'The Abnormal Beauty Company',
'location': {'address': '18 Distillery Lane',
'lat': 43.649892147380996,
'lng': -79.3610051085219,
'labeledLatLngs': [{'label': 'display',
'lat': 43.649892147380996,
'lng': -79.3610051085219}],
'distance': 487,
'postalCode': 'M5A 3C4',
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['18 Distillery Lane',
'Toronto ON M5A 3C4',
'Canada']},
'categories': [{id: '4bf58dd8d48988d10c951735',
name: 'Cosmetics Shop',
pluralName: 'Cosmetics Shops',
shortName: 'Cosmetics',
icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/beauty_cosmetic_',
'suffix': '.png'},
primary: True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-59d00b871ffe9715e465071c-36'},
{'reasons': {'count': 0,
'items': [{summary: 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '50e88fc6e4b007fcbb57aae8',
'name': 'Izumi',
'location': {'lat': 43.6499697935016,
{lng: -79.36015326451906,
'labeledLatLngs': [{'label': 'display',
'lat': 43.6499697935016,
'lng': -79.36015326451906}],
'distance': 479,
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['Toronto ON', 'Canada']},
'categories': [{id: '4bf58dd8d48988d142941735',
name: 'Asian Restaurant',
pluralName: 'Asian Restaurants',
shortName: 'Asian',
icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/asia_n_',
'suffix': '.png'},
primary: True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-50e88fc6e4b007fcbb57aae8-37'},
{'reasons': {'count': 0,
```

```
'items': [{  
    'summary': 'This spot is popular',  
    'type': 'general',  
    'reasonName': 'globalInteractionReason'}]}],  
'venue': {'id': '4ad8d551f964a5201f1521e3',  
    'name': 'Vistek',  
    'location': {'address': '496 Queen St E',  
        'crossStreet': 'at Sumach',  
        'lat': 43.65704640025721,  
        'lng': -79.35966695607046,  
        'labeledLatLngs': [{  
            'label': 'display',  
            'lat': 43.65704640025721,  
            'lng': -79.35966695607046}],  
    'distance': 319,  
    'cc': 'CA',  
    'city': 'Toronto',  
    'state': 'ON',  
    'country': 'Canada',  
    'formattedAddress': ['496 Queen St E (at Sumach)',  
        'Toronto ON',  
        'Canada']},  
    'categories': [{  
        'id': '4bf58dd8d48988d122951735',  
        'name': 'Electronics Store',  
        'pluralName': 'Electronics Stores',  
        'shortName': 'Electronics',  
        'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/tec  
hnology_',  
            'suffix': '.png'},  
        'primary': True}],  
    'photos': {'count': 0, 'groups': []}},  
    'referralId': 'e-0-4ad8d551f964a5201f1521e3-38'},  
    {'reasons': {  
        'count': 0,  
        'items': [{  
            'summary': 'This spot is popular',  
            'type': 'general',  
            'reasonName': 'globalInteractionReason'}]}],  
    'venue': {'id': '4b8acdfdf964a520378232e3',  
        'name': 'The Beer Store',  
        'location': {'address': '28 River Street',  
            'crossStreet': 'Queen St E',  
            'lat': 43.657773,  
            'lng': -79.3574632,  
            'labeledLatLngs': [{  
                'label': 'display',  
                'lat': 43.657773,  
                'lng': -79.3574632}],  
        'distance': 467,  
        'postalCode': 'M5A 3N9',  
        'cc': 'CA',  
        'city': 'Toronto',  
        'state': 'ON',  
        'country': 'Canada',  
        'formattedAddress': ['28 River Street (Queen St E)',  
            'Toronto ON M5A 3N9',  
            'Canada']},  
        'categories': [{  
            'id': '5370f356bcfc57f1066c94c2',  
            'name': 'Beer Store',  
            'pluralName': 'Beer Stores',  
            'shortName': 'Beer Store',  
            'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/nightli
```

```
e/beergarden_',
    'suffix': '.png'},
    'primary': True}],
    'photos': {'count': 0, 'groups': []}},
    'referralId': 'e-0-4b8acdfdf964a520378232e3-39'},
{'reasons': {'count': 0,
    'items': [{"summary": "This spot is popular",
        'type': 'general',
        'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4d126fd3d1848cfad5cfbb71',
    'name': 'TD Canada Trust',
    'location': {'address': '457 Front Street East',
        'lat': 43.652779762702764,
        'lng': -79.3563362625656,
        'labeledLatLngs': [{"label": 'display',
            'lat': 43.652779762702764,
            'lng': -79.3563362625656}],
        'distance': 383,
        'postalCode': 'M5A 0J2',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['457 Front Street East',
            'Toronto ON M5A 0J2',
            'Canada']}},
    'categories': [{"id': '4bf58dd8d48988d10a951735',
        'name': 'Bank',
        'pluralName': 'Banks',
        'shortName': 'Bank',
        'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/financial_',
            'suffix': '.png'},
        'primary': True}],
    'photos': {'count': 0, 'groups': []}},
    'referralId': 'e-0-4d126fd3d1848cfad5cfbb71-40'},
{'reasons': {'count': 0,
    'items': [{"summary": "This spot is popular",
        'type': 'general',
        'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4b2cf78cf964a52077cb24e3',
    'name': 'Savoury Grounds',
    'location': {'address': '481 Queen St. East',
        'crossStreet': 'Sumach St',
        'lat': 43.656820970004496,
        'lng': -79.3589698353747,
        'labeledLatLngs': [{"label": 'display',
            'lat': 43.656820970004496,
            'lng': -79.3589698353747}],
        'distance': 315,
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['481 Queen St. East (Sumach St)',
            'Toronto ON',
            'Canada']}},
```

```
'categories': [{id: '4bf58dd8d48988d1e0931735',
  name: 'Coffee Shop',
  pluralName: 'Coffee Shops',
  shortName: 'Coffee Shop',
  icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
    suffix: '.png'},
  primary: true}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4b2cf78cf964a52077cb24e3-41'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    type: 'general',
    reasonName: 'globalInteractionReason'}]}},
'venue': {'id': '51c085d3498eadedb67ba6cd',
  'name': 'Flame Shack',
  'location': {'address': '506 Queen St E',
    'crossStreet': 'Sumach St',
    'lat': 43.656844075440546,
    'lng': -79.35891727496157,
    'labeledLatLngs': [{label: 'display',
      lat: 43.656844075440546,
      lng: -79.35891727496157}],
    'distance': 319,
    'postalCode': 'M5A 1V2',
    'cc': 'CA',
    'city': 'Toronto',
    'state': 'ON',
    'country': 'Canada',
    'formattedAddress': ['506 Queen St E (Sumach St)',
      'Toronto ON M5A 1V2',
      'Canada']}},
  'categories': [{id: '4bf58dd8d48988d1c4941735',
    name: 'Restaurant',
    pluralName: 'Restaurants',
    shortName: 'Restaurant',
    icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/default_',
      suffix: '.png'},
    primary: true}],
  'photos': {'count': 0, 'groups': []}},
  'referralId': 'e-0-51c085d3498eadedb67ba6cd-42'},
  {'reasons': {'count': 0,
    'items': [{summary: 'This spot is popular',
      type: 'general',
      reasonName: 'globalInteractionReason'}]}},
  'venue': {'id': '57dd8bcd498e8eec19b9c896',
    'name': 'FUEL+',
    'location': {'address': '469 Front St',
      'crossStreet': 'Rolling Mills Rd',
      'lat': 43.653193,
      'lng': -79.355867,
      'labeledLatLngs': [{label: 'display',
        lat: 43.653193,
        lng: -79.355867}],
      'distance': 402,
      'postalCode': 'M5A',
```

```
'cc': 'CA',
'city': 'Toronto',
'state': 'ON',
'country': 'Canada',
'formattedAddress': ['469 Front St (Rolling Mills Rd)',  
    'Toronto ON M5A',
    'Canada']],
'categories': [{id: '4bf58dd8d48988d1e0931735',
    name: 'Coffee Shop',
    pluralName: 'Coffee Shops',
    shortName: 'Coffee Shop',
    icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coff  
eeshop_',
        'suffix': '.png'},
    primary: True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-57dd8bcd498e8eec19b9c896-43'},
{'reasons': {'count': 0,
    items: [{summary: 'This spot is popular',
        type: 'general',
        reasonName: 'globalInteractionReason'}]}},
'venue': {id: '579cc5f2498e0fe8397deebb',
    name: 'GW General',
    location: {'lat': 43.650495,
        'lng': -79.357538,
        labeledLatLngs: [{'label': 'display',
            'lat': 43.650495,
            'lng': -79.357538}],
        distance: 487,
        cc: 'CA',
        country: 'Canada',
        formattedAddress: ['Canada']}],
    categories: [{id: '4bf58dd8d48988d116951735',
        name: 'Antique Shop',
        pluralName: 'Antique Shops',
        shortName: 'Antiques',
        icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/ant  
ique_',
            'suffix': '.png'},
        primary: True}],
    photos: {'count': 0, 'groups': []}},
    'referralId': 'e-0-579cc5f2498e0fe8397deebb-44'}]}]}}
```

In [146]:

```
# function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']
```

In [147]:

```
venues = results['response'][‘groups’][0][‘items’]

nearby_venues = json_normalize(venues) # flatten JSON
# filter columns
filtered_columns = [‘venue.name’, ‘venue.categories’, ‘venue.location.lat’, ‘venue.location.lng’]
nearby_venues =nearby_venues.loc[:, filtered_columns]
# filter the category for each row
nearby_venues[‘venue.categories’] = nearby_venues.apply(get_category_type, axis=1)
# clean columns
nearby_venues.columns = [col.split(“.”)[-1] for col in nearby_venues.columns]
nearby_venues.head()
```

/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/ipykernel_launcher.py:3: FutureWarning: pandas.io.json.json_normalize is deprecated, use pandas.json_normalize instead

This is separate from the ipykernel package so we can avoid doing imports until

Out[147]:

	name	categories	lat	lng
0	Tandem Coffee	Coffee Shop	43.653559	-79.361809
1	Roselle Desserts	Bakery	43.653447	-79.362017
2	Cooper Koo Family YMCA	Distribution Center	43.653249	-79.358008
3	Body Blitz Spa East	Spa	43.654735	-79.359874
4	Impact Kitchen	Restaurant	43.656369	-79.356980

Explore Neighborhoods in Downtown Toronto

Let's create a function to repeat the same process to all the neighborhoods in Downtown Toronto

In [148]:

```
def getNearbyVenues(names, latitudes, longitudes, radius=500):  
  
    venues_list=[]  
    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_list.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_list in venues_list for item in venue_list  
])  
    nearby_venues.columns = ['Neighbourhood',  
                            'Neighbourhood Latitude',  
                            'Neighbourhood Longitude',  
                            'Venue',  
                            'Venue Latitude',  
                            'Venue Longitude',  
                            'Venue Category']  
  
    return(nearby_venues)
```

In [149]:

```
# type your answer here
DowntownToronto_venues = getNearbyVenues(names=DowntownToronto_data['Neighbourhood'],latitudes=DowntownToronto_data['Latitude'],longitudes=DowntownToronto_data['Longitude'])
```

Regent Park, Harbourfront
Queen's Park, Ontario Provincial Government
Garden District, Ryerson
St. James Town
Berczy Park
Central Bay Street
Christie
Richmond, Adelaide, King
Harbourfront East, Union Station, Toronto Islands
Toronto Dominion Centre, Design Exchange
Commerce Court, Victoria Hotel
University of Toronto, Harbord
Kensington Market, Chinatown, Grange Park
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay,
South Niagara, Island airport
Rosedale
Stn A PO Boxes
St. James Town, Cabbagetown
First Canadian Place, Underground city
Church and Wellesley

In [150]:

```
print(DowntownToronto_venues.shape)
DowntownToronto_venues.head(20)
```

(1210, 7)

Out[150]:

	Neighbourhood	Neighbourhood Latitude	Neighbourhood Longitude	Venue	Venue Latitude	Venue Longitude	Cat
0	Regent Park, Harbourfront	43.65426	-79.360636	Tandem Coffee	43.653559	-79.361809	'
1	Regent Park, Harbourfront	43.65426	-79.360636	Roselle Desserts	43.653447	-79.362017	E
2	Regent Park, Harbourfront	43.65426	-79.360636	Cooper Koo Family YMCA	43.653249	-79.358008	Distri ('
3	Regent Park, Harbourfront	43.65426	-79.360636	Body Blitz Spa East	43.654735	-79.359874	
4	Regent Park, Harbourfront	43.65426	-79.360636	Impact Kitchen	43.656369	-79.356980	Rest
5	Regent Park, Harbourfront	43.65426	-79.360636	Dominion Pub and Kitchen	43.656919	-79.358967	
6	Regent Park, Harbourfront	43.65426	-79.360636	Morning Glory Cafe	43.653947	-79.361149	Bre
7	Regent Park, Harbourfront	43.65426	-79.360636	Corktown Common	43.655618	-79.356211	
8	Regent Park, Harbourfront	43.65426	-79.360636	The Extension Room	43.653313	-79.359725	F ('
9	Regent Park, Harbourfront	43.65426	-79.360636	The Distillery Historic District	43.650244	-79.359323	H
10	Regent Park, Harbourfront	43.65426	-79.360636	Distillery Sunday Market	43.650075	-79.361832	Fee ('
11	Regent Park, Harbourfront	43.65426	-79.360636	SOMA chocolatemaker	43.650622	-79.358127	Cho
12	Regent Park, Harbourfront	43.65426	-79.360636	Figs Breakfast & Lunch	43.655675	-79.364503	Bre
13	Regent Park, Harbourfront	43.65426	-79.360636	Sumach Espresso	43.658135	-79.359515	'
14	Regent Park, Harbourfront	43.65426	-79.360636	Arvo	43.649963	-79.361442	'
15	Regent Park, Harbourfront	43.65426	-79.360636	Rooster Coffee	43.651900	-79.365609	'
16	Regent Park, Harbourfront	43.65426	-79.360636	Brick Street Bakery	43.650574	-79.359539	E
17	Regent Park, Harbourfront	43.65426	-79.360636	Starbucks	43.651613	-79.364917	'
18	Regent Park, Harbourfront	43.65426	-79.360636	Young Centre for the Performing Arts	43.650825	-79.357593	Perf Arts'
19	Regent Park, Harbourfront	43.65426	-79.360636	Underpass Park	43.655764	-79.354806	

In [151]:

```
DowntownToronto_venues.groupby('Neighbourhood').count()
```

Out[151]:

Neighbourhood	Neighbourhood Latitude	Neighbourhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Berczy Park	55	55	55	55	55	55
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport	18	18	18	18	18	18
Central Bay Street	61	61	61	61	61	61
Christie	16	16	16	16	16	16
Church and Wellesley	79	79	79	79	79	79
Commerce Court, Victoria Hotel	100	100	100	100	100	100
First Canadian Place, Underground city	100	100	100	100	100	100
Garden District, Ryerson	100	100	100	100	100	100
Harbourfront East, Union Station, Toronto Islands	100	100	100	100	100	100
Kensington Market, Chinatown, Grange Park	59	59	59	59	59	59
Queen's Park, Ontario Provincial Government	31	31	31	31	31	31
Regent Park, Harbourfront	45	45	45	45	45	45
Richmond, Adelaide, King	93	93	93	93	93	93
Rosedale	4	4	4	4	4	4
St. James Town	79	79	79	79	79	79
St. James Town, Cabbagetown	43	43	43	43	43	43
Stn A PO Boxes	97	97	97	97	97	97
Toronto Dominion Centre, Design Exchange	100	100	100	100	100	100
University of Toronto, Harbord	30	30	30	30	30	30

In [152]:

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

```

There are 206 uniques categories.

Analyze Each Neighborhood

In [153]:

```
# one hot encoding
DowntownToronto_onehot = pd.get_dummies(DowntownToronto_venues[['Venue Category']], prefix = "", prefix_sep = "")

# add neighbourhood column back to dataframe
DowntownToronto_onehot['Neighbourhood'] = DowntownToronto_venues['Neighbourhood']

# move neighbourhood column to the first column
fixed_columns = [DowntownToronto_onehot.columns[-1]] + list(DowntownToronto_onehot.columns[:-1])
DowntownToronto_onehot = DowntownToronto_onehot[fixed_columns]
DowntownToronto_onehot.head()
```

Out[153]:

	Neighbourhood	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	A
0	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0
1	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0
2	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0
3	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0
4	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0



In [154]:

```
DowntownToronto_onehot.shape
```

Out[154]:

(1210, 207)

Next, let's group rows by neighborhood and by taking the mean of the frequency of occurrence of each category

In [155]:

```
DowntownToronto_grouped = DowntownToronto_onehot.groupby('Neighbourhood').mean().reset_index()
DowntownToronto_grouped
```

Out[155]:

	Neighbourhood	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	Ame Resta
0	Berczy Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
1	CN Tower, King and Spadina, Railway Lands, Har...	0.000000	0.055556	0.055556	0.055556	0.111111	0.166667	0.111111	0.0
2	Central Bay Street	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
3	Christie	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
4	Church and Wellesley	0.012658	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
5	Commerce Court, Victoria Hotel	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
6	First Canadian Place, Underground city	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
7	Garden District, Ryerson	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
8	Harbourfront East, Union Station, Toronto Islands	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
9	Kensington Market, Chinatown, Grange Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
10	Queen's Park, Ontario Provincial Government	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
11	Regent Park, Harbourfront	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
12	Richmond, Adelaide, King	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
13	Rosedale	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
14	St. James Town	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
15	St. James Town, Cabbagetown	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
16	Stn A PO Boxes	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
17	Toronto Dominion Centre, Design Exchange	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0

	Neighbourhood	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	Ame Resta
18	University of Toronto, Harbord	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0

◀ ▶

In [156]:

```
DowntownToronto_grouped.shape
```

Out[156]:

```
(19, 207)
```

Let's print each neighborhood along with the top 5 most common venues

In [157]:

```
num_top_venues = 5
for hood in DowntownToronto_grouped['Neighbourhood']:
    print("----"+hood+"----")
    temp = DowntownToronto_grouped[DowntownToronto_grouped['Neighbourhood'] == hood].T.reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_top_venues))
    print('\n')
```

----Berczy Park----

	venue	freq
0	Coffee Shop	0.09
1	Cocktail Bar	0.05
2	Bakery	0.04
3	Cheese Shop	0.04
4	Seafood Restaurant	0.04

----CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport----

	venue	freq
0	Airport Service	0.17
1	Airport Lounge	0.11
2	Airport Terminal	0.11
3	Boat or Ferry	0.06
4	Airport	0.06

----Central Bay Street----

	venue	freq
0	Coffee Shop	0.18
1	Italian Restaurant	0.05
2	Sandwich Place	0.05
3	Café	0.05
4	Salad Place	0.03

----Christie----

	venue	freq
0	Grocery Store	0.25
1	Café	0.19
2	Park	0.12
3	Nightclub	0.06
4	Candy Store	0.06

----Church and Wellesley----

	venue	freq
0	Coffee Shop	0.09
1	Sushi Restaurant	0.06
2	Japanese Restaurant	0.06
3	Restaurant	0.04
4	Gay Bar	0.04

----Commerce Court, Victoria Hotel----

	venue	freq
0	Coffee Shop	0.12
1	Restaurant	0.07
2	Café	0.06
3	Hotel	0.06
4	Gym	0.04

----First Canadian Place, Underground city----

	venue	freq

0	Coffee Shop	0.11
1	Café	0.07
2	Hotel	0.06
3	Gym	0.04
4	Japanese Restaurant	0.04

----Garden District, Ryerson----

	venue	freq
0	Clothing Store	0.09
1	Coffee Shop	0.09
2	Café	0.03
3	Middle Eastern Restaurant	0.03
4	Italian Restaurant	0.03

----Harbourfront East, Union Station, Toronto Islands----

	venue	freq
0	Coffee Shop	0.12
1	Aquarium	0.05
2	Café	0.04
3	Hotel	0.04
4	Brewery	0.03

----Kensington Market, Chinatown, Grange Park----

	venue	freq
0	Café	0.08
1	Coffee Shop	0.07
2	Vegetarian / Vegan Restaurant	0.07
3	Vietnamese Restaurant	0.05
4	Mexican Restaurant	0.05

----Queen's Park, Ontario Provincial Government----

	venue	freq
0	Coffee Shop	0.26
1	Sushi Restaurant	0.06
2	Gym	0.03
3	Fried Chicken Joint	0.03
4	Café	0.03

----Regent Park, Harbourfront----

	venue	freq
0	Coffee Shop	0.18
1	Park	0.07
2	Bakery	0.07
3	Pub	0.07
4	Breakfast Spot	0.04

----Richmond, Adelaide, King----

	venue	freq
0	Coffee Shop	0.11
1	Café	0.05
2	Restaurant	0.04

3	Gym	0.03
4	Thai Restaurant	0.03

----Rosedale----

	venue	freq
0	Park	0.50
1	Playground	0.25
2	Trail	0.25
3	Museum	0.00
4	Mediterranean Restaurant	0.00

----St. James Town----

	venue	freq
0	Café	0.06
1	Coffee Shop	0.06
2	Cocktail Bar	0.05
3	American Restaurant	0.04
4	Gastropub	0.04

----St. James Town, Cabbagetown----

	venue	freq
0	Coffee Shop	0.07
1	Restaurant	0.05
2	Pizza Place	0.05
3	Pub	0.05
4	Italian Restaurant	0.05

----Stn A PO Boxes----

	venue	freq
0	Coffee Shop	0.11
1	Seafood Restaurant	0.04
2	Café	0.03
3	Cocktail Bar	0.03
4	Hotel	0.03

----Toronto Dominion Centre, Design Exchange----

	venue	freq
0	Coffee Shop	0.11
1	Hotel	0.09
2	Café	0.06
3	Italian Restaurant	0.03
4	American Restaurant	0.03

----University of Toronto, Harbord----

	venue	freq
0	Café	0.17
1	Bar	0.07
2	Japanese Restaurant	0.07
3	Bookstore	0.07
4	Bakery	0.07

In [158]:

```
# Create dataframe for individual neighborhoods
df_berczypark = pd.DataFrame(np.array([[0, 'Coffee Shop', 0.09], [1, 'Cocktail Bar', 0.05], [2, 'Bakery', 0.04], [3, 'Cheese Shop', 0.04], [4, 'Seafood Restaurant', 0.04]]), columns=['Berczy Park', 'venue', 'freq'])
df_berczypark
```

Out[158]:

Berczy Park	venue	freq
0	Coffee Shop	0.09
1	Cocktail Bar	0.05
2	Bakery	0.04
3	Cheese Shop	0.04
4	Seafood Restaurant	0.04

In [159]:

```
df_cbay = pd.DataFrame(np.array([[0, 'Coffee Shop', 0.18], [1, 'Italian Restaurant', 0.05], [2, 'Sandwich Place', 0.05], [3, 'Café', 0.05], [4, 'Salad Place', 0.03]]), columns=['Central Bay Street', 'venue', 'freq'])
df_cbay
```

Out[159]:

Central Bay Street	venue	freq
0	Coffee Shop	0.18
1	Italian Restaurant	0.05
2	Sandwich Place	0.05
3	Café	0.05
4	Salad Place	0.03

In [160]:

```
df_christie = pd.DataFrame(np.array([[0, 'Grocery Store', 0.25], [1, 'Cafe', 0.19], [2, 'Park', 0.12], [3, 'Nightclub', 0.06], [4, 'Candy Store', 0.06]]), columns=['Christie', 'venue', 'freq'])
df_christie
```

Out[160]:

	Christie	venue	freq
0	0	Grocery Store	0.25
1	1	Cafe	0.19
2	2	Park	0.12
3	3	Nightclub	0.06
4	4	Candy Store	0.06

In [161]:

```
df_churchwell = pd.DataFrame(np.array([[0, 'Coffee Shop', 0.09], [1, 'Sushi Restaurant', 0.19], [2, 'Japanese Restaurant', 0.06], [3, 'Restaurant', 0.06], [4, 'Gay Bar', 0.06]]), columns=['Church and Wellesley', 'venue', 'freq'])
df_churchwell
```

Out[161]:

	Church and Wellesley	venue	freq
0	0	Coffee Shop	0.09
1	1	Sushi Restaurant	0.19
2	2	Japanese Restaurant	0.06
3	3	Restaurant	0.06
4	4	Gay Bar	0.06

In [162]:

```
df_commerce = pd.DataFrame(np.array([[0, 'Coffee Shop', 0.12], [1, 'Restaurant', 0.07], [2, 'Cafe', 0.06], [3, 'Hotel', 0.06], [4, 'Gym', 0.04]]), columns=['Commerce Court', 'venue', 'freq'])
df_commerce
```

Out[162]:

	Commerce Court	venue	freq
0	0	Coffee Shop	0.12
1	1	Restaurant	0.07
2	2	Cafe	0.06
3	3	Hotel	0.06
4	4	Gym	0.04

In [163]:

```
import matplotlib.pyplot as plt

df_berczypark['freq'] = df_berczypark['freq'].astype(float)
print (df_berczypark)
print (df_berczypark.dtypes)

df_cbay['freq'] = df_cbay['freq'].astype(float)
print (df_cbay)
print (df_cbay.dtypes)

df_christie['freq'] = df_christie['freq'].astype(float)
print (df_christie)
print (df_christie.dtypes)

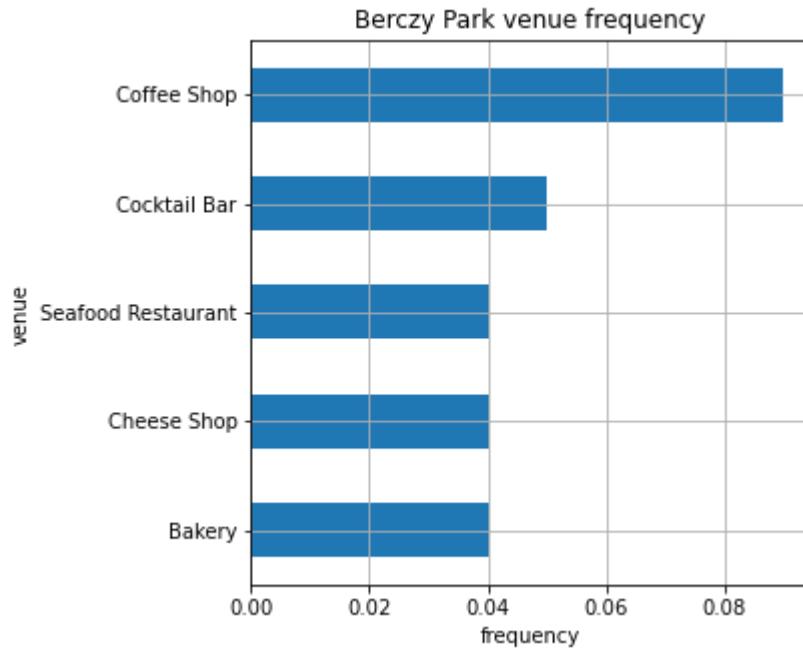
df_churchwell['freq'] = df_churchwell['freq'].astype(float)
print (df_churchwell)
print (df_churchwell.dtypes)

df_commerce['freq'] = df_commerce['freq'].astype(float)
print (df_commerce)
print (df_commerce.dtypes)
```

```
Berczy Park           venue  freq
0                 0   Coffee Shop  0.09
1                 1   Cocktail Bar 0.05
2                 2      Bakery  0.04
3                 3 Cheese Shop  0.04
4                 4 Seafood Restaurant 0.04
Berczy Park      object
venue            object
freq             float64
dtype: object
Central Bay Street           venue  freq
0                  0   Coffee Shop  0.18
1                  1 Italian Restaurant 0.05
2                  2 Sandwich Place 0.05
3                  3      Café  0.05
4                  4 Salad Place  0.03
Central Bay Street      object
venue            object
freq             float64
dtype: object
Christie           venue  freq
0                 0 Grocery Store  0.25
1                 1      Cafe  0.19
2                 2      Park  0.12
3                 3 Nightclub  0.06
4                 4 Candy Store  0.06
Christie      object
venue            object
freq             float64
dtype: object
Church and Wellesley           venue  freq
0                  0   Coffee Shop  0.09
1                  1 Sushi Restaurant 0.19
2                  2 Japanese Restaurant 0.06
3                  3      Restaurant 0.06
4                  4      Gay Bar  0.06
Church and Wellesley      object
venue            object
freq             float64
dtype: object
Commerce Court           venue  freq
0                 0 Coffee Shop  0.12
1                 1 Restaurant  0.07
2                 2      Cafe  0.06
3                 3      Hotel  0.06
4                 4      Gym  0.04
Commerce Court      object
venue            object
freq             float64
dtype: object
```

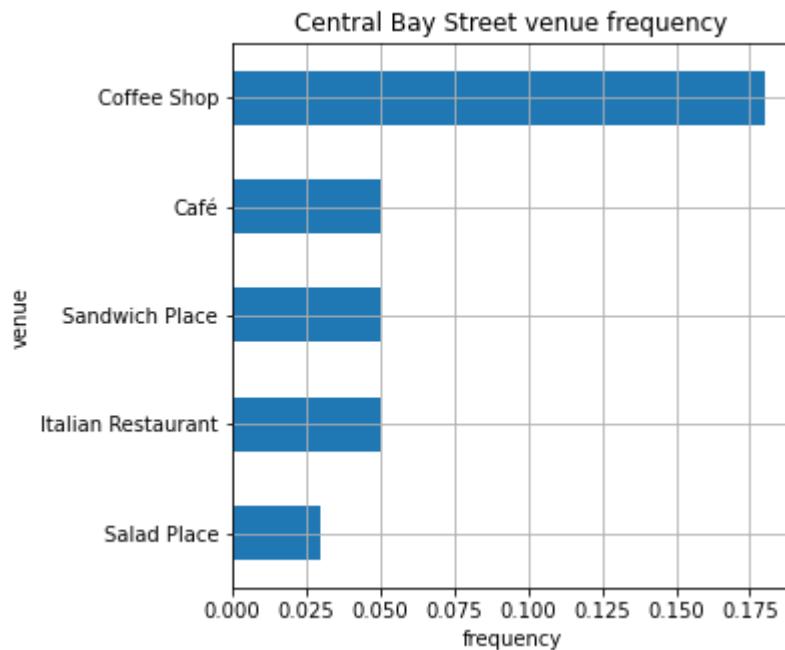
In [164]:

```
df_berczypark.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('frequency')
plt.title('Berczy Park venue frequency')
plt.show()
```



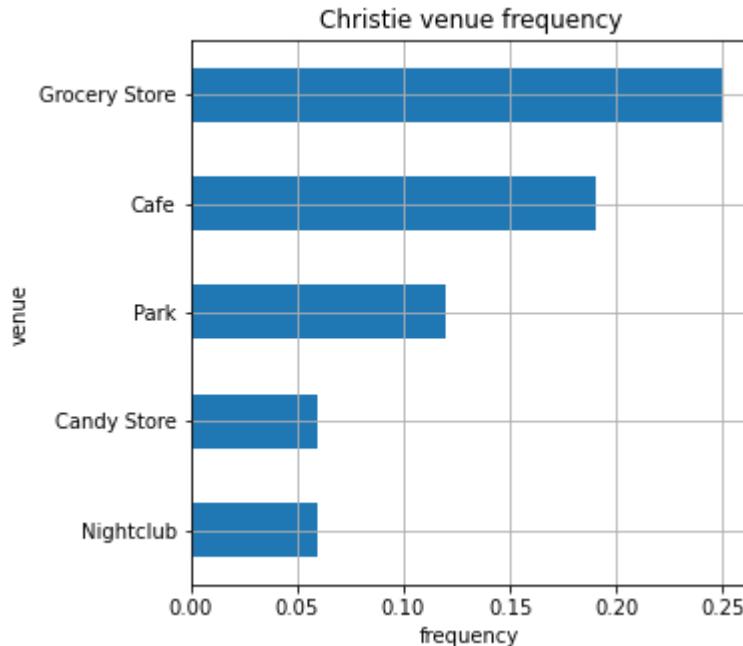
In [165]:

```
df_cbay.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('frequency')
plt.title('Central Bay Street venue frequency')
plt.show()
```



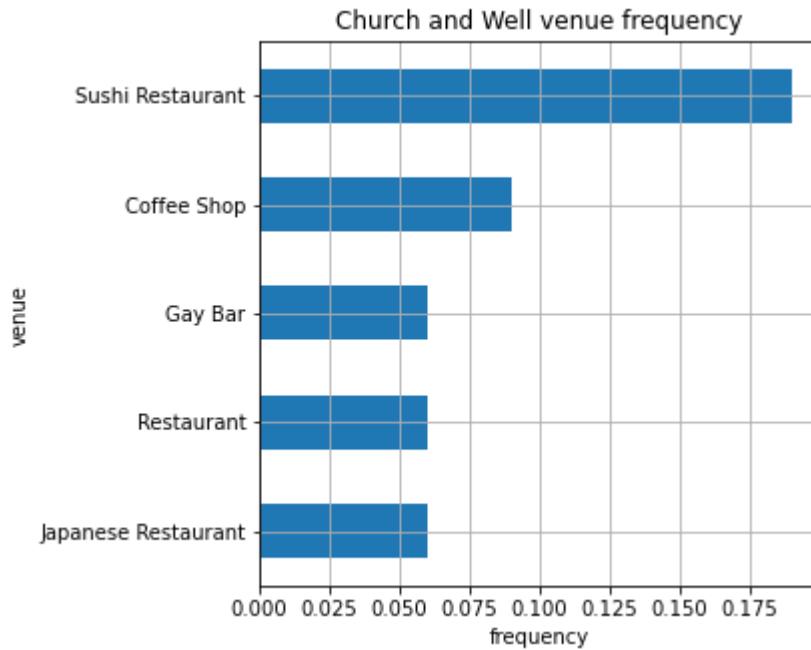
In [166]:

```
df_christie.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('frequency')
plt.title('Christie venue frequency')
plt.show()
```



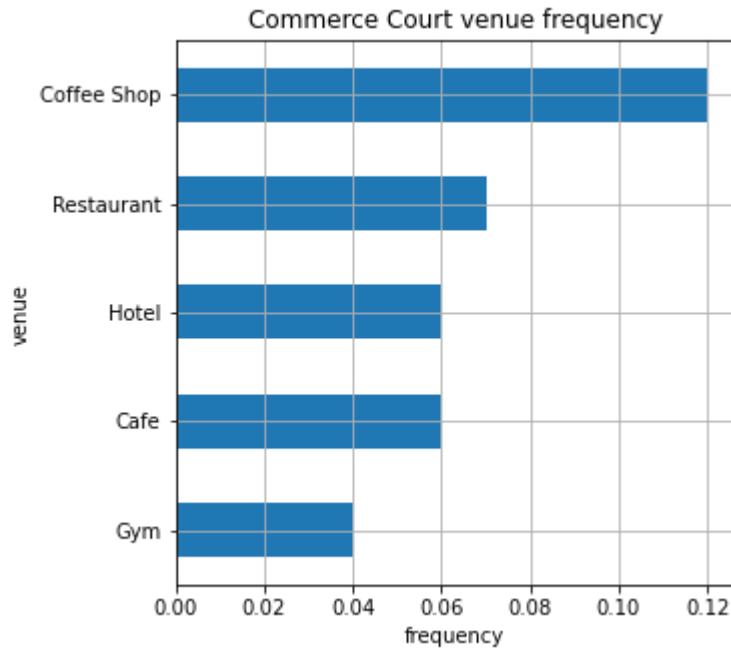
In [167]:

```
df_churchwell.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('frequency')
plt.title('Church and Well venue frequency')
plt.show()
```



In [168]:

```
df_commerce.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('frequency')
plt.title('Commerce Court venue frequency')
plt.show()
```



Let's put that into a pandas dataframe

In [169]:

```
# Let's write a function to sort the venues in descending order.
def return_most_common_venues(row, num_top_venues):
    row_categories = row.iloc[1:]
    row_categories_sorted = row_categories.sort_values(ascending=False)

    return row_categories_sorted.index.values[0:num_top_venues]
```

In [170]:

```

num_top_venues = 10
indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['Neighbourhood']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{}{} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
neighbourhoods_venues_sorted = pd.DataFrame(columns=columns)
neighbourhoods_venues_sorted['Neighbourhood'] = DowntownToronto_grouped['Neighbourhood']
for ind in np.arange(DowntownToronto_grouped.shape[0]):
    neighbourhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(DowntownToronto_grouped.iloc[ind, :], num_top_venues)
neighbourhoods_venues_sorted.head()

```

Out[170]:

	Neighbourhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
0	Berczy Park	Coffee Shop	Cocktail Bar	Seafood Restaurant	Bakery	Beer Bar	Cheese Shop	Farmers Market
1	CN Tower, King and Spadina, Railway Lands, Har...	Airport Service	Airport Lounge	Airport Terminal	Boutique	Boat or Ferry	Plane	Rental Car Location
2	Central Bay Street	Coffee Shop	Sandwich Place	Italian Restaurant	Café	Bubble Tea Shop	Salad Place	Burger Joint
3	Christie	Grocery Store	Café	Park	Athletics & Sports	Italian Restaurant	Nightclub	Candy Store
4	Church and Wellesley	Coffee Shop	Japanese Restaurant	Sushi Restaurant	Restaurant	Gay Bar	Yoga Studio	Men's Store

Cluster Neighborhoods

Run k-means to cluster the neighborhood into 5 clusters

In [171]:

```
# set number of clusters
kclusters = 5
DowntownToronto_grouped_clustering = DowntownToronto_grouped.drop('Neighbourhood', 1)
# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(DowntownToronto_grouped_clustering)
# check cluster Labels generated for each row in the dataframe
kmeans.labels_[0:10]
```

Out[171]:

```
array([0, 3, 4, 2, 0, 0, 0, 0, 0], dtype=int32)
```

Let's create a new dataframe that includes the cluster as well as the top 10 venues for each neighborhood.

In [172]:

```
# add clustering Labels
neighbourhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)
DowntownToronto_merged = DowntownToronto_data
# merge Downtown_Toronto_grouped with Downtown_Toronto_data to add Latitude/Longitude for
# each neighborhood
DowntownToronto_merged = DowntownToronto_merged.join(neighbourhoods_venues_sorted.set_index(
    'Neighbourhood'), on='Neighbourhood')
DowntownToronto_merged.head() # check the last columns!
```

Out[172]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Con
0	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636	4	Coffee Shop	Bakery	
1	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494	4	Coffee Shop	Sushi Restaurant	\$
2	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937	0	Clothing Store	Coffee Shop	Cosmopolitan
3	M5C	Downtown Toronto	St. James Town	43.651494	-79.375418	0	Café	Coffee Shop	Coffee
4	M5E	Downtown Toronto	Berczy Park	43.644771	-79.373306	0	Coffee Shop	Cocktail Bar	Se Resta

Finally, let's visualize the resulting clusters

In [173]:

```
# create map
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(DowntownToronto_merged['Latitude'], DowntownToronto_merged['Longitude'], DowntownToronto_merged['Neighbourhood'], DowntownToronto_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[173]:

Make this Notebook Trusted to load map: File -> Trust Notebook

Examine Clusters

Cluster 1

In [174]:

```
DowntownToronto_merged.loc[DowntownToronto_merged['Cluster Labels'] == 0, DowntownToronto_
merged.columns[[2] + list(range(5, DowntownToronto_merged.shape[1]))]]
```

Out[174]:

	Neighbourhood	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue
2	Garden District, Ryerson	0	Clothing Store	Coffee Shop	Cosmetics Shop	Bubble Tea Shop	Café	Middle Eastern Restaurant
3	St. James Town	0	Café	Coffee Shop	Cocktail Bar	American Restaurant	Gastropub	Bakery
4	Berczy Park	0	Coffee Shop	Cocktail Bar	Seafood Restaurant	Bakery	Beer Bar	Cheese Shop
7	Richmond, Adelaide, King	0	Coffee Shop	Café	Restaurant	Gym	Deli / Bodega	Thai Restaurant
8	Harbourfront East, Union Station, Toronto Islands	0	Coffee Shop	Aquarium	Hotel	Café	Brewery	Restaurant
9	Toronto Dominion Centre, Design Exchange	0	Coffee Shop	Hotel	Café	Restaurant	Salad Place	Seafood Restaurant
10	Commerce Court, Victoria Hotel	0	Coffee Shop	Restaurant	Hotel	Café	American Restaurant	Italian Restaurant
11	University of Toronto, Harbord	0	Café	Bakery	Bookstore	Japanese Restaurant	Bar	Yoga Studio
12	Kensington Market, Chinatown, Grange Park	0	Café	Vegetarian / Vegan Restaurant	Coffee Shop	Vietnamese Restaurant	Mexican Restaurant	Grocery Store
15	Stn A PO Boxes	0	Coffee Shop	Seafood Restaurant	Hotel	Italian Restaurant	Restaurant	Beer Bar
16	St. James Town, Cabbagetown	0	Coffee Shop	Restaurant	Pub	Pizza Place	Italian Restaurant	Bakery
17	First Canadian Place, Underground city	0	Coffee Shop	Café	Hotel	Japanese Restaurant	Restaurant	Gym
18	Church and Wellesley	0	Coffee Shop	Japanese Restaurant	Sushi Restaurant	Restaurant	Gay Bar	Yoga Studio

Cluster 2

In [175]:

```
DowntownToronto_merged.loc[DowntownToronto_merged['Cluster Labels'] == 1, DowntownToronto_
merged.columns[[2] + list(range(5, DowntownToronto_merged.shape[1]))]]
```

Out[175]:

Neighbourhood	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th M
14	Rosedale	1	Park	Trail	Playground	Creperie	Doner Restaurant	Dog Run

Cluster 3

In [176]:

```
DowntownToronto_merged.loc[DowntownToronto_merged['Cluster Labels'] == 2, DowntownToronto_
merged.columns[[2] + list(range(5, DowntownToronto_merged.shape[1]))]]
```

Out[176]:

Neighbourhood	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th M
6	Christie	2	Grocery Store	Café	Park	Athletics & Sports	Italian Restaurant	Nightclub

Cluster 4

In [177]:

```
DowntownToronto_merged.loc[DowntownToronto_merged['Cluster Labels'] == 3, DowntownToronto_
merged.columns[[2] + list(range(5, DowntownToronto_merged.shape[1]))]]
```

Out[177]:

Neighbourhood	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th M
13	CN Tower, King and Spadina, Railway Lands, Har...	3	Airport Service	Airport Lounge	Airport Terminal	Boutique	Boat or Ferry	Plane

Cluster 5

In [178]:

```
DowntownToronto_merged.loc[DowntownToronto_merged['Cluster Labels'] == 4, DowntownToronto_
merged.columns[[2] + list(range(5, DowntownToronto_merged.shape[1]))]]
```

Out[178]:

Neighbourhood	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
Regent Park, Harbourfront	4	Coffee Shop	Bakery	Pub	Park	Restaurant	Breakfast Spot	...
Queen's Park, Ontario Provincial Government	4	Coffee Shop	Sushi Restaurant	Yoga Studio	Diner	Smoothie Shop	Italian Restaurant	J Re...
Central Bay Street	4	Coffee Shop	Sandwich Place	Italian Restaurant	Café	Bubble Tea Shop	Salad Place	...

The above data was meant for Toronto. Preparing similar data for New York City

Getting new york data

In [179]:

```
import json # Library to handle JSON files

import requests # Library to handle requests
from pandas.io.json import json_normalize # transform JSON file into a pandas dataframe
```

In [180]:

```
!wget -q -O 'newyork_data.json' https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDriverSkillsNetwork-DS0701EN-SkillsNetwork/labs/newyork_data.json
print('Data downloaded!')
```

Data downloaded!

In [181]:

```
with open('newyork_data.json') as json_data:
    newyork_data = json.load(json_data)
```

In [182]:

```
newyork_data
```

Out[182]:

```
{'type': 'FeatureCollection',
'totalFeatures': 306,
'features': [{ 'type': 'Feature',
'id': 'nyu_2451_34572.1',
'geometry': { 'type': 'Point',
'coordinates': [-73.84720052054902, 40.89470517661]},
'geometry_name': 'geom',
'properties': { 'name': 'Wakefield',
'stacked': 1,
'annoline1': 'Wakefield',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.84720052054902,
40.89470517661,
-73.84720052054902,
40.89470517661]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.2',
'geometry': { 'type': 'Point',
'coordinates': [-73.82993910812398, 40.87429419303012]},
'geometry_name': 'geom',
'properties': { 'name': 'Co-op City',
'stacked': 2,
'annoline1': 'Co-op',
'annoline2': 'City',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.82993910812398,
40.87429419303012,
-73.82993910812398,
40.87429419303012]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.3',
'geometry': { 'type': 'Point',
'coordinates': [-73.82780644716412, 40.887555677350775]},
'geometry_name': 'geom',
'properties': { 'name': 'Eastchester',
'stacked': 1,
'annoline1': 'Eastchester',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.82780644716412,
40.887555677350775,
-73.82780644716412,
40.887555677350775]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.4',
'geometry': { 'type': 'Point',
'coordinates': [-73.90564259591682, 40.89543742690383]},
'geometry_name': 'geom',
```

```
'properties': {'name': 'Fieldston',
 'stacked': 1,
 'annoline1': 'Fieldston',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
 'bbox': [-73.90564259591682,
 40.89543742690383,
 -73.90564259591682,
 40.89543742690383]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.5',
 'geometry': {'type': 'Point',
 'coordinates': [-73.9125854610857, 40.890834493891305]},
 'geometry_name': 'geom',
 'properties': {'name': 'Riverdale',
 'stacked': 1,
 'annoline1': 'Riverdale',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
 'bbox': [-73.9125854610857,
 40.890834493891305,
 -73.9125854610857,
 40.890834493891305]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.6',
 'geometry': {'type': 'Point',
 'coordinates': [-73.90281798724604, 40.88168737120521]},
 'geometry_name': 'geom',
 'properties': {'name': 'Kingsbridge',
 'stacked': 1,
 'annoline1': 'Kingsbridge',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
 'bbox': [-73.90281798724604,
 40.88168737120521,
 -73.90281798724604,
 40.88168737120521]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.7',
 'geometry': {'type': 'Point',
 'coordinates': [-73.91065965862981, 40.87655077879964]},
 'geometry_name': 'geom',
 'properties': {'name': 'Marble Hill',
 'stacked': 2,
 'annoline1': 'Marble',
 'annoline2': 'Hill',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
 'bbox': [-73.91065965862981,
 40.87655077879964,
```

```
-73.91065965862981,
40.87655077879964]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.8',
'geometry': {'type': 'Point',
'coordinates': [-73.86731496814176, 40.89827261213805]},
'geometry_name': 'geom',
'properties': {'name': 'Woodlawn',
'stacked': 1,
'annoline1': 'Woodlawn',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.86731496814176,
40.89827261213805,
-73.86731496814176,
40.89827261213805]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.9',
'geometry': {'type': 'Point',
'coordinates': [-73.8793907395681, 40.87722415599446]},
'geometry_name': 'geom',
'properties': {'name': 'Norwood',
'stacked': 1,
'annoline1': 'Norwood',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.8793907395681,
40.87722415599446,
-73.8793907395681,
40.87722415599446]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.10',
'geometry': {'type': 'Point',
'coordinates': [-73.85744642974207, 40.88103887819211]},
'geometry_name': 'geom',
'properties': {'name': 'Williamsbridge',
'stacked': 1,
'annoline1': 'Williamsbridge',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.85744642974207,
40.88103887819211,
-73.85744642974207,
40.88103887819211]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.11',
'geometry': {'type': 'Point',
'coordinates': [-73.83579759808117, 40.866858107252696]},
'geometry_name': 'geom',
'properties': {'name': 'Baychester',
'stacked': 1,
```

```
'annoline1': 'Baychester',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.83579759808117,
40.866858107252696,
-73.83579759808117,
40.866858107252696]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.12',
'geometry': {'type': 'Point',
'coordinates': [-73.85475564017999, 40.85741349808865]},
'geometry_name': 'geom',
'properties': {'name': 'Pelham Parkway',
'stacked': 1,
'annoline1': 'Pelham Parkway',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.85475564017999,
40.85741349808865,
-73.85475564017999,
40.85741349808865]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.13',
'geometry': {'type': 'Point',
'coordinates': [-73.78648845267413, 40.84724670491813]},
'geometry_name': 'geom',
'properties': {'name': 'City Island',
'stacked': 2,
'annoline1': 'City',
'annoline2': 'Island',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.78648845267413,
40.84724670491813,
-73.78648845267413,
40.84724670491813]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.14',
'geometry': {'type': 'Point',
'coordinates': [-73.8855121841913, 40.870185164975325]},
'geometry_name': 'geom',
'properties': {'name': 'Bedford Park',
'stacked': 2,
'annoline1': 'Bedford',
'annoline2': 'Park',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.8855121841913,
40.870185164975325,
-73.8855121841913,
40.870185164975325]}},
```

```
{'type': 'Feature',
'id': 'nyu_2451_34572.15',
'geometry': {'type': 'Point',
'coordinates': [-73.9104159619131, 40.85572707719664]},
'geometry_name': 'geom',
'properties': {'name': 'University Heights',
'stacked': 2,
'annoline1': 'University',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.9104159619131,
40.85572707719664,
-73.9104159619131,
40.85572707719664]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.16',
'geometry': {'type': 'Point',
'coordinates': [-73.91967159119565, 40.84789792606271]},
'geometry_name': 'geom',
'properties': {'name': 'Morris Heights',
'stacked': 2,
'annoline1': 'Morris',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.91967159119565,
40.84789792606271,
-73.91967159119565,
40.84789792606271]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.17',
'geometry': {'type': 'Point',
'coordinates': [-73.89642655981623, 40.86099679638654]},
'geometry_name': 'geom',
'properties': {'name': 'Fordham',
'stacked': 1,
'annoline1': 'Fordham',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.89642655981623,
40.86099679638654,
-73.89642655981623,
40.86099679638654]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.18',
'geometry': {'type': 'Point',
'coordinates': [-73.88735617532338, 40.84269615786053]},
'geometry_name': 'geom',
'properties': {'name': 'East Tremont',
'stacked': 2,
'annoline1': 'East',
'annoline2': 'Tremont',
```

```
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.88735617532338,
 40.84269615786053,
 -73.88735617532338,
 40.84269615786053]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.19',
'geometry': {'type': 'Point',
 'coordinates': [-73.87774474910545, 40.83947505672653]},
'geometry_name': 'geom',
'properties': {'name': 'West Farms',
 'stacked': 2,
 'annoline1': 'West',
 'annoline2': 'Farms',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
'bbox': [-73.87774474910545,
 40.83947505672653,
 -73.87774474910545,
 40.83947505672653]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.20',
'geometry': {'type': 'Point',
 'coordinates': [-73.9261020935813, 40.836623010706056]},
'geometry_name': 'geom',
'properties': {'name': 'High Bridge',
 'stacked': 1,
 'annoline1': 'Highbridge',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
'bbox': [-73.9261020935813,
 40.836623010706056,
 -73.9261020935813,
 40.836623010706056]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.21',
'geometry': {'type': 'Point',
 'coordinates': [-73.90942160757436, 40.819754370594936]},
'geometry_name': 'geom',
'properties': {'name': 'Melrose',
 'stacked': 1,
 'annoline1': 'Melrose',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
'bbox': [-73.90942160757436,
 40.819754370594936,
 -73.90942160757436,
 40.819754370594936]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.22',
```

```
'geometry': {'type': 'Point',
  'coordinates': [-73.91609987487575, 40.80623874935177]},
'geometry_name': 'geom',
'properties': {'name': 'Mott Haven',
  'stacked': 1,
  'annoline1': 'Mott Haven',
  'annoline2': None,
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Bronx',
  'bbox': [-73.91609987487575,
  40.80623874935177,
  -73.91609987487575,
  40.80623874935177]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.23',
  'geometry': {'type': 'Point',
    'coordinates': [-73.91322139386135, 40.801663627756206]},
  'geometry_name': 'geom',
  'properties': {'name': 'Port Morris',
    'stacked': 2,
    'annoline1': 'Port',
    'annoline2': 'Morris',
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Bronx',
    'bbox': [-73.91322139386135,
    40.801663627756206,
    -73.91322139386135,
    40.801663627756206]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.24',
  'geometry': {'type': 'Point',
    'coordinates': [-73.8957882009446, 40.81509904545822]},
  'geometry_name': 'geom',
  'properties': {'name': 'Longwood',
    'stacked': 1,
    'annoline1': 'Longwood',
    'annoline2': None,
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Bronx',
    'bbox': [-73.8957882009446,
    40.81509904545822,
    -73.8957882009446,
    40.81509904545822]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.25',
  'geometry': {'type': 'Point',
    'coordinates': [-73.88331505955291, 40.80972987938709]},
  'geometry_name': 'geom',
  'properties': {'name': 'Hunts Point',
    'stacked': 2,
    'annoline1': 'Hunts',
    'annoline2': 'Point',
    'annoline3': None,
    'annoangle': 0.0,
```

```
'borough': 'Bronx',
'bbox': [-73.88331505955291,
40.80972987938709,
-73.88331505955291,
40.80972987938709]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.26',
'geometry': {'type': 'Point',
'coordinates': [-73.90150648943059, 40.82359198585534]},
'geometry_name': 'geom',
'properties': {'name': 'Morrisania',
'stacked': 1,
'annoline1': 'Morrisania',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.90150648943059,
40.82359198585534,
-73.90150648943059,
40.82359198585534]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.27',
'geometry': {'type': 'Point',
'coordinates': [-73.86574609554924, 40.821012197914015]},
'geometry_name': 'geom',
'properties': {'name': 'Soundview',
'stacked': 1,
'annoline1': 'Soundview',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.86574609554924,
40.821012197914015,
-73.86574609554924,
40.821012197914015]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.28',
'geometry': {'type': 'Point',
'coordinates': [-73.85414416189266, 40.80655112003589]},
'geometry_name': 'geom',
'properties': {'name': 'Clason Point',
'stacked': 2,
'annoline1': 'Clason',
'annoline2': 'Point',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.85414416189266,
40.80655112003589,
-73.85414416189266,
40.80655112003589]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.29',
'geometry': {'type': 'Point',
'coordinates': [-73.81635002158441, 40.81510925804005]},
```

```
'geometry_name': 'geom',
'properties': {'name': 'Throgs Neck',
 'stacked': 1,
 'annoline1': 'Throgs Neck',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
 'bbox': [-73.81635002158441,
  40.81510925804005,
  -73.81635002158441,
  40.81510925804005]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.30',
 'geometry': {'type': 'Point',
  'coordinates': [-73.8240992675385, 40.844245936947374]},
 'geometry_name': 'geom',
 'properties': {'name': 'Country Club',
 'stacked': 2,
 'annoline1': 'Country',
 'annoline2': 'Club',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
 'bbox': [-73.8240992675385,
  40.844245936947374,
  -73.8240992675385,
  40.844245936947374]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.31',
 'geometry': {'type': 'Point',
  'coordinates': [-73.85600310535783, 40.837937822267286]},
 'geometry_name': 'geom',
 'properties': {'name': 'Parkchester',
 'stacked': 1,
 'annoline1': 'Parkchester',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
 'bbox': [-73.85600310535783,
  40.837937822267286,
  -73.85600310535783,
  40.837937822267286]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.32',
 'geometry': {'type': 'Point',
  'coordinates': [-73.84219407604444, 40.8406194964327]},
 'geometry_name': 'geom',
 'properties': {'name': 'Westchester Square',
 'stacked': 2,
 'annoline1': 'Westchester',
 'annoline2': 'Square',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Bronx',
 'bbox': [-73.84219407604444,
```

```
40.8406194964327,  
-73.84219407604444,  
40.8406194964327]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.33',  
'geometry': {'type': 'Point',  
'coordinates': [-73.8662991807561, 40.84360847124718]},  
'geometry_name': 'geom',  
'properties': {'name': 'Van Nest',  
'stacked': 2,  
'annoline1': 'Van',  
'annoline2': 'Nest',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Bronx',  
'bbox': [-73.8662991807561,  
40.84360847124718,  
-73.8662991807561,  
40.84360847124718]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.34',  
'geometry': {'type': 'Point',  
'coordinates': [-73.85040178030421, 40.847549063536334]},  
'geometry_name': 'geom',  
'properties': {'name': 'Morris Park',  
'stacked': 1,  
'annoline1': 'Morris Park',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Bronx',  
'bbox': [-73.85040178030421,  
40.847549063536334,  
-73.85040178030421,  
40.847549063536334]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.35',  
'geometry': {'type': 'Point',  
'coordinates': [-73.88845196134804, 40.85727710073895]},  
'geometry_name': 'geom',  
'properties': {'name': 'Belmont',  
'stacked': 1,  
'annoline1': 'Belmont',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Bronx',  
'bbox': [-73.88845196134804,  
40.85727710073895,  
-73.88845196134804,  
40.85727710073895]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.36',  
'geometry': {'type': 'Point',  
'coordinates': [-73.91719048210393, 40.88139497727086]},  
'geometry_name': 'geom',  
'properties': {'name': 'Spuyten Duyvil',
```

```
'stacked': 2,
'annoline1': 'Spuyten',
'annoline2': 'Duyvil',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.91719048210393,
40.88139497727086,
-73.91719048210393,
40.88139497727086]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.37',
'geometry': {'type': 'Point',
'coordinates': [-73.90453054908927, 40.90854282950666]},
'geometry_name': 'geom',
'properties': {'name': 'North Riverdale',
'stacked': 2,
'annoline1': 'North',
'annoline2': 'Riverdale',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.90453054908927,
40.90854282950666,
-73.90453054908927,
40.90854282950666]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.38',
'geometry': {'type': 'Point',
'coordinates': [-73.8320737824047, 40.85064140940335]},
'geometry_name': 'geom',
'properties': {'name': 'Pelham Bay',
'stacked': 2,
'annoline1': 'Pelham',
'annoline2': 'Bay',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.8320737824047,
40.85064140940335,
-73.8320737824047,
40.85064140940335]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.39',
'geometry': {'type': 'Point',
'coordinates': [-73.82620275994073, 40.82657951686922]},
'geometry_name': 'geom',
'properties': {'name': 'Schuylerville',
'stacked': 1,
'annoline1': 'Schuylerville',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.82620275994073,
40.82657951686922,
-73.82620275994073,
```

```
        40.82657951686922]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.40',  
'geometry': {'type': 'Point',  
'coordinates': [-73.81388514428619, 40.821986118163494]},  
'geometry_name': 'geom',  
'properties': {'name': 'Edgewater Park',  
'stacked': 2,  
'annoline1': 'Edgewater',  
'annoline2': 'Park',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Bronx',  
'bbox': [-73.81388514428619,  
40.821986118163494,  
-73.81388514428619,  
40.821986118163494]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.41',  
'geometry': {'type': 'Point',  
'coordinates': [-73.84802729582735, 40.819014376988314]},  
'geometry_name': 'geom',  
'properties': {'name': 'Castle Hill',  
'stacked': 2,  
'annoline1': 'Castle',  
'annoline2': 'Hill',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Bronx',  
'bbox': [-73.84802729582735,  
40.819014376988314,  
-73.84802729582735,  
40.819014376988314]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.42',  
'geometry': {'type': 'Point',  
'coordinates': [-73.86332361652777, 40.87137078192371]},  
'geometry_name': 'geom',  
'properties': {'name': 'Olinville',  
'stacked': 1,  
'annoline1': 'Olinville',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Bronx',  
'bbox': [-73.86332361652777,  
40.87137078192371,  
-73.86332361652777,  
40.87137078192371]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.43',  
'geometry': {'type': 'Point',  
'coordinates': [-73.84161194831223, 40.86296562477998]},  
'geometry_name': 'geom',  
'properties': {'name': 'Pelham Gardens',  
'stacked': 2,  
'annoline1': 'Pelham',
```

```
'annoline2': 'Gardens',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.84161194831223,
40.86296562477998,
-73.84161194831223,
40.86296562477998]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.44',
'geometry': {'type': 'Point',
'coordinates': [-73.91558941773444, 40.83428380733851]},
'geometry_name': 'geom',
'properties': {'name': 'Concourse',
'stacked': 1,
'annoline1': 'Concourse',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.91558941773444,
40.83428380733851,
-73.91558941773444,
40.83428380733851]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.45',
'geometry': {'type': 'Point',
'coordinates': [-73.85053524451935, 40.82977429787161]},
'geometry_name': 'geom',
'properties': {'name': 'Unionport',
'stacked': 1,
'annoline1': 'Unionport',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.85053524451935,
40.82977429787161,
-73.85053524451935,
40.82977429787161]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.46',
'geometry': {'type': 'Point',
'coordinates': [-73.84808271877168, 40.88456130303732]},
'geometry_name': 'geom',
'properties': {'name': 'Edenwald',
'stacked': 1,
'annoline1': 'Edenwald',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.84808271877168,
40.88456130303732,
-73.84808271877168,
40.88456130303732]}},
{'type': 'Feature',
```

```
'id': 'nyu_2451_34572.47',
'geometry': {'type': 'Point',
 'coordinates': [-74.03062069353813, 40.625801065010656]},
'geometry_name': 'geom',
'properties': {'name': 'Bay Ridge',
 'stacked': 1,
 'annoline1': 'Bay Ridge',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-74.03062069353813,
 40.625801065010656,
 -74.03062069353813,
 40.625801065010656]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.48',
'geometry': {'type': 'Point',
 'coordinates': [-73.99517998380729, 40.61100890202044]},
'geometry_name': 'geom',
'properties': {'name': 'Bensonhurst',
 'stacked': 1,
 'annoline1': 'Bensonhurst',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-73.99517998380729,
 40.61100890202044,
 -73.99517998380729,
 40.61100890202044]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.49',
'geometry': {'type': 'Point',
 'coordinates': [-74.01031618527784, 40.64510294925429]},
'geometry_name': 'geom',
'properties': {'name': 'Sunset Park',
 'stacked': 2,
 'annoline1': 'Sunset',
 'annoline2': 'Park',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-74.01031618527784,
 40.64510294925429,
 -74.01031618527784,
 40.64510294925429]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.50',
'geometry': {'type': 'Point',
 'coordinates': [-73.95424093127393, 40.7302009848647]},
'geometry_name': 'geom',
'properties': {'name': 'Greenpoint',
 'stacked': 1,
 'annoline1': 'Greenpoint',
 'annoline2': None,
 'annoline3': None,
```

```
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.95424093127393,
40.7302009848647,
-73.95424093127393,
40.7302009848647]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.51',
'geometry': {'type': 'Point',
'coordinates': [-73.97347087708445, 40.59526001306593]},
'geometry_name': 'geom',
'properties': {'name': 'Gravesend',
'stacked': 1,
'annoline1': 'Gravesend',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.97347087708445,
40.59526001306593,
-73.97347087708445,
40.59526001306593]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.52',
'geometry': {'type': 'Point',
'coordinates': [-73.96509448785336, 40.57682506566604]},
'geometry_name': 'geom',
'properties': {'name': 'Brighton Beach',
'stacked': 2,
'annoline1': 'Brighton',
'annoline2': 'Beach',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.96509448785336,
40.57682506566604,
-73.96509448785336,
40.57682506566604]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.53',
'geometry': {'type': 'Point',
'coordinates': [-73.94318640482979, 40.58689012678384]},
'geometry_name': 'geom',
'properties': {'name': 'Sheepshead Bay',
'stacked': 2,
'annoline1': 'Sheepshead',
'annoline2': 'Bay',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.94318640482979,
40.58689012678384,
-73.94318640482979,
40.58689012678384]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.54',
'geometry': {'type': 'Point',
```

```
'coordinates': [-73.95743840559939, 40.61443251335098]},  
'geometry_name': 'geom',  
'properties': {'name': 'Manhattan Terrace',  
'stacked': 2,  
'annoline1': 'Manhattan',  
'annoline2': 'Terrace',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.95743840559939,  
40.61443251335098,  
-73.95743840559939,  
40.61443251335098]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.55',  
'geometry': {'type': 'Point',  
'coordinates': [-73.95840106533903, 40.63632589026677]},  
'geometry_name': 'geom',  
'properties': {'name': 'Flatbush',  
'stacked': 1,  
'annoline1': 'Flatbush',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.95840106533903,  
40.63632589026677,  
-73.95840106533903,  
40.63632589026677]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.56',  
'geometry': {'type': 'Point',  
'coordinates': [-73.94329119073582, 40.67082917695294]},  
'geometry_name': 'geom',  
'properties': {'name': 'Crown Heights',  
'stacked': 2,  
'annoline1': 'Crown',  
'annoline2': 'Heights',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.94329119073582,  
40.67082917695294,  
-73.94329119073582,  
40.67082917695294]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.57',  
'geometry': {'type': 'Point',  
'coordinates': [-73.93610256185836, 40.64171776668961]},  
'geometry_name': 'geom',  
'properties': {'name': 'East Flatbush',  
'stacked': 1,  
'annoline1': 'East Flatbush',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',
```

```
'bbox': [-73.93610256185836,
40.64171776668961,
-73.93610256185836,
40.64171776668961]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.58',
'geometry': {'type': 'Point',
'coordinates': [-73.98042110559474, 40.642381958003526]},
'geometry_name': 'geom',
'properties': {'name': 'Kensington',
'stacked': 1,
'annoline1': 'Kensington',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.98042110559474,
40.642381958003526,
-73.98042110559474,
40.642381958003526]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.59',
'geometry': {'type': 'Point',
'coordinates': [-73.98007340430172, 40.65694583575104]},
'geometry_name': 'geom',
'properties': {'name': 'Windsor Terrace',
'stacked': 2,
'annoline1': 'Windsor',
'annoline2': 'Terrace',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.98007340430172,
40.65694583575104,
-73.98007340430172,
40.65694583575104]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.60',
'geometry': {'type': 'Point',
'coordinates': [-73.9648592426269, 40.676822262254724]},
'geometry_name': 'geom',
'properties': {'name': 'Prospect Heights',
'stacked': 2,
'annoline1': 'Prospect',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.9648592426269,
40.676822262254724,
-73.9648592426269,
40.676822262254724]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.61',
'geometry': {'type': 'Point',
'coordinates': [-73.91023536176607, 40.66394994339755]},
'geometry_name': 'geom',
```

```
'properties': {'name': 'Brownsville',
 'stacked': 1,
 'annoline1': 'Brownsville',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-73.91023536176607,
 40.66394994339755,
 -73.91023536176607,
 40.66394994339755]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.62',
 'geometry': {'type': 'Point',
 'coordinates': [-73.95811529220927, 40.70714439344251]},
 'geometry_name': 'geom',
 'properties': {'name': 'Williamsburg',
 'stacked': 1,
 'annoline1': 'Williamsburg',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-73.95811529220927,
 40.70714439344251,
 -73.95811529220927,
 40.70714439344251]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.63',
 'geometry': {'type': 'Point',
 'coordinates': [-73.92525797487045, 40.69811611017901]},
 'geometry_name': 'geom',
 'properties': {'name': 'Bushwick',
 'stacked': 1,
 'annoline1': 'Bushwick',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-73.92525797487045,
 40.69811611017901,
 -73.92525797487045,
 40.69811611017901]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.64',
 'geometry': {'type': 'Point',
 'coordinates': [-73.94178488690297, 40.687231607720456]},
 'geometry_name': 'geom',
 'properties': {'name': 'Bedford Stuyvesant',
 'stacked': 1,
 'annoline1': 'Bedford Stuyvesant',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-73.94178488690297,
 40.687231607720456,
```

```
-73.94178488690297,  
 40.687231607720456]}}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.65',  
 'geometry': {'type': 'Point',  
  'coordinates': [-73.99378225496424, 40.695863722724084]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Brooklyn Heights',  
 'stacked': 2,  
 'annoline1': 'Brooklyn',  
 'annoline2': 'Heights',  
 'annoline3': None,  
 'annoangle': 0.0,  
 'borough': 'Brooklyn',  
 'bbox': [-73.99378225496424,  
 40.695863722724084,  
 -73.99378225496424,  
 40.695863722724084]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.66',  
 'geometry': {'type': 'Point',  
  'coordinates': [-73.99856139218463, 40.687919722485574]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Cobble Hill',  
 'stacked': 2,  
 'annoline1': 'Cobble',  
 'annoline2': 'Hill',  
 'annoline3': None,  
 'annoangle': 0.0,  
 'borough': 'Brooklyn',  
 'bbox': [-73.99856139218463,  
 40.687919722485574,  
 -73.99856139218463,  
 40.687919722485574]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.67',  
 'geometry': {'type': 'Point',  
  'coordinates': [-73.99465372828006, 40.680540231076485]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Carroll Gardens',  
 'stacked': 2,  
 'annoline1': 'Carroll',  
 'annoline2': 'Gardens',  
 'annoline3': None,  
 'annoangle': 0.0,  
 'borough': 'Brooklyn',  
 'bbox': [-73.99465372828006,  
 40.680540231076485,  
 -73.99465372828006,  
 40.680540231076485]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.68',  
 'geometry': {'type': 'Point',  
  'coordinates': [-74.0127589747356, 40.676253230250886]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Red Hook',  
 'stacked': 2,
```

```
'annoline1': 'Red',
'annoline2': 'Hook',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-74.0127589747356,
40.676253230250886,
-74.0127589747356,
40.676253230250886]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.69',
'geometry': {'type': 'Point',
'coordinates': [-73.99444087145339, 40.673931143187154]},
'geometry_name': 'geom',
'properties': {'name': 'Gowanus',
'stacked': 1,
'annoline1': 'Gowanus',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.99444087145339,
40.673931143187154,
-73.99444087145339,
40.673931143187154]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.70',
'geometry': {'type': 'Point',
'coordinates': [-73.97290574369092, 40.68852726018977]},
'geometry_name': 'geom',
'properties': {'name': 'Fort Greene',
'stacked': 2,
'annoline1': 'Fort',
'annoline2': 'Greene',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.97290574369092,
40.68852726018977,
-73.97290574369092,
40.68852726018977]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.71',
'geometry': {'type': 'Point',
'coordinates': [-73.97705030183924, 40.67232052268197]},
'geometry_name': 'geom',
'properties': {'name': 'Park Slope',
'stacked': 2,
'annoline1': 'Park',
'annoline2': 'Slope',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.97705030183924,
40.67232052268197,
-73.97705030183924,
40.67232052268197]}},
```

```
{'type': 'Feature',
'id': 'nyu_2451_34572.72',
'geometry': {'type': 'Point',
'coordinates': [-73.87661596457296, 40.68239101144211]},
'geometry_name': 'geom',
'properties': {'name': 'Cypress Hills',
'stacked': 2,
'annoline1': 'Cypress',
'annoline2': 'Hills',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.87661596457296,
40.68239101144211,
-73.87661596457296,
40.68239101144211]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.73',
'geometry': {'type': 'Point',
'coordinates': [-73.88069863917366, 40.669925700847045]},
'geometry_name': 'geom',
'properties': {'name': 'East New York',
'stacked': 1,
'annoline1': 'East New York',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.88069863917366,
40.669925700847045,
-73.88069863917366,
40.669925700847045]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.74',
'geometry': {'type': 'Point',
'coordinates': [-73.87936970045875, 40.64758905230874]},
'geometry_name': 'geom',
'properties': {'name': 'Starrett City',
'stacked': 2,
'annoline1': 'Starrett',
'annoline2': 'City',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.87936970045875,
40.64758905230874,
-73.87936970045875,
40.64758905230874]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.75',
'geometry': {'type': 'Point',
'coordinates': [-73.90209269778966, 40.63556432797428]},
'geometry_name': 'geom',
'properties': {'name': 'Canarsie',
'stacked': 1,
'annoline1': 'Canarsie',
'annoline2': None,
```

```
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.90209269778966,
 40.63556432797428,
 -73.90209269778966,
 40.63556432797428]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.76',
'geometry': {'type': 'Point',
 'coordinates': [-73.92911302644674, 40.630446043757466]},
'geometry_name': 'geom',
'properties': {'name': 'Flatlands',
 'stacked': 1,
 'annoline1': 'Flatlands',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
'bbox': [-73.92911302644674,
 40.630446043757466,
 -73.92911302644674,
 40.630446043757466]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.77',
'geometry': {'type': 'Point',
 'coordinates': [-73.90818571777423, 40.606336421685626]},
'geometry_name': 'geom',
'properties': {'name': 'Mill Island',
 'stacked': 2,
 'annoline1': 'Mill',
 'annoline2': 'Island',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
'bbox': [-73.90818571777423,
 40.606336421685626,
 -73.90818571777423,
 40.606336421685626]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.78',
'geometry': {'type': 'Point',
 'coordinates': [-73.94353722891886, 40.57791350308657]},
'geometry_name': 'geom',
'properties': {'name': 'Manhattan Beach',
 'stacked': 2,
 'annoline1': 'Manhattan',
 'annoline2': 'Beach',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
'bbox': [-73.94353722891886,
 40.57791350308657,
 -73.94353722891886,
 40.57791350308657]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.79',
```

```
'geometry': {'type': 'Point',
  'coordinates': [-73.98868295821637, 40.57429256471601]},
'geometry_name': 'geom',
'properties': {'name': 'Coney Island',
  'stacked': 1,
  'annoline1': 'Coney Island',
  'annoline2': None,
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Brooklyn',
  'bbox': [-73.98868295821637,
    40.57429256471601,
    -73.98868295821637,
    40.57429256471601]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.80',
  'geometry': {'type': 'Point',
    'coordinates': [-73.99875221443519, 40.59951870282238]},
  'geometry_name': 'geom',
  'properties': {'name': 'Bath Beach',
    'stacked': 2,
    'annoline1': 'Bath',
    'annoline2': 'Beach',
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Brooklyn',
    'bbox': [-73.99875221443519,
      40.59951870282238,
      -73.99875221443519,
      40.59951870282238]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.81',
  'geometry': {'type': 'Point',
    'coordinates': [-73.99049823044811, 40.633130512758015]},
  'geometry_name': 'geom',
  'properties': {'name': 'Borough Park',
    'stacked': 2,
    'annoline1': 'Borough',
    'annoline2': 'Park',
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Brooklyn',
    'bbox': [-73.99049823044811,
      40.633130512758015,
      -73.99049823044811,
      40.633130512758015]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.82',
  'geometry': {'type': 'Point',
    'coordinates': [-74.01931375636022, 40.619219457722636]},
  'geometry_name': 'geom',
  'properties': {'name': 'Dyker Heights',
    'stacked': 2,
    'annoline1': 'Dyker',
    'annoline2': 'Heights',
    'annoline3': None,
    'annoangle': 0.0,
```

```
'borough': 'Brooklyn',
'bbox': [-74.01931375636022,
40.619219457722636,
-74.01931375636022,
40.619219457722636]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.83',
'geometry': {'type': 'Point',
'coordinates': [-73.93010170691196, 40.590848433902046]},
'geometry_name': 'geom',
'properties': {'name': 'Gerritsen Beach',
'stacked': 2,
'annoline1': 'Gerritsen',
'annoline2': 'Beach',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.93010170691196,
40.590848433902046,
-73.93010170691196,
40.590848433902046]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.84',
'geometry': {'type': 'Point',
'coordinates': [-73.93134404108497, 40.609747779894604]},
'geometry_name': 'geom',
'properties': {'name': 'Marine Park',
'stacked': 1,
'annoline1': 'Marine Park',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.93134404108497,
40.609747779894604,
-73.93134404108497,
40.609747779894604]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.85',
'geometry': {'type': 'Point',
'coordinates': [-73.96784306216367, 40.693229421881504]},
'geometry_name': 'geom',
'properties': {'name': 'Clinton Hill',
'stacked': 2,
'annoline1': 'Clinton',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.96784306216367,
40.693229421881504,
-73.96784306216367,
40.693229421881504]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.86',
'geometry': {'type': 'Point',
'coordinates': [-74.0078731120024, 40.57637537890224]},
```

```
'geometry_name': 'geom',
'properties': {'name': 'Sea Gate',
 'stacked': 2,
 'annoline1': 'Sea',
 'annoline2': 'Gate',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-74.0078731120024,
  40.57637537890224,
  -74.0078731120024,
  40.57637537890224]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.87',
 'geometry': {'type': 'Point',
  'coordinates': [-73.98346337431099, 40.69084402109802]},
 'geometry_name': 'geom',
 'properties': {'name': 'Downtown',
  'stacked': 1,
  'annoline1': 'Downtown',
  'annoline2': None,
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Brooklyn',
  'bbox': [-73.98346337431099,
  40.69084402109802,
  -73.98346337431099,
  40.69084402109802]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.88',
 'geometry': {'type': 'Point',
  'coordinates': [-73.98374824115798, 40.685682912091444]},
 'geometry_name': 'geom',
 'properties': {'name': 'Boerum Hill',
  'stacked': 2,
  'annoline1': 'Boerum',
  'annoline2': 'Hill',
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Brooklyn',
  'bbox': [-73.98374824115798,
  40.685682912091444,
  -73.98374824115798,
  40.685682912091444]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.89',
 'geometry': {'type': 'Point',
  'coordinates': [-73.95489867077713, 40.658420017469815]},
 'geometry_name': 'geom',
 'properties': {'name': 'Prospect Lefferts Gardens',
  'stacked': 3,
  'annoline1': 'Prospect',
  'annoline2': 'Lefferts',
  'annoline3': 'Gardens',
  'annoangle': 0.0,
  'borough': 'Brooklyn',
  'bbox': [-73.95489867077713,
```

```
40.658420017469815,  
-73.95489867077713,  
40.658420017469815]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.90',  
'geometry': {'type': 'Point',  
'coordinates': [-73.91306831787395, 40.678402554795355]},  
'geometry_name': 'geom',  
'properties': {'name': 'Ocean Hill',  
'stacked': 2,  
'annoline1': 'Ocean',  
'annoline2': 'Hill',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.91306831787395,  
40.678402554795355,  
-73.91306831787395,  
40.678402554795355]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.91',  
'geometry': {'type': 'Point',  
'coordinates': [-73.86797598081334, 40.67856995727479]},  
'geometry_name': 'geom',  
'properties': {'name': 'City Line',  
'stacked': 2,  
'annoline1': 'City',  
'annoline2': 'Line',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.86797598081334,  
40.67856995727479,  
-73.86797598081334,  
40.67856995727479]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.92',  
'geometry': {'type': 'Point',  
'coordinates': [-73.89855633630317, 40.61514955045308]},  
'geometry_name': 'geom',  
'properties': {'name': 'Bergen Beach',  
'stacked': 2,  
'annoline1': 'Bergen',  
'annoline2': 'Beach',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.89855633630317,  
40.61514955045308,  
-73.89855633630317,  
40.61514955045308]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.93',  
'geometry': {'type': 'Point',  
'coordinates': [-73.95759523489838, 40.62559589869843]},  
'geometry_name': 'geom',  
'properties': {'name': 'Midwood',
```

```
'stacked': 1,
'annoline1': 'Midwood',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.95759523489838,
40.62559589869843,
-73.95759523489838,
40.62559589869843]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.94',
'geometry': {'type': 'Point',
'coordinates': [-73.96261316716048, 40.647008603185185]},
'geometry_name': 'geom',
'properties': {'name': 'Prospect Park South',
'stacked': 2,
'annoline1': 'Prospect',
'annoline2': 'Park South',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.96261316716048,
40.647008603185185,
-73.96261316716048,
40.647008603185185]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.95',
'geometry': {'type': 'Point',
'coordinates': [-73.91607483951324, 40.62384524478419]},
'geometry_name': 'geom',
'properties': {'name': 'Georgetown',
'stacked': 1,
'annoline1': 'Georgetown',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.91607483951324,
40.62384524478419,
-73.91607483951324,
40.62384524478419]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.96',
'geometry': {'type': 'Point',
'coordinates': [-73.93885815269195, 40.70849241041548]},
'geometry_name': 'geom',
'properties': {'name': 'East Williamsburg',
'stacked': 2,
'annoline1': 'East',
'annoline2': 'Williamsburg',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.93885815269195,
40.70849241041548,
-73.93885815269195,
```



```
'annoline2': 'Hamilton',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-74.03197914537984,
40.61476812694226,
-74.03197914537984,
40.61476812694226]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.101',
'geometry': {'type': 'Point',
'coordinates': [-73.99427936255978, 40.71561842231432]},
'geometry_name': 'geom',
'properties': {'name': 'Chinatown',
'stacked': 1,
'annoline1': 'Chinatown',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.99427936255978,
40.71561842231432,
-73.99427936255978,
40.71561842231432]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.102',
'geometry': {'type': 'Point',
'coordinates': [-73.93690027985234, 40.85190252555305]},
'geometry_name': 'geom',
'properties': {'name': 'Washington Heights',
'stacked': 2,
'annoline1': 'Washington',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.93690027985234,
40.85190252555305,
-73.93690027985234,
40.85190252555305]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.103',
'geometry': {'type': 'Point',
'coordinates': [-73.92121042203897, 40.86768396449915]},
'geometry_name': 'geom',
'properties': {'name': 'Inwood',
'stacked': 1,
'annoline1': 'Inwood',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.92121042203897,
40.86768396449915,
-73.92121042203897,
40.86768396449915]}},
{'type': 'Feature',
```

```
'id': 'nyu_2451_34572.104',
'geometry': {'type': 'Point',
 'coordinates': [-73.94968791883366, 40.823604284811935]},
'geometry_name': 'geom',
'properties': {'name': 'Hamilton Heights',
 'stacked': 2,
 'annoline1': 'Hamilton',
 'annoline2': 'Heights',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
 'bbox': [-73.94968791883366,
 40.823604284811935,
 -73.94968791883366,
 40.823604284811935]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.105',
'geometry': {'type': 'Point',
 'coordinates': [-73.9573853935188, 40.8169344294978]},
'geometry_name': 'geom',
'properties': {'name': 'Manhattanville',
 'stacked': 2,
 'annoline1': 'Manhattanville',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
 'bbox': [-73.9573853935188,
 40.8169344294978,
 -73.9573853935188,
 40.8169344294978]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.106',
'geometry': {'type': 'Point',
 'coordinates': [-73.94321112603905, 40.81597606742414]},
'geometry_name': 'geom',
'properties': {'name': 'Central Harlem',
 'stacked': 2,
 'annoline1': 'Central',
 'annoline2': 'Harlem',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
 'bbox': [-73.94321112603905,
 40.81597606742414,
 -73.94321112603905,
 40.81597606742414]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.107',
'geometry': {'type': 'Point',
 'coordinates': [-73.94418223148524, 40.79224946663033]},
'geometry_name': 'geom',
'properties': {'name': 'East Harlem',
 'stacked': 2,
 'annoline1': 'East',
 'annoline2': 'Harlem',
 'annoline3': None,
```

```
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.94418223148524,
40.79224946663033,
-73.94418223148524,
40.79224946663033]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.108',
'geometry': {'type': 'Point',
'coordinates': [-73.96050763135, 40.775638573301805]},
'geometry_name': 'geom',
'properties': {'name': 'Upper East Side',
'stacked': 3,
'annoline1': 'Upper',
'annoline2': 'East',
'annoline3': 'Side',
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.96050763135,
40.775638573301805,
-73.96050763135,
40.775638573301805]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.109',
'geometry': {'type': 'Point',
'coordinates': [-73.94711784471826, 40.775929849884875]},
'geometry_name': 'geom',
'properties': {'name': 'Yorkville',
'stacked': 1,
'annoline1': 'Yorkville',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.94711784471826,
40.775929849884875,
-73.94711784471826,
40.775929849884875]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.110',
'geometry': {'type': 'Point',
'coordinates': [-73.9588596881376, 40.76811265828733]},
'geometry_name': 'geom',
'properties': {'name': 'Lenox Hill',
'stacked': 2,
'annoline1': 'Lenox',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.9588596881376,
40.76811265828733,
-73.9588596881376,
40.76811265828733]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.111',
'geometry': {'type': 'Point',
```

```
'coordinates': [-73.94916769227953, 40.76215960576283]},  
'geometry_name': 'geom',  
'properties': {'name': 'Roosevelt Island',  
'stacked': 1,  
'annoline1': 'Roosevelt Island',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 56,  
'borough': 'Manhattan',  
'bbox': [-73.94916769227953,  
40.76215960576283,  
-73.94916769227953,  
40.76215960576283]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.112',  
'geometry': {'type': 'Point',  
'coordinates': [-73.97705923630603, 40.787657998534854]},  
'geometry_name': 'geom',  
'properties': {'name': 'Upper West Side',  
'stacked': 3,  
'annoline1': 'Upper',  
'annoline2': 'West',  
'annoline3': 'Side',  
'annoangle': 0.0,  
'borough': 'Manhattan',  
'bbox': [-73.97705923630603,  
40.787657998534854,  
-73.97705923630603,  
40.787657998534854]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.113',  
'geometry': {'type': 'Point',  
'coordinates': [-73.98533777001262, 40.77352888942166]},  
'geometry_name': 'geom',  
'properties': {'name': 'Lincoln Square',  
'stacked': 2,  
'annoline1': 'Lincoln',  
'annoline2': 'Square',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Manhattan',  
'bbox': [-73.98533777001262,  
40.77352888942166,  
-73.98533777001262,  
40.77352888942166]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.114',  
'geometry': {'type': 'Point',  
'coordinates': [-73.99611936309479, 40.75910089146212]},  
'geometry_name': 'geom',  
'properties': {'name': 'Clinton',  
'stacked': 1,  
'annoline1': 'Clinton',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Manhattan',
```

```
'bbox': [-73.99611936309479,
40.75910089146212,
-73.99611936309479,
40.75910089146212]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.115',
'geometry': {'type': 'Point',
'coordinates': [-73.98166882730304, 40.75469110270623]},  
'geometry_name': 'geom',
'properties': {'name': 'Midtown',
'stacked': 1,
'annoline1': 'Midtown',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.98166882730304,
40.75469110270623,
-73.98166882730304,
40.75469110270623]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.116',
'geometry': {'type': 'Point',
'coordinates': [-73.97833207924127, 40.748303077252174]},  
'geometry_name': 'geom',
'properties': {'name': 'Murray Hill',
'stacked': 2,
'annoline1': 'Murray',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.97833207924127,
40.748303077252174,
-73.97833207924127,
40.748303077252174]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.117',
'geometry': {'type': 'Point',
'coordinates': [-74.00311633472813, 40.744034706747975]},  
'geometry_name': 'geom',
'properties': {'name': 'Chelsea',
'stacked': 1,
'annoline1': 'Chelsea',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-74.00311633472813,
40.744034706747975,
-74.00311633472813,
40.744034706747975]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.118',
'geometry': {'type': 'Point',
'coordinates': [-73.99991402945902, 40.72693288536128]},  
'geometry_name': 'geom',
```

```
'properties': {'name': 'Greenwich Village',
  'stacked': 2,
  'annoline1': 'Greenwich',
  'annoline2': 'Village',
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Manhattan',
  'bbox': [-73.99991402945902,
  40.72693288536128,
  -73.99991402945902,
  40.72693288536128]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.119',
 'geometry': {'type': 'Point',
  'coordinates': [-73.98222616506416, 40.727846777270244]},
 'geometry_name': 'geom',
 'properties': {'name': 'East Village',
  'stacked': 2,
  'annoline1': 'East',
  'annoline2': 'Village',
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Manhattan',
  'bbox': [-73.98222616506416,
  40.727846777270244,
  -73.98222616506416,
  40.727846777270244]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.120',
 'geometry': {'type': 'Point',
  'coordinates': [-73.98089031999291, 40.71780674892765]},
 'geometry_name': 'geom',
 'properties': {'name': 'Lower East Side',
  'stacked': 3,
  'annoline1': 'Lower',
  'annoline2': 'East',
  'annoline3': 'Side',
  'annoangle': 0.0,
  'borough': 'Manhattan',
  'bbox': [-73.98089031999291,
  40.71780674892765,
  -73.98089031999291,
  40.71780674892765]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.121',
 'geometry': {'type': 'Point',
  'coordinates': [-74.01068328559087, 40.721521967443216]},
 'geometry_name': 'geom',
 'properties': {'name': 'Tribeca',
  'stacked': 1,
  'annoline1': 'Tribeca',
  'annoline2': None,
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Manhattan',
  'bbox': [-74.01068328559087,
  40.721521967443216,
```

```
-74.01068328559087,  
 40.721521967443216]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.122',  
 'geometry': {'type': 'Point',  
  'coordinates': [-73.99730467208073, 40.71932379395907]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Little Italy',  
 'stacked': 2,  
 'annoline1': 'Little',  
 'annoline2': 'Italy',  
 'annoline3': None,  
 'annoangle': 0.0,  
 'borough': 'Manhattan',  
 'bbox': [-73.99730467208073,  
 40.71932379395907,  
 -73.99730467208073,  
 40.71932379395907]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.123',  
 'geometry': {'type': 'Point',  
  'coordinates': [-74.00065666959759, 40.72218384131794]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Soho',  
 'stacked': 1,  
 'annoline1': 'Soho',  
 'annoline2': None,  
 'annoline3': None,  
 'annoangle': 0.0,  
 'borough': 'Manhattan',  
 'bbox': [-74.00065666959759,  
 40.72218384131794,  
 -74.00065666959759,  
 40.72218384131794]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.124',  
 'geometry': {'type': 'Point',  
  'coordinates': [-74.00617998126812, 40.73443393572434]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'West Village',  
 'stacked': 2,  
 'annoline1': 'West',  
 'annoline2': 'Village',  
 'annoline3': None,  
 'annoangle': 0.0,  
 'borough': 'Manhattan',  
 'bbox': [-74.00617998126812,  
 40.73443393572434,  
 -74.00617998126812,  
 40.73443393572434]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.125',  
 'geometry': {'type': 'Point',  
  'coordinates': [-73.96428617740655, 40.797307041702865]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Manhattan Valley',  
 'stacked': 2,
```

```
'annoline1': 'Manhattan',
'annoline2': 'Valley',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.96428617740655,
40.797307041702865,
-73.96428617740655,
40.797307041702865]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.126',
'geometry': {'type': 'Point',
'coordinates': [-73.96389627905332, 40.807999738165826]},
'geometry_name': 'geom',
'properties': {'name': 'Morningside Heights',
'stacked': 2,
'annoline1': 'Morningside',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.96389627905332,
40.807999738165826,
-73.96389627905332,
40.807999738165826]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.127',
'geometry': {'type': 'Point',
'coordinates': [-73.98137594833541, 40.737209832715]},
'geometry_name': 'geom',
'properties': {'name': 'Gramercy',
'stacked': 1,
'annoline1': 'Gramercy',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.98137594833541,
40.737209832715,
-73.98137594833541,
40.737209832715]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.128',
'geometry': {'type': 'Point',
'coordinates': [-74.01686930508617, 40.71193198394565]},
'geometry_name': 'geom',
'properties': {'name': 'Battery Park City',
'stacked': 3,
'annoline1': 'Battery',
'annoline2': 'Park',
'annoline3': 'City',
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-74.01686930508617,
40.71193198394565,
-74.01686930508617,
40.71193198394565]}},
```

```
{'type': 'Feature',
'id': 'nyu_2451_34572.129',
'geometry': {'type': 'Point',
'coordinates': [-74.0106654452127, 40.70710710727048]},
'geometry_name': 'geom',
'properties': {'name': 'Financial District',
'stacked': 2,
'annoline1': 'Financial',
'annoline2': 'District',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-74.0106654452127,
40.70710710727048,
-74.0106654452127,
40.70710710727048]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.130',
'geometry': {'type': 'Point',
'coordinates': [-73.91565374304234, 40.76850859335492]},
'geometry_name': 'geom',
'properties': {'name': 'Astoria',
'stacked': 1,
'annoline1': 'Astoria',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.91565374304234,
40.76850859335492,
-73.91565374304234,
40.76850859335492]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.131',
'geometry': {'type': 'Point',
'coordinates': [-73.90184166838284, 40.74634908860222]},
'geometry_name': 'geom',
'properties': {'name': 'Woodside',
'stacked': 1,
'annoline1': 'Woodside',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.90184166838284,
40.74634908860222,
-73.90184166838284,
40.74634908860222]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.132',
'geometry': {'type': 'Point',
'coordinates': [-73.88282109164365, 40.75198138007367]},
'geometry_name': 'geom',
'properties': {'name': 'Jackson Heights',
'stacked': 2,
'annoline1': 'Jackson',
'annoline2': 'Heights',
```

```
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.88282109164365,
 40.75198138007367,
 -73.88282109164365,
 40.75198138007367]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.133',
'geometry': {'type': 'Point',
 'coordinates': [-73.88165622288388, 40.744048505122024]},
'geometry_name': 'geom',
'properties': {'name': 'Elmhurst',
 'stacked': 1,
 'annoline1': 'Elmhurst',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
'bbox': [-73.88165622288388,
 40.744048505122024,
 -73.88165622288388,
 40.744048505122024]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.134',
'geometry': {'type': 'Point',
 'coordinates': [-73.8381376460028, 40.65422527738487]},
'geometry_name': 'geom',
'properties': {'name': 'Howard Beach',
 'stacked': 2,
 'annoline1': 'Howard',
 'annoline2': 'Beach',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
'bbox': [-73.8381376460028,
 40.65422527738487,
 -73.8381376460028,
 40.65422527738487]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.135',
'geometry': {'type': 'Point',
 'coordinates': [-73.85682497345258, 40.74238175015667]},
'geometry_name': 'geom',
'properties': {'name': 'Corona',
 'stacked': 1,
 'annoline1': 'Corona',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
'bbox': [-73.85682497345258,
 40.74238175015667,
 -73.85682497345258,
 40.74238175015667]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.136',
```

```
'geometry': {'type': 'Point',
  'coordinates': [-73.84447500788983, 40.72526378216503]},
'geometry_name': 'geom',
'properties': {'name': 'Forest Hills',
  'stacked': 2,
  'annoline1': 'Forest',
  'annoline2': 'Hills',
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Queens',
  'bbox': [-73.84447500788983,
  40.72526378216503,
  -73.84447500788983,
  40.72526378216503]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.137',
  'geometry': {'type': 'Point',
    'coordinates': [-73.82981905825703, 40.7051790354148]},
  'geometry_name': 'geom',
  'properties': {'name': 'Kew Gardens',
    'stacked': 2,
    'annoline1': 'Kew',
    'annoline2': 'Gardens',
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Queens',
    'bbox': [-73.82981905825703,
    40.7051790354148,
    -73.82981905825703,
    40.7051790354148]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.138',
  'geometry': {'type': 'Point',
    'coordinates': [-73.83183321446887, 40.69794731471763]},
  'geometry_name': 'geom',
  'properties': {'name': 'Richmond Hill',
    'stacked': 2,
    'annoline1': 'Richmond',
    'annoline2': 'Hill',
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Queens',
    'bbox': [-73.83183321446887,
    40.69794731471763,
    -73.83183321446887,
    40.69794731471763]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.139',
  'geometry': {'type': 'Point',
    'coordinates': [-73.83177300329582, 40.76445419697846]},
  'geometry_name': 'geom',
  'properties': {'name': 'Flushing',
    'stacked': 1,
    'annoline1': 'Flushing',
    'annoline2': None,
    'annoline3': None,
    'annoangle': 0.0,
```

```
'borough': 'Queens',
'bbox': [-73.83177300329582,
40.76445419697846,
-73.83177300329582,
40.76445419697846]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.140',
'geometry': {'type': 'Point',
'coordinates': [-73.93920223915505, 40.75021734610528]},
'geometry_name': 'geom',
'properties': {'name': 'Long Island City',
'stacked': 3,
'annoline1': 'Long',
'annoline2': 'Island',
'annoline3': 'City',
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.93920223915505,
40.75021734610528,
-73.93920223915505,
40.75021734610528]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.141',
'geometry': {'type': 'Point',
'coordinates': [-73.92691617561577, 40.74017628351924]},
'geometry_name': 'geom',
'properties': {'name': 'Sunnyside',
'stacked': 1,
'annoline1': 'Sunnyside',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.92691617561577,
40.74017628351924,
-73.92691617561577,
40.74017628351924]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.142',
'geometry': {'type': 'Point',
'coordinates': [-73.86704147658772, 40.76407323883091]},
'geometry_name': 'geom',
'properties': {'name': 'East Elmhurst',
'stacked': 2,
'annoline1': 'East',
'annoline2': 'Elmhurst',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.86704147658772,
40.76407323883091,
-73.86704147658772,
40.76407323883091]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.143',
'geometry': {'type': 'Point',
'coordinates': [-73.89621713626859, 40.725427374093606]}},
```

```
'geometry_name': 'geom',
'properties': {'name': 'Maspeth',
 'stacked': 1,
 'annoline1': 'Maspeth',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.89621713626859,
 40.725427374093606,
 -73.89621713626859,
 40.725427374093606]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.144',
 'geometry': {'type': 'Point',
 'coordinates': [-73.90143517559589, 40.70832315613858]},
 'geometry_name': 'geom',
 'properties': {'name': 'Ridgewood',
 'stacked': 1,
 'annoline1': 'Ridgewood',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.90143517559589,
 40.70832315613858,
 -73.90143517559589,
 40.70832315613858]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.145',
 'geometry': {'type': 'Point',
 'coordinates': [-73.87074167435605, 40.70276242967838]},
 'geometry_name': 'geom',
 'properties': {'name': 'Glendale',
 'stacked': 1,
 'annoline1': 'Glendale',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.87074167435605,
 40.70276242967838,
 -73.87074167435605,
 40.70276242967838]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.146',
 'geometry': {'type': 'Point',
 'coordinates': [-73.8578268690537, 40.72897409480735]},
 'geometry_name': 'geom',
 'properties': {'name': 'Rego Park',
 'stacked': 1,
 'annoline1': 'Rego Park',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.8578268690537,
```

```
40.72897409480735,
-73.8578268690537,
40.72897409480735]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.147',
'geometry': {'type': 'Point',
'coordinates': [-73.8581104655432, 40.68988687915789]},
'geometry_name': 'geom',
'properties': {'name': 'Woodhaven',
'stacked': 1,
'annoline1': 'Woodhaven',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.8581104655432,
40.68988687915789,
-73.8581104655432,
40.68988687915789]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.148',
'geometry': {'type': 'Point',
'coordinates': [-73.84320266173447, 40.680708468265415]},
'geometry_name': 'geom',
'properties': {'name': 'Ozone Park',
'stacked': 1,
'annoline1': 'Ozone Park',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.84320266173447,
40.680708468265415,
-73.84320266173447,
40.680708468265415]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.149',
'geometry': {'type': 'Point',
'coordinates': [-73.80986478649041, 40.66854957767195]},
'geometry_name': 'geom',
'properties': {'name': 'South Ozone Park',
'stacked': 2,
'annoline1': 'South',
'annoline2': 'Ozone Park',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.80986478649041,
40.66854957767195,
-73.80986478649041,
40.66854957767195]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.150',
'geometry': {'type': 'Point',
'coordinates': [-73.84304528896125, 40.784902749260205]},
'geometry_name': 'geom',
'properties': {'name': 'College Point',
```

```
'stacked': 2,
'annoline1': 'College',
'annoline2': 'Point',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.84304528896125,
40.784902749260205,
-73.84304528896125,
40.784902749260205]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.151',
'geometry': {'type': 'Point',
'coordinates': [-73.81420216610863, 40.78129076602694]},
'geometry_name': 'geom',
'properties': {'name': 'Whitestone',
'stacked': 1,
'annoline1': 'Whitestone',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.81420216610863,
40.78129076602694,
-73.81420216610863,
40.78129076602694]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.152',
'geometry': {'type': 'Point',
'coordinates': [-73.7742736306867, 40.76604063281064]},
'geometry_name': 'geom',
'properties': {'name': 'Bayside',
'stacked': 1,
'annoline1': 'Bayside',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.7742736306867,
40.76604063281064,
-73.7742736306867,
40.76604063281064]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.153',
'geometry': {'type': 'Point',
'coordinates': [-73.79176243728061, 40.76172954903262]},
'geometry_name': 'geom',
'properties': {'name': 'Auburndale',
'stacked': 1,
'annoline1': 'Auburndale',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.79176243728061,
40.76172954903262,
-73.79176243728061,
```

```
        40.76172954903262]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.154',  
'geometry': {'type': 'Point',  
'coordinates': [-73.7388977558074, 40.7708261928267]},  
'geometry_name': 'geom',  
'properties': {'name': 'Little Neck',  
'stacked': 2,  
'annoline1': 'Little',  
'annoline2': 'Neck',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Queens',  
'bbox': [-73.7388977558074,  
40.7708261928267,  
-73.7388977558074,  
40.7708261928267]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.155',  
'geometry': {'type': 'Point',  
'coordinates': [-73.7424982072733, 40.76684609790763]},  
'geometry_name': 'geom',  
'properties': {'name': 'Douglaston',  
'stacked': 1,  
'annoline1': 'Douglaston',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Queens',  
'bbox': [-73.7424982072733,  
40.76684609790763,  
-73.7424982072733,  
40.76684609790763]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.156',  
'geometry': {'type': 'Point',  
'coordinates': [-73.71548118999145, 40.74944079974332]},  
'geometry_name': 'geom',  
'properties': {'name': 'Glen Oaks',  
'stacked': 2,  
'annoline1': 'Glen',  
'annoline2': 'Oaks',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Queens',  
'bbox': [-73.71548118999145,  
40.74944079974332,  
-73.71548118999145,  
40.74944079974332]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.157',  
'geometry': {'type': 'Point',  
'coordinates': [-73.72012814826903, 40.72857318176675]},  
'geometry_name': 'geom',  
'properties': {'name': 'Bellerose',  
'stacked': 1,  
'annoline1': 'Bellerose',
```

```
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Queens',  
'bbox': [-73.72012814826903,  
40.72857318176675,  
-73.72012814826903,  
40.72857318176675]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.158',  
'geometry': {'type': 'Point',  
'coordinates': [-73.82087764933566, 40.722578244228046]},  
'geometry_name': 'geom',  
'properties': {'name': 'Kew Gardens Hills',  
'stacked': 3,  
'annoline1': 'Kew',  
'annoline2': 'Gardens',  
'annoline3': 'Hills',  
'annoangle': 0.0,  
'borough': 'Queens',  
'bbox': [-73.82087764933566,  
40.722578244228046,  
-73.82087764933566,  
40.722578244228046]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.159',  
'geometry': {'type': 'Point',  
'coordinates': [-73.78271337003264, 40.7343944653313]},  
'geometry_name': 'geom',  
'properties': {'name': 'Fresh Meadows',  
'stacked': 2,  
'annoline1': 'Fresh',  
'annoline2': 'Meadows',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Queens',  
'bbox': [-73.78271337003264,  
40.7343944653313,  
-73.78271337003264,  
40.7343944653313]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.160',  
'geometry': {'type': 'Point',  
'coordinates': [-73.81174822458634, 40.71093547252271]},  
'geometry_name': 'geom',  
'properties': {'name': 'Briarwood',  
'stacked': 1,  
'annoline1': 'Briarwood',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Queens',  
'bbox': [-73.81174822458634,  
40.71093547252271,  
-73.81174822458634,  
40.71093547252271]}},  
{'type': 'Feature',
```

```
'id': 'nyu_2451_34572.161',
'geometry': {'type': 'Point',
 'coordinates': [-73.79690165888289, 40.70465736068717]},
'geometry_name': 'geom',
'properties': {'name': 'Jamaica Center',
 'stacked': 2,
 'annoline1': 'Jamaica',
 'annoline2': 'Center',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.79690165888289,
 40.70465736068717,
 -73.79690165888289,
 40.70465736068717]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.162',
'geometry': {'type': 'Point',
 'coordinates': [-73.75494976234332, 40.74561857141855]},
'geometry_name': 'geom',
'properties': {'name': 'Oakland Gardens',
 'stacked': 2,
 'annoline1': 'Oakland',
 'annoline2': 'Gardens',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.75494976234332,
 40.74561857141855,
 -73.75494976234332,
 40.74561857141855]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.163',
'geometry': {'type': 'Point',
 'coordinates': [-73.73871484578424, 40.718893092167356]},
'geometry_name': 'geom',
'properties': {'name': 'Queens Village',
 'stacked': 2,
 'annoline1': 'Queens',
 'annoline2': 'Village',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.73871484578424,
 40.718893092167356,
 -73.73871484578424,
 40.718893092167356]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.164',
'geometry': {'type': 'Point',
 'coordinates': [-73.75925009335594, 40.71124344191904]},
'geometry_name': 'geom',
'properties': {'name': 'Hollis',
 'stacked': 1,
 'annoline1': 'Hollis',
 'annoline2': None,
 'annoline3': None,
```

```
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.75925009335594,
40.71124344191904,
-73.75925009335594,
40.71124344191904]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.165',
'geometry': {'type': 'Point',
'coordinates': [-73.7904261313554, 40.696911253789885]},
'geometry_name': 'geom',
'properties': {'name': 'South Jamaica',
'stacked': 1,
'annoline1': 'South Jamaica',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.7904261313554,
40.696911253789885,
-73.7904261313554,
40.696911253789885]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.166',
'geometry': {'type': 'Point',
'coordinates': [-73.75867603727717, 40.69444538522359]},
'geometry_name': 'geom',
'properties': {'name': 'St. Albans',
'stacked': 1,
'annoline1': 'St. Albans',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.75867603727717,
40.69444538522359,
-73.75867603727717,
40.69444538522359]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.167',
'geometry': {'type': 'Point',
'coordinates': [-73.77258787620906, 40.67521139591733]},
'geometry_name': 'geom',
'properties': {'name': 'Rochdale',
'stacked': 1,
'annoline1': 'Rochdale',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.77258787620906,
40.67521139591733,
-73.77258787620906,
40.67521139591733]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.168',
'geometry': {'type': 'Point',
```



```
'bbox': [-73.75497968043872,
40.603134432500894,
-73.75497968043872,
40.603134432500894]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.172',
'geometry': {'type': 'Point',
'coordinates': [-73.8200548911032, 40.60302658351238]},  
'geometry_name': 'geom',
'properties': {'name': 'Broad Channel',
'stacked': 2,
'annoline1': 'Broad',
'annoline2': 'Channel',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.8200548911032,
40.60302658351238,
-73.8200548911032,
40.60302658351238]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.173',
'geometry': {'type': 'Point',
'coordinates': [-73.92551196994168, 40.55740128845452]},  
'geometry_name': 'geom',
'properties': {'name': 'Breezy Point',
'stacked': 2,
'annoline1': 'Breezy',
'annoline2': 'Point',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.92551196994168,
40.55740128845452,
-73.92551196994168,
40.55740128845452]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.174',
'geometry': {'type': 'Point',
'coordinates': [-73.90228960391673, 40.775923015642896]},  
'geometry_name': 'geom',
'properties': {'name': 'Steinway',
'stacked': 1,
'annoline1': 'Steinway',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.90228960391673,
40.775923015642896,
-73.90228960391673,
40.775923015642896]}},  
{'type': 'Feature',
'id': 'nyu_2451_34572.175',
'geometry': {'type': 'Point',
'coordinates': [-73.80436451720988, 40.79278140360048]},  
'geometry_name': 'geom',
```

```
'properties': {'name': 'Beechhurst',
 'stacked': 1,
 'annoline1': 'Beechhurst',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.80436451720988,
 40.79278140360048,
 -73.80436451720988,
 40.79278140360048]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.176',
 'geometry': {'type': 'Point',
 'coordinates': [-73.7768022262158, 40.782842806245554]},
 'geometry_name': 'geom',
 'properties': {'name': 'Bay Terrace',
 'stacked': 2,
 'annoline1': 'Bay',
 'annoline2': 'Terrace',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.7768022262158,
 40.782842806245554,
 -73.7768022262158,
 40.782842806245554]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.177',
 'geometry': {'type': 'Point',
 'coordinates': [-73.77613282391705, 40.595641807368494]},
 'geometry_name': 'geom',
 'properties': {'name': 'Edgemere',
 'stacked': 1,
 'annoline1': 'Edgemere',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.77613282391705,
 40.595641807368494,
 -73.77613282391705,
 40.595641807368494]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.178',
 'geometry': {'type': 'Point',
 'coordinates': [-73.79199233136943, 40.58914394372971]},
 'geometry_name': 'geom',
 'properties': {'name': 'Arverne',
 'stacked': 1,
 'annoline1': 'Arverne',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.79199233136943,
 40.58914394372971,
```

```
-73.79199233136943,
40.58914394372971]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.179',
'geometry': {'type': 'Point',
'coordinates': [-73.82236121088751, 40.582801696845586]},
'geometry_name': 'geom',
'properties': {'name': 'Rockaway Beach',
'stacked': 2,
'annoline1': 'Rockaway',
'annoline2': 'Beach',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.82236121088751,
40.582801696845586,
-73.82236121088751,
40.582801696845586]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.180',
'geometry': {'type': 'Point',
'coordinates': [-73.85754672410827, 40.572036730217015]},
'geometry_name': 'geom',
'properties': {'name': 'Neponsit',
'stacked': 1,
'annoline1': 'Neponsit',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.85754672410827,
40.572036730217015,
-73.85754672410827,
40.572036730217015]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.181',
'geometry': {'type': 'Point',
'coordinates': [-73.81276269135866, 40.764126122614066]},
'geometry_name': 'geom',
'properties': {'name': 'Murray Hill',
'stacked': 2,
'annoline1': 'Murray',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.81276269135866,
40.764126122614066,
-73.81276269135866,
40.764126122614066]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.182',
'geometry': {'type': 'Point',
'coordinates': [-73.70884705889246, 40.741378421945434]},
'geometry_name': 'geom',
'properties': {'name': 'Floral Park',
'stacked': 1,
```

```
'annoline1': 'Floral Park',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.70884705889246,
40.741378421945434,
-73.70884705889246,
40.741378421945434]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.183',
'geometry': {'type': 'Point',
'coordinates': [-73.76714166714729, 40.7209572076444]},
'geometry_name': 'geom',
'properties': {'name': 'Holliswood',
'stacked': 1,
'annoline1': 'Holliswood',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.76714166714729,
40.7209572076444,
-73.76714166714729,
40.7209572076444]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.184',
'geometry': {'type': 'Point',
'coordinates': [-73.7872269693666, 40.71680483014613]},
'geometry_name': 'geom',
'properties': {'name': 'Jamaica Estates',
'stacked': 2,
'annoline1': 'Jamaica',
'annoline2': 'Estates',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.7872269693666,
40.71680483014613,
-73.7872269693666,
40.71680483014613]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.185',
'geometry': {'type': 'Point',
'coordinates': [-73.82580915110559, 40.7445723092867]},
'geometry_name': 'geom',
'properties': {'name': 'Queensboro Hill',
'stacked': 2,
'annoline1': 'Queensboro',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.82580915110559,
40.7445723092867,
-73.82580915110559,
40.7445723092867]}},
```

```
{'type': 'Feature',
'id': 'nyu_2451_34572.186',
'geometry': {'type': 'Point',
'coordinates': [-73.79760300912672, 40.723824901829204]},
'geometry_name': 'geom',
'properties': {'name': 'Hillcrest',
'stacked': 1,
'annoline1': 'Hillcrest',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.79760300912672,
40.723824901829204,
-73.79760300912672,
40.723824901829204]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.187',
'geometry': {'type': 'Point',
'coordinates': [-73.93157506072878, 40.761704526054146]},
'geometry_name': 'geom',
'properties': {'name': 'Ravenswood',
'stacked': 1,
'annoline1': 'Ravenswood',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.93157506072878,
40.761704526054146,
-73.93157506072878,
40.761704526054146]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.188',
'geometry': {'type': 'Point',
'coordinates': [-73.84963782402441, 40.66391841925139]},
'geometry_name': 'geom',
'properties': {'name': 'Lindenwood',
'stacked': 1,
'annoline1': 'Lindenwood',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.84963782402441,
40.66391841925139,
-73.84963782402441,
40.66391841925139]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.189',
'geometry': {'type': 'Point',
'coordinates': [-73.74025607989822, 40.66788389660247]},
'geometry_name': 'geom',
'properties': {'name': 'Laurelton',
'stacked': 1,
'annoline1': 'Laurelton',
'annoline2': None,
```

```
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.74025607989822,
 40.66788389660247,
 -73.74025607989822,
 40.66788389660247]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.190',
'geometry': {'type': 'Point',
'coordinates': [-73.8625247141374, 40.736074570830795]},
'geometry_name': 'geom',
'properties': {'name': 'Lefrak City',
'stacked': 2,
'annoline1': 'Lefrak',
'annoline2': 'City',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.8625247141374,
 40.736074570830795,
 -73.8625247141374,
 40.736074570830795]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.191',
'geometry': {'type': 'Point',
'coordinates': [-73.8540175039252, 40.57615556543109]},
'geometry_name': 'geom',
'properties': {'name': 'Belle Harbor',
'stacked': 1,
'annoline1': 'Belle Harbor',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.8540175039252,
 40.57615556543109,
 -73.8540175039252,
 40.57615556543109]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.192',
'geometry': {'type': 'Point',
'coordinates': [-73.84153370226186, 40.58034295646131]},
'geometry_name': 'geom',
'properties': {'name': 'Rockaway Park',
'stacked': 1,
'annoline1': 'Rockaway Park',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.84153370226186,
 40.58034295646131,
 -73.84153370226186,
 40.58034295646131]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.193',
```

```
'geometry': {'type': 'Point',
  'coordinates': [-73.79664750844047, 40.59771061565768]},
'geometry_name': 'geom',
'properties': {'name': 'Somerville',
  'stacked': 1,
  'annoline1': 'Somerville',
  'annoline2': None,
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Queens',
  'bbox': [-73.79664750844047,
  40.59771061565768,
  -73.79664750844047,
  40.59771061565768]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.194',
  'geometry': {'type': 'Point',
    'coordinates': [-73.75175310731153, 40.66000322733613]},
  'geometry_name': 'geom',
  'properties': {'name': 'Brookville',
    'stacked': 1,
    'annoline1': 'Brookville',
    'annoline2': None,
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Queens',
    'bbox': [-73.75175310731153,
    40.66000322733613,
    -73.75175310731153,
    40.66000322733613]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.195',
  'geometry': {'type': 'Point',
    'coordinates': [-73.73889198912481, 40.73301404027834]},
  'geometry_name': 'geom',
  'properties': {'name': 'Bellaire',
    'stacked': 1,
    'annoline1': 'Bellaire',
    'annoline2': None,
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Queens',
    'bbox': [-73.73889198912481,
    40.73301404027834,
    -73.73889198912481,
    40.73301404027834]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.196',
  'geometry': {'type': 'Point',
    'coordinates': [-73.85751790676447, 40.7540709990489]},
  'geometry_name': 'geom',
  'properties': {'name': 'North Corona',
    'stacked': 2,
    'annoline1': 'North',
    'annoline2': 'Corona',
    'annoline3': None,
    'annoangle': 0.0,
```

```
'borough': 'Queens',
'bbox': [-73.85751790676447,
40.7540709990489,
-73.85751790676447,
40.7540709990489]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.197',
'geometry': {'type': 'Point',
'coordinates': [-73.8410221123401, 40.7146110815117]},
'geometry_name': 'geom',
'properties': {'name': 'Forest Hills Gardens',
'stacked': 3,
'annoline1': 'Forest',
'annoline2': 'Hills',
'annoline3': 'Gardens',
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.8410221123401,
40.7146110815117,
-73.8410221123401,
40.7146110815117]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.198',
'geometry': {'type': 'Point',
'coordinates': [-74.07935312512797, 40.6449815710044]},
'geometry_name': 'geom',
'properties': {'name': 'St. George',
'stacked': 2,
'annoline1': 'St.',
'annoline2': 'George',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.07935312512797,
40.6449815710044,
-74.07935312512797,
40.6449815710044]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.199',
'geometry': {'type': 'Point',
'coordinates': [-74.08701650516625, 40.64061455913511]},
'geometry_name': 'geom',
'properties': {'name': 'New Brighton',
'stacked': 2,
'annoline1': 'New',
'annoline2': 'Brighton',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.08701650516625,
40.64061455913511,
-74.08701650516625,
40.64061455913511]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.200',
'geometry': {'type': 'Point',
'coordinates': [-74.07790192660066, 40.62692762538176]},
```

```
'geometry_name': 'geom',
'properties': {'name': 'Stapleton',
 'stacked': 1,
 'annoline1': 'Stapleton',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.07790192660066,
  40.62692762538176,
  -74.07790192660066,
  40.62692762538176]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.201',
 'geometry': {'type': 'Point',
  'coordinates': [-74.06980526716141, 40.61530494652761]},
 'geometry_name': 'geom',
 'properties': {'name': 'Rosebank',
 'stacked': 1,
 'annoline1': 'Rosebank',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.06980526716141,
  40.61530494652761,
  -74.06980526716141,
  40.61530494652761]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.202',
 'geometry': {'type': 'Point',
  'coordinates': [-74.1071817826561, 40.63187892654607]},
 'geometry_name': 'geom',
 'properties': {'name': 'West Brighton',
 'stacked': 2,
 'annoline1': 'West',
 'annoline2': 'Brighton',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.1071817826561,
  40.63187892654607,
  -74.1071817826561,
  40.63187892654607]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.203',
 'geometry': {'type': 'Point',
  'coordinates': [-74.08724819983729, 40.624184791313006]},
 'geometry_name': 'geom',
 'properties': {'name': 'Grymes Hill',
 'stacked': 2,
 'annoline1': 'Grymes',
 'annoline2': 'Hill',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.08724819983729,
```

```
40.624184791313006,  
-74.08724819983729,  
40.624184791313006]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.204',  
'geometry': {'type': 'Point',  
'coordinates': [-74.1113288180088, 40.59706851814673]},  
'geometry_name': 'geom',  
'properties': {'name': 'Todt Hill',  
'stacked': 2,  
'annoline1': 'Todt',  
'annoline2': 'Hill',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',  
'bbox': [-74.1113288180088,  
40.59706851814673,  
-74.1113288180088,  
40.59706851814673]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.205',  
'geometry': {'type': 'Point',  
'coordinates': [-74.0795529253982, 40.58024741350956]},  
'geometry_name': 'geom',  
'properties': {'name': 'South Beach',  
'stacked': 2,  
'annoline1': 'South',  
'annoline2': 'Beach',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',  
'bbox': [-74.0795529253982,  
40.58024741350956,  
-74.0795529253982,  
40.58024741350956]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.206',  
'geometry': {'type': 'Point',  
'coordinates': [-74.12943426797008, 40.63366930554365]},  
'geometry_name': 'geom',  
'properties': {'name': 'Port Richmond',  
'stacked': 2,  
'annoline1': 'Port',  
'annoline2': 'Richmond',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',  
'bbox': [-74.12943426797008,  
40.63366930554365,  
-74.12943426797008,  
40.63366930554365]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.207',  
'geometry': {'type': 'Point',  
'coordinates': [-74.15008537046981, 40.632546390481124]},  
'geometry_name': 'geom',  
'properties': {'name': "Mariner's Harbor",
```

```
'stacked': 2,
'annoline1': "Mariner's",
'annoline2': 'Harbor',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.15008537046981,
 40.632546390481124,
 -74.15008537046981,
 40.632546390481124]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.208',
'geometry': {'type': 'Point',
'coordinates': [-74.17464532993542, 40.63968297845542]},
'geometry_name': 'geom',
'properties': {'name': 'Port Ivory',
'stacked': 2,
'annoline1': 'Port',
'annoline2': 'Ivory',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.17464532993542,
 40.63968297845542,
 -74.17464532993542,
 40.63968297845542]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.209',
'geometry': {'type': 'Point',
'coordinates': [-74.11918058534842, 40.61333593766742]},
'geometry_name': 'geom',
'properties': {'name': 'Castleton Corners',
'stacked': 2,
'annoline1': 'Castleton',
'annoline2': 'Corners',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.11918058534842,
 40.61333593766742,
 -74.11918058534842,
 40.61333593766742]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.210',
'geometry': {'type': 'Point',
'coordinates': [-74.16496031329827, 40.594252379161695]},
'geometry_name': 'geom',
'properties': {'name': 'New Springville',
'stacked': 2,
'annoline1': 'New',
'annoline2': 'Springville',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.16496031329827,
 40.594252379161695,
 -74.16496031329827,
```



```
'annoline2': 'Kill$',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.14932381490992,
40.549480228713605,
-74.14932381490992,
40.549480228713605]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.215',
'geometry': {'type': 'Point',
'coordinates': [-74.1643308041936, 40.542230747450745]},
'geometry_name': 'geom',
'properties': {'name': 'Eltingville',
'stacked': 1,
'annoline1': 'Eltingville',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.1643308041936,
40.542230747450745,
-74.1643308041936,
40.542230747450745]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.216',
'geometry': {'type': 'Point',
'coordinates': [-74.17854866165878, 40.53811417474507]},
'geometry_name': 'geom',
'properties': {'name': 'Annadale',
'stacked': 1,
'annoline1': 'Annadale',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.17854866165878,
40.53811417474507,
-74.17854866165878,
40.53811417474507]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.217',
'geometry': {'type': 'Point',
'coordinates': [-74.20524582480326, 40.541967622888755]},
'geometry_name': 'geom',
'properties': {'name': 'Woodrow',
'stacked': 1,
'annoline1': 'Woodrow',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.20524582480326,
40.541967622888755,
-74.20524582480326,
40.541967622888755]}},
{'type': 'Feature',
```

```
'id': 'nyu_2451_34572.218',
'geometry': {'type': 'Point',
 'coordinates': [-74.24656934235283, 40.50533376115642]},
'geometry_name': 'geom',
'properties': {'name': 'Tottenville',
 'stacked': 1,
 'annoline1': 'Tottenville',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.24656934235283,
 40.50533376115642,
 -74.24656934235283,
 40.50533376115642]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.219',
'geometry': {'type': 'Point',
 'coordinates': [-74.08055351790115, 40.637316067110326]},
'geometry_name': 'geom',
'properties': {'name': 'Tompkinsville',
 'stacked': 1,
 'annoline1': 'Tompkinsville',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.08055351790115,
 40.637316067110326,
 -74.08055351790115,
 40.637316067110326]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.220',
'geometry': {'type': 'Point',
 'coordinates': [-74.09629029235458, 40.61919310792676]},
'geometry_name': 'geom',
'properties': {'name': 'Silver Lake',
 'stacked': 2,
 'annoline1': 'Silver',
 'annoline2': 'Lake',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.09629029235458,
 40.61919310792676,
 -74.09629029235458,
 40.61919310792676]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.221',
'geometry': {'type': 'Point',
 'coordinates': [-74.0971255217853, 40.61276015756489]},
'geometry_name': 'geom',
'properties': {'name': 'Sunnyside',
 'stacked': 1,
 'annoline1': 'Sunnyside',
 'annoline2': None,
 'annoline3': None,
```

```
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.0971255217853,
40.61276015756489,
-74.0971255217853,
40.61276015756489]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.222',
'geometry': {'type': 'Point',
'coordinates': [-73.96101312466779, 40.643675183340974]},
'geometry_name': 'geom',
'properties': {'name': 'Ditmas Park',
'stacked': 2,
'annoline1': 'Ditmas',
'annoline2': 'Park',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.96101312466779,
40.643675183340974,
-73.96101312466779,
40.643675183340974]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.223',
'geometry': {'type': 'Point',
'coordinates': [-73.93718680559314, 40.66094656188111]},
'geometry_name': 'geom',
'properties': {'name': 'Wingate',
'stacked': 1,
'annoline1': 'Wingate',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.93718680559314,
40.66094656188111,
-73.93718680559314,
40.66094656188111]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.224',
'geometry': {'type': 'Point',
'coordinates': [-73.92688212616955, 40.655572313280764]},
'geometry_name': 'geom',
'properties': {'name': 'Rugby',
'stacked': 1,
'annoline1': 'Rugby',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.92688212616955,
40.655572313280764,
-73.92688212616955,
40.655572313280764]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.225',
'geometry': {'type': 'Point',
```

```
'coordinates': [-74.08015734936296, 40.60919044434558]},  
'geometry_name': 'geom',  
'properties': {'name': 'Park Hill',  
'stacked': 2,  
'annoline1': 'Park',  
'annoline2': 'Hill',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',  
'bbox': [-74.08015734936296,  
40.60919044434558,  
-74.08015734936296,  
40.60919044434558]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.226',  
'geometry': {'type': 'Point',  
'coordinates': [-74.13304143951704, 40.62109047275409]},  
'geometry_name': 'geom',  
'properties': {'name': 'Westerleigh',  
'stacked': 1,  
'annoline1': 'Westerleigh',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',  
'bbox': [-74.13304143951704,  
40.62109047275409,  
-74.13304143951704,  
40.62109047275409]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.227',  
'geometry': {'type': 'Point',  
'coordinates': [-74.15315246387762, 40.620171512231884]},  
'geometry_name': 'geom',  
'properties': {'name': 'Graniteville',  
'stacked': 1,  
'annoline1': 'Graniteville',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',  
'bbox': [-74.15315246387762,  
40.620171512231884,  
-74.15315246387762,  
40.620171512231884]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.228',  
'geometry': {'type': 'Point',  
'coordinates': [-74.16510420241124, 40.63532509911492]},  
'geometry_name': 'geom',  
'properties': {'name': 'Arlington',  
'stacked': 1,  
'annoline1': 'Arlington',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',
```

```
'bbox': [-74.16510420241124,
40.63532509911492,
-74.16510420241124,
40.63532509911492]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.229',
'geometry': {'type': 'Point',
'coordinates': [-74.06712363225574, 40.596312571276734]},
'geometry_name': 'geom',
'properties': {'name': 'Arrochar',
'stacked': 1,
'annoline1': 'Arrochar',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.06712363225574,
40.596312571276734,
-74.06712363225574,
40.596312571276734]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.230',
'geometry': {'type': 'Point',
'coordinates': [-74.0766743627905, 40.59826835959991]},
'geometry_name': 'geom',
'properties': {'name': 'Grasmere',
'stacked': 1,
'annoline1': 'Grasmere',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.0766743627905,
40.59826835959991,
-74.0766743627905,
40.59826835959991]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.231',
'geometry': {'type': 'Point',
'coordinates': [-74.08751118005578, 40.59632891379513]},
'geometry_name': 'geom',
'properties': {'name': 'Old Town',
'stacked': 2,
'annoline1': 'Old',
'annoline2': 'Town',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.08751118005578,
40.59632891379513,
-74.08751118005578,
40.59632891379513]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.232',
'geometry': {'type': 'Point',
'coordinates': [-74.09639905312521, 40.588672948199275]},
'geometry_name': 'geom',
```

```
'properties': {'name': 'Dongan Hills',
 'stacked': 2,
 'annoline1': 'Dongan',
 'annoline2': 'Hills',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.09639905312521,
 40.588672948199275,
 -74.09639905312521,
 40.588672948199275]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.233',
 'geometry': {'type': 'Point',
 'coordinates': [-74.09348266303591, 40.57352690574283]},
 'geometry_name': 'geom',
 'properties': {'name': 'Midland Beach',
 'stacked': 2,
 'annoline1': 'Midland',
 'annoline2': 'Beach',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.09348266303591,
 40.57352690574283,
 -74.09348266303591,
 40.57352690574283]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.234',
 'geometry': {'type': 'Point',
 'coordinates': [-74.10585598545434, 40.57621558711788]},
 'geometry_name': 'geom',
 'properties': {'name': 'Grant City',
 'stacked': 2,
 'annoline1': 'Grant',
 'annoline2': 'City',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.10585598545434,
 40.57621558711788,
 -74.10585598545434,
 40.57621558711788]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.235',
 'geometry': {'type': 'Point',
 'coordinates': [-74.10432707469124, 40.56425549307335]},
 'geometry_name': 'geom',
 'properties': {'name': 'New Dorp Beach',
 'stacked': 3,
 'annoline1': 'New',
 'annoline2': 'Dorp',
 'annoline3': 'Beach',
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.10432707469124,
 40.56425549307335,
```

```
-74.10432707469124,
40.56425549307335]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.236',
'geometry': {'type': 'Point',
'coordinates': [-74.13916622175768, 40.55398800858462]},
'geometry_name': 'geom',
'properties': {'name': 'Bay Terrace',
'stacked': 2,
'annoline1': 'Bay',
'annoline2': 'Terrace',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.13916622175768,
40.55398800858462,
-74.13916622175768,
40.55398800858462]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.237',
'geometry': {'type': 'Point',
'coordinates': [-74.19174105747814, 40.531911920489605]},
'geometry_name': 'geom',
'properties': {'name': 'Huguenot',
'stacked': 1,
'annoline1': 'Huguenot',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.19174105747814,
40.531911920489605,
-74.19174105747814,
40.531911920489605]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.238',
'geometry': {'type': 'Point',
'coordinates': [-74.21983106616777, 40.524699376118136]},
'geometry_name': 'geom',
'properties': {'name': 'Pleasant Plains',
'stacked': 2,
'annoline1': 'Pleasant',
'annoline2': 'Plains',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.21983106616777,
40.524699376118136,
-74.21983106616777,
40.524699376118136]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.239',
'geometry': {'type': 'Point',
'coordinates': [-74.22950350260027, 40.50608165346305]},
'geometry_name': 'geom',
'properties': {'name': 'Butler Manor',
'stacked': 2,
```

```
'annoline1': 'Butler',
'annoline2': 'Manor',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.22950350260027,
40.50608165346305,
-74.22950350260027,
40.50608165346305]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.240',
'geometry': {'type': 'Point',
'coordinates': [-74.23215775896526, 40.53053148283314]},
'geometry_name': 'geom',
'properties': {'name': 'Charleston',
'stacked': 1,
'annoline1': 'Charleston',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.23215775896526,
40.53053148283314,
-74.23215775896526,
40.53053148283314]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.241',
'geometry': {'type': 'Point',
'coordinates': [-74.21572851113952, 40.54940400650072]},
'geometry_name': 'geom',
'properties': {'name': 'Rossville',
'stacked': 1,
'annoline1': 'Rossville',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.21572851113952,
40.54940400650072,
-74.21572851113952,
40.54940400650072]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.242',
'geometry': {'type': 'Point',
'coordinates': [-74.18588674583893, 40.54928582278321]},
'geometry_name': 'geom',
'properties': {'name': 'Arden Heights',
'stacked': 2,
'annoline1': 'Arden',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.18588674583893,
40.54928582278321,
-74.18588674583893,
40.54928582278321]}},
```

```
{'type': 'Feature',
'id': 'nyu_2451_34572.243',
'geometry': {'type': 'Point',
'coordinates': [-74.17079414786092, 40.555295236173194]},
'geometry_name': 'geom',
'properties': {'name': 'Greenridge',
'stacked': 1,
'annoline1': 'Greenridge',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.17079414786092,
40.555295236173194,
-74.17079414786092,
40.555295236173194]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.244',
'geometry': {'type': 'Point',
'coordinates': [-74.15902208156601, 40.58913894875281]},
'geometry_name': 'geom',
'properties': {'name': 'Heartland Village',
'stacked': 2,
'annoline1': 'Heartland',
'annoline2': 'Village',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.15902208156601,
40.58913894875281,
-74.15902208156601,
40.58913894875281]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.245',
'geometry': {'type': 'Point',
'coordinates': [-74.1895604551969, 40.59472602746295]},
'geometry_name': 'geom',
'properties': {'name': 'Chelsea',
'stacked': 1,
'annoline1': 'Chelsea',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.1895604551969,
40.59472602746295,
-74.1895604551969,
40.59472602746295]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.246',
'geometry': {'type': 'Point',
'coordinates': [-74.18725638381567, 40.60577868452358]},
'geometry_name': 'geom',
'properties': {'name': 'Bloomfield',
'stacked': 1,
'annoline1': 'Bloomfield',
'annoline2': None,
```

```
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.18725638381567,
 40.60577868452358,
 -74.18725638381567,
 40.60577868452358]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.247',
'geometry': {'type': 'Point',
 'coordinates': [-74.15940948657122, 40.6095918004203]},
'geometry_name': 'geom',
'properties': {'name': 'Bulls Head',
 'stacked': 2,
 'annoline1': 'Bulls',
 'annoline2': 'Head',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
'bbox': [-74.15940948657122,
 40.6095918004203,
 -74.15940948657122,
 40.6095918004203]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.248',
'geometry': {'type': 'Point',
 'coordinates': [-73.95325646837112, 40.7826825671257]},
'geometry_name': 'geom',
'properties': {'name': 'Carnegie Hill',
 'stacked': 2,
 'annoline1': 'Carnegie',
 'annoline2': 'Hill',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
'bbox': [-73.95325646837112,
 40.7826825671257,
 -73.95325646837112,
 40.7826825671257]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.249',
'geometry': {'type': 'Point',
 'coordinates': [-73.98843368023597, 40.72325901885768]},
'geometry_name': 'geom',
'properties': {'name': 'Noho',
 'stacked': 1,
 'annoline1': 'Noho',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
'bbox': [-73.98843368023597,
 40.72325901885768,
 -73.98843368023597,
 40.72325901885768]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.250',
```

```
'geometry': {'type': 'Point',
  'coordinates': [-74.00541529873355, 40.71522892046282]},
'geometry_name': 'geom',
'properties': {'name': 'Civic Center',
  'stacked': 2,
  'annoline1': 'Civic',
  'annoline2': 'Center',
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Manhattan',
  'bbox': [-74.00541529873355,
  40.71522892046282,
  -74.00541529873355,
  40.71522892046282]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.251',
  'geometry': {'type': 'Point',
    'coordinates': [-73.98871313285247, 40.7485096643122]},
  'geometry_name': 'geom',
  'properties': {'name': 'Midtown South',
    'stacked': 2,
    'annoline1': 'Midtown',
    'annoline2': 'South',
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Manhattan',
    'bbox': [-73.98871313285247,
    40.7485096643122,
    -73.98871313285247,
    40.7485096643122]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.252',
  'geometry': {'type': 'Point',
    'coordinates': [-74.1340572986257, 40.56960594275505]},
  'geometry_name': 'geom',
  'properties': {'name': 'Richmond Town',
    'stacked': 2,
    'annoline1': 'Richmond',
    'annoline2': 'Town',
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Staten Island',
    'bbox': [-74.1340572986257,
    40.56960594275505,
    -74.1340572986257,
    40.56960594275505]}},
{'type': 'Feature',
  'id': 'nyu_2451_34572.253',
  'geometry': {'type': 'Point',
    'coordinates': [-74.06667766061771, 40.60971934079284]},
  'geometry_name': 'geom',
  'properties': {'name': 'Shore Acres',
    'stacked': 2,
    'annoline1': 'Shore',
    'annoline2': 'Acres',
    'annoline3': None,
    'annoangle': 0.0,
```

```
'borough': 'Staten Island',
'bbox': [-74.06667766061771,
40.60971934079284,
-74.06667766061771,
40.60971934079284]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.254',
'geometry': {'type': 'Point',
'coordinates': [-74.072642445484, 40.61917845202843]},
'geometry_name': 'geom',
'properties': {'name': 'Clifton',
'stacked': 1,
'annoline1': 'Clifton',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.072642445484,
40.61917845202843,
-74.072642445484,
40.61917845202843]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.255',
'geometry': {'type': 'Point',
'coordinates': [-74.08402364740358, 40.6044731896879]},
'geometry_name': 'geom',
'properties': {'name': 'Concord',
'stacked': 1,
'annoline1': 'Concord',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.08402364740358,
40.6044731896879,
-74.08402364740358,
40.6044731896879]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.256',
'geometry': {'type': 'Point',
'coordinates': [-74.09776206972522, 40.606794394801]},
'geometry_name': 'geom',
'properties': {'name': 'Emerson Hill',
'stacked': 2,
'annoline1': 'Emerson',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.09776206972522,
40.606794394801,
-74.09776206972522,
40.606794394801]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.257',
'geometry': {'type': 'Point',
'coordinates': [-74.09805062373887, 40.63563000681151]},
```

```
'geometry_name': 'geom',
'properties': {'name': 'Randall Manor',
 'stacked': 2,
 'annoline1': 'Randall',
 'annoline2': 'Manor',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.09805062373887,
  40.63563000681151,
  -74.09805062373887,
  40.63563000681151]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.258',
 'geometry': {'type': 'Point',
  'coordinates': [-74.18622331749823, 40.63843283794795]},
 'geometry_name': 'geom',
 'properties': {'name': 'Howland Hook',
 'stacked': 2,
 'annoline1': 'Howland',
 'annoline2': 'Hook',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.18622331749823,
  40.63843283794795,
  -74.18622331749823,
  40.63843283794795]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.259',
 'geometry': {'type': 'Point',
  'coordinates': [-74.1418167896889, 40.630146741193826]},
 'geometry_name': 'geom',
 'properties': {'name': 'Elm Park',
 'stacked': 2,
 'annoline1': 'Elm',
 'annoline2': 'Park',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.1418167896889,
  40.630146741193826,
  -74.1418167896889,
  40.630146741193826]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.260',
 'geometry': {'type': 'Point',
  'coordinates': [-73.91665331978048, 40.652117451793494]},
 'geometry_name': 'geom',
 'properties': {'name': 'Remsen Village',
 'stacked': 2,
 'annoline1': 'Remsen',
 'annoline2': 'Village',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-73.91665331978048,
```

```
40.652117451793494,  
-73.91665331978048,  
40.652117451793494]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.261',  
'geometry': {'type': 'Point',  
'coordinates': [-73.88511776379292, 40.6627442796966]},  
'geometry_name': 'geom',  
'properties': {'name': 'New Lots',  
'stacked': 2,  
'annoline1': 'New',  
'annoline2': 'Lots',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.88511776379292,  
40.6627442796966,  
-73.88511776379292,  
40.6627442796966]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.262',  
'geometry': {'type': 'Point',  
'coordinates': [-73.90233474295836, 40.63131755039667]},  
'geometry_name': 'geom',  
'properties': {'name': 'Paerdegat Basin',  
'stacked': 2,  
'annoline1': 'Paerdegat',  
'annoline2': 'Basin',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.90233474295836,  
40.63131755039667,  
-73.90233474295836,  
40.63131755039667]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.263',  
'geometry': {'type': 'Point',  
'coordinates': [-73.91515391550404, 40.61597423962336]},  
'geometry_name': 'geom',  
'properties': {'name': 'Mill Basin',  
'stacked': 2,  
'annoline1': 'Mill',  
'annoline2': 'Basin',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.91515391550404,  
40.61597423962336,  
-73.91515391550404,  
40.61597423962336]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.264',  
'geometry': {'type': 'Point',  
'coordinates': [-73.79646462081593, 40.71145964370482]},  
'geometry_name': 'geom',  
'properties': {'name': 'Jamaica Hills',
```

```
'stacked': 2,
'annoline1': 'Jamaica',
'annoline2': 'Hills',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.79646462081593,
40.71145964370482,
-73.79646462081593,
40.71145964370482]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.265',
'geometry': {'type': 'Point',
'coordinates': [-73.79671678028349, 40.73350025429757]},
'geometry_name': 'geom',
'properties': {'name': 'Utopia',
'stacked': 1,
'annoline1': 'Utopia',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.79671678028349,
40.73350025429757,
-73.79671678028349,
40.73350025429757]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.266',
'geometry': {'type': 'Point',
'coordinates': [-73.80486120040537, 40.73493618075478]},
'geometry_name': 'geom',
'properties': {'name': 'Pomonok',
'stacked': 1,
'annoline1': 'Pomonok',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.80486120040537,
40.73493618075478,
-73.80486120040537,
40.73493618075478]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.267',
'geometry': {'type': 'Point',
'coordinates': [-73.89467996270574, 40.7703173929982]},
'geometry_name': 'geom',
'properties': {'name': 'Astoria Heights',
'stacked': 2,
'annoline1': 'Astoria',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.89467996270574,
40.7703173929982,
-73.89467996270574,
```



```
'annoline2': 'Hope',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.90829930881988,
40.84884160724665,
-73.90829930881988,
40.84884160724665]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.272',
'geometry': {'type': 'Point',
'coordinates': [-73.96355614094303, 40.76028033131374]},
'geometry_name': 'geom',
'properties': {'name': 'Sutton Place',
'stacked': 2,
'annoline1': 'Sutton',
'annoline2': 'Place',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.96355614094303,
40.76028033131374,
-73.96355614094303,
40.76028033131374]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.273',
'geometry': {'type': 'Point',
'coordinates': [-73.95386782130745, 40.743414090073536]},
'geometry_name': 'geom',
'properties': {'name': 'Hunters Point',
'stacked': 2,
'annoline1': 'Hunters',
'annoline2': 'Point',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.95386782130745,
40.743414090073536,
-73.95386782130745,
40.743414090073536]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.274',
'geometry': {'type': 'Point',
'coordinates': [-73.96770824581834, 40.75204236950722]},
'geometry_name': 'geom',
'properties': {'name': 'Turtle Bay',
'stacked': 2,
'annoline1': 'Turtle',
'annoline2': 'Bay',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-73.96770824581834,
40.75204236950722,
-73.96770824581834,
40.75204236950722]}},
{'type': 'Feature',
```

```
'id': 'nyu_2451_34572.275',
'geometry': {'type': 'Point',
 'coordinates': [-73.97121928722265, 40.746917410740195]},
'geometry_name': 'geom',
'properties': {'name': 'Tudor City',
 'stacked': 2,
 'annoline1': 'Tudor',
 'annoline2': 'City',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
 'bbox': [-73.97121928722265,
 40.746917410740195,
 -73.97121928722265,
 40.746917410740195]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.276',
'geometry': {'type': 'Point',
 'coordinates': [-73.97405170469203, 40.73099955477061]},
'geometry_name': 'geom',
'properties': {'name': 'Stuyvesant Town',
 'stacked': 2,
 'annoline1': 'Stuyvesant',
 'annoline2': 'Town',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
 'bbox': [-73.97405170469203,
 40.73099955477061,
 -73.97405170469203,
 40.73099955477061]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.277',
'geometry': {'type': 'Point',
 'coordinates': [-73.9909471052826, 40.739673047638426]},
'geometry_name': 'geom',
'properties': {'name': 'Flatiron',
 'stacked': 1,
 'annoline1': 'Flatiron',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Manhattan',
 'bbox': [-73.9909471052826,
 40.739673047638426,
 -73.9909471052826,
 40.739673047638426]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.278',
'geometry': {'type': 'Point',
 'coordinates': [-73.91819286431682, 40.74565180608076]},
'geometry_name': 'geom',
'properties': {'name': 'Sunnyside Gardens',
 'stacked': 2,
 'annoline1': 'Sunnyside',
 'annoline2': 'Gardens',
 'annoline3': None,
```

```
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.91819286431682,
40.74565180608076,
-73.91819286431682,
40.74565180608076]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.279',
'geometry': {'type': 'Point',
'coordinates': [-73.93244235260178, 40.73725071694497]},
'geometry_name': 'geom',
'properties': {'name': 'Blissville',
'stacked': 1,
'annoline1': 'Blissville',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.93244235260178,
40.73725071694497,
-73.93244235260178,
40.73725071694497]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.280',
'geometry': {'type': 'Point',
'coordinates': [-73.99550751888415, 40.70328109093014]},
'geometry_name': 'geom',
'properties': {'name': 'Fulton Ferry',
'stacked': 2,
'annoline1': 'Fulton',
'annoline2': 'Ferry',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.99550751888415,
40.70328109093014,
-73.99550751888415,
40.70328109093014]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.281',
'geometry': {'type': 'Point',
'coordinates': [-73.98111603592393, 40.70332149882874]},
'geometry_name': 'geom',
'properties': {'name': 'Vinegar Hill',
'stacked': 2,
'annoline1': 'Vinegar',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.98111603592393,
40.70332149882874,
-73.98111603592393,
40.70332149882874]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.282',
'geometry': {'type': 'Point',
```

```
'coordinates': [-73.93053108817338, 40.67503986503237]},  
'geometry_name': 'geom',  
'properties': {'name': 'Weeksville',  
'stacked': 1,  
'annoline1': 'Weeksville',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.93053108817338,  
40.67503986503237,  
-73.93053108817338,  
40.67503986503237]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.283',  
'geometry': {'type': 'Point',  
'coordinates': [-73.90331684852599, 40.67786104769531]},  
'geometry_name': 'geom',  
'properties': {'name': 'Broadway Junction',  
'stacked': 2,  
'annoline1': 'Broadway',  
'annoline2': 'Junction',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.90331684852599,  
40.67786104769531,  
-73.90331684852599,  
40.67786104769531]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.284',  
'geometry': {'type': 'Point',  
'coordinates': [-73.9887528074504, 40.70317632822692]},  
'geometry_name': 'geom',  
'properties': {'name': 'Dumbo',  
'stacked': 1,  
'annoline1': 'Dumbo',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Brooklyn',  
'bbox': [-73.9887528074504,  
40.70317632822692,  
-73.9887528074504,  
40.70317632822692]}},  
{'type': 'Feature',  
'id': 'nyu_2451_34572.285',  
'geometry': {'type': 'Point',  
'coordinates': [-74.12059399718001, 40.60180957631444]},  
'geometry_name': 'geom',  
'properties': {'name': 'Manor Heights',  
'stacked': 2,  
'annoline1': 'Manor',  
'annoline2': 'Heights',  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Staten Island',
```

```
'bbox': [-74.12059399718001,  
40.60180957631444,  
-74.12059399718001,  
40.60180957631444]}}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.286',  
 'geometry': {'type': 'Point',  
  'coordinates': [-74.13208447484298, 40.60370692627371]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Willowbrook',  
  'stacked': 1,  
  'annoline1': 'Willowbrook',  
  'annoline2': None,  
  'annoline3': None,  
  'annoangle': 0.0,  
  'borough': 'Staten Island',  
  'bbox': [-74.13208447484298,  
40.60370692627371,  
-74.13208447484298,  
40.60370692627371]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.287',  
 'geometry': {'type': 'Point',  
  'coordinates': [-74.21776636068567, 40.541139922091766]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Sandy Ground',  
  'stacked': 2,  
  'annoline1': 'Sandy',  
  'annoline2': 'Ground',  
  'annoline3': None,  
  'annoangle': 0.0,  
  'borough': 'Staten Island',  
  'bbox': [-74.21776636068567,  
40.541139922091766,  
-74.21776636068567,  
40.541139922091766]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.288',  
 'geometry': {'type': 'Point',  
  'coordinates': [-74.12727240604946, 40.579118742961214]},  
 'geometry_name': 'geom',  
 'properties': {'name': 'Egbertville',  
  'stacked': 1,  
  'annoline1': 'Egbertville',  
  'annoline2': None,  
  'annoline3': None,  
  'annoangle': 0.0,  
  'borough': 'Staten Island',  
  'bbox': [-74.12727240604946,  
40.579118742961214,  
-74.12727240604946,  
40.579118742961214]}},  
{'type': 'Feature',  
 'id': 'nyu_2451_34572.289',  
 'geometry': {'type': 'Point',  
  'coordinates': [-73.89213760232822, 40.56737588957032]},  
 'geometry_name': 'geom',
```

```
'properties': {'name': 'Roxbury',
 'stacked': 1,
 'annoline1': 'Roxbury',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.89213760232822,
 40.56737588957032,
 -73.89213760232822,
 40.56737588957032]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.290',
 'geometry': {'type': 'Point',
 'coordinates': [-73.95918459428702, 40.598525095137255]},
 'geometry_name': 'geom',
 'properties': {'name': 'Homecrest',
 'stacked': 1,
 'annoline1': 'Homecrest',
 'annoline2': None,
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Brooklyn',
 'bbox': [-73.95918459428702,
 40.598525095137255,
 -73.95918459428702,
 40.598525095137255]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.291',
 'geometry': {'type': 'Point',
 'coordinates': [-73.88114319200604, 40.716414511158185]},
 'geometry_name': 'geom',
 'properties': {'name': 'Middle Village',
 'stacked': 2,
 'annoline1': 'Middle',
 'annoline2': 'Village',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Queens',
 'bbox': [-73.88114319200604,
 40.716414511158185,
 -73.88114319200604,
 40.716414511158185]}},
{'type': 'Feature',
 'id': 'nyu_2451_34572.292',
 'geometry': {'type': 'Point',
 'coordinates': [-74.20152556457658, 40.52626406734812]},
 'geometry_name': 'geom',
 'properties': {'name': "Prince's Bay",
 'stacked': 2,
 'annoline1': "Prince's",
 'annoline2': 'Bay',
 'annoline3': None,
 'annoangle': 0.0,
 'borough': 'Staten Island',
 'bbox': [-74.20152556457658,
 40.52626406734812,
```

```
-74.20152556457658,
40.52626406734812]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.293',
'geometry': {'type': 'Point',
'coordinates': [-74.13792663771568, 40.57650629379489]},
'geometry_name': 'geom',
'properties': {'name': 'Lighthouse Hill',
'stacked': 2,
'annoline1': 'Lighthouse',
'annoline2': 'Hill',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.13792663771568,
40.57650629379489,
-74.13792663771568,
40.57650629379489]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.294',
'geometry': {'type': 'Point',
'coordinates': [-74.22957080626941, 40.51954145748909]},
'geometry_name': 'geom',
'properties': {'name': 'Richmond Valley',
'stacked': 2,
'annoline1': 'Richmond',
'annoline2': 'Valley',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.22957080626941,
40.51954145748909,
-74.22957080626941,
40.51954145748909]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.295',
'geometry': {'type': 'Point',
'coordinates': [-73.82667757138641, 40.79060155670148]},
'geometry_name': 'geom',
'properties': {'name': 'Malba',
'stacked': 1,
'annoline1': 'Malba',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.82667757138641,
40.79060155670148,
-73.82667757138641,
40.79060155670148]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.296',
'geometry': {'type': 'Point',
'coordinates': [-73.890345709872, 40.6819989345173]},
'geometry_name': 'geom',
'properties': {'name': 'Highland Park',
'stacked': 2,
```

```
'annoline1': 'Highland',
'annoline2': 'Park',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.890345709872,
40.6819989345173,
-73.890345709872,
40.6819989345173]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.297',
'geometry': {'type': 'Point',
'coordinates': [-73.94841515328893, 40.60937770113766]},
'geometry_name': 'geom',
'properties': {'name': 'Madison',
'stacked': 1,
'annoline1': 'Madison',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.94841515328893,
40.60937770113766,
-73.94841515328893,
40.60937770113766]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.298',
'geometry': {'type': 'Point',
'coordinates': [-73.86172577555115, 40.85272297633017]},
'geometry_name': 'geom',
'properties': {'name': 'Bronxdale',
'stacked': 1,
'annoline1': 'Bronxdale',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.86172577555115,
40.85272297633017,
-73.86172577555115,
40.85272297633017]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.299',
'geometry': {'type': 'Point',
'coordinates': [-73.85931863221647, 40.86578787802982]},
'geometry_name': 'geom',
'properties': {'name': 'Allerton',
'stacked': 1,
'annoline1': 'Allerton',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.85931863221647,
40.86578787802982,
-73.85931863221647,
40.86578787802982]}},
```

```
{'type': 'Feature',
'id': 'nyu_2451_34572.300',
'geometry': {'type': 'Point',
'coordinates': [-73.90152264513144, 40.8703923914147]},
'geometry_name': 'geom',
'properties': {'name': 'Kingsbridge Heights',
'stacked': 2,
'annoline1': 'Kingsbridge',
'annoline2': 'Heights',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Bronx',
'bbox': [-73.90152264513144,
40.8703923914147,
-73.90152264513144,
40.8703923914147]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.301',
'geometry': {'type': 'Point',
'coordinates': [-73.94817709920184, 40.64692606658579]},
'geometry_name': 'geom',
'properties': {'name': 'Erasmus',
'stacked': 1,
'annoline1': 'Erasmus',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Brooklyn',
'bbox': [-73.94817709920184,
40.64692606658579,
-73.94817709920184,
40.64692606658579]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.302',
'geometry': {'type': 'Point',
'coordinates': [-74.00011136202637, 40.75665808227519]},
'geometry_name': 'geom',
'properties': {'name': 'Hudson Yards',
'stacked': 2,
'annoline1': 'Hudson',
'annoline2': 'Yards',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Manhattan',
'bbox': [-74.00011136202637,
40.75665808227519,
-74.00011136202637,
40.75665808227519]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.303',
'geometry': {'type': 'Point',
'coordinates': [-73.80553002968718, 40.58733774018741]},
'geometry_name': 'geom',
'properties': {'name': 'Hammels',
'stacked': 1,
'annoline1': 'Hammels',
'annoline2': None,
```

```
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.80553002968718,
 40.58733774018741,
 -73.80553002968718,
 40.58733774018741]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.304',
'geometry': {'type': 'Point',
'coordinates': [-73.76596781445627, 40.611321691283834]},
'geometry_name': 'geom',
'properties': {'name': 'Bayswater',
'stacked': 1,
'annoline1': 'Bayswater',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.76596781445627,
 40.611321691283834,
 -73.76596781445627,
 40.611321691283834]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.305',
'geometry': {'type': 'Point',
'coordinates': [-73.94563070334091, 40.756091297094706]},
'geometry_name': 'geom',
'properties': {'name': 'Queensbridge',
'stacked': 1,
'annoline1': 'Queensbridge',
'annoline2': None,
'annoline3': None,
'annoangle': 0.0,
'borough': 'Queens',
'bbox': [-73.94563070334091,
 40.756091297094706,
 -73.94563070334091,
 40.756091297094706]}},
{'type': 'Feature',
'id': 'nyu_2451_34572.306',
'geometry': {'type': 'Point',
'coordinates': [-74.08173992211962, 40.61731079252983]},
'geometry_name': 'geom',
'properties': {'name': 'Fox Hills',
'stacked': 2,
'annoline1': 'Fox',
'annoline2': 'Hills',
'annoline3': None,
'annoangle': 0.0,
'borough': 'Staten Island',
'bbox': [-74.08173992211962,
 40.61731079252983,
 -74.08173992211962,
 40.61731079252983]}],
'crs': {'type': 'name', 'properties': {'name': 'urn:ogc:def:crs:EPSG::4326'}}},
```

```
'bbox': [-74.2492599487305,  
40.5033187866211,  
-73.7061614990234,  
40.9105606079102]}
```

In [183]:

```
neighborhoods_data = newyork_data['features']
```

In [184]:

```
neighborhoods_data[0]
```

Out[184]:

```
{'type': 'Feature',  
'id': 'nyu_2451_34572.1',  
'geometry': {'type': 'Point',  
'coordinates': [-73.84720052054902, 40.89470517661]},  
'geometry_name': 'geom',  
'properties': {'name': 'Wakefield',  
'stacked': 1,  
'annoline1': 'Wakefield',  
'annoline2': None,  
'annoline3': None,  
'annoangle': 0.0,  
'borough': 'Bronx',  
'bbox': [-73.84720052054902,  
40.89470517661,  
-73.84720052054902,  
40.89470517661]}}
```

Transform the data into a pandas dataframe

In [185]:

```
# define the dataframe columns  
column_names = ['Borough', 'Neighborhood', 'Latitude', 'Longitude']  
  
# instantiate the dataframe  
neighborhoods = pd.DataFrame(columns=column_names)
```

In [186]:

```
neighborhoods
```

Out[186]:

Borough	Neighborhood	Latitude	Longitude
---------	--------------	----------	-----------

Then let's loop through the data and fill the dataframe one row at a time.

In [187]:

```
for data in neighborhoods_data:
    borough = neighborhood_name = data['properties']['borough']
    neighborhood_name = data['properties']['name']

    neighborhood_latlon = data['geometry']['coordinates']
    neighborhood_lat = neighborhood_latlon[1]
    neighborhood_lon = neighborhood_latlon[0]

    neighborhoods = neighborhoods.append({ 'Borough': borough,
                                            'Neighborhood': neighborhood_name,
                                            'Latitude': neighborhood_lat,
                                            'Longitude': neighborhood_lon}, ignore_index=True)
```

In [188]:

```
neighborhoods
```

Out[188]:

Borough	Neighborhood	Latitude	Longitude
0	Bronx	Wakefield	40.894705
1	Bronx	Co-op City	40.874294
2	Bronx	Eastchester	40.887556
3	Bronx	Fieldston	40.895437
4	Bronx	Riverdale	40.890834
5	Bronx	Kingsbridge	40.881687
6	Manhattan	Marble Hill	40.876551
7	Bronx	Woodlawn	40.898273
8	Bronx	Norwood	40.877224
9	Bronx	Williamsbridge	40.881039
10	Bronx	Baychester	40.866858
11	Bronx	Pelham Parkway	40.857413
12	Bronx	City Island	40.847247
13	Bronx	Bedford Park	40.870185
14	Bronx	University Heights	40.855727
15	Bronx	Morris Heights	40.847898
16	Bronx	Fordham	40.860997
17	Bronx	East Tremont	40.842696
18	Bronx	West Farms	40.839475
19	Bronx	High Bridge	40.836623
20	Bronx	Melrose	40.819754
21	Bronx	Mott Haven	40.806239
22	Bronx	Port Morris	40.801664
23	Bronx	Longwood	40.815099
24	Bronx	Hunts Point	40.809730
25	Bronx	Morrisania	40.823592
26	Bronx	Soundview	40.821012
27	Bronx	Clason Point	40.806551
28	Bronx	Throgs Neck	40.815109
29	Bronx	Country Club	40.844246
30	Bronx	Parkchester	40.837938
31	Bronx	Westchester Square	40.840619
32	Bronx	Van Nest	40.843608
33	Bronx	Morris Park	40.847549

	Borough	Neighborhood	Latitude	Longitude
34	Bronx	Belmont	40.857277	-73.888452
35	Bronx	Spuyten Duyvil	40.881395	-73.917190
36	Bronx	North Riverdale	40.908543	-73.904531
37	Bronx	Pelham Bay	40.850641	-73.832074
38	Bronx	Schuylerville	40.826580	-73.826203
39	Bronx	Edgewater Park	40.821986	-73.813885
40	Bronx	Castle Hill	40.819014	-73.848027
41	Bronx	Olinville	40.871371	-73.863324
42	Bronx	Pelham Gardens	40.862966	-73.841612
43	Bronx	Concourse	40.834284	-73.915589
44	Bronx	Unionport	40.829774	-73.850535
45	Bronx	Edenwald	40.884561	-73.848083
46	Brooklyn	Bay Ridge	40.625801	-74.030621
47	Brooklyn	Bensonhurst	40.611009	-73.995180
48	Brooklyn	Sunset Park	40.645103	-74.010316
49	Brooklyn	Greenpoint	40.730201	-73.954241
50	Brooklyn	Gravesend	40.595260	-73.973471
51	Brooklyn	Brighton Beach	40.576825	-73.965094
52	Brooklyn	Sheepshead Bay	40.586890	-73.943186
53	Brooklyn	Manhattan Terrace	40.614433	-73.957438
54	Brooklyn	Flatbush	40.636326	-73.958401
55	Brooklyn	Crown Heights	40.670829	-73.943291
56	Brooklyn	East Flatbush	40.641718	-73.936103
57	Brooklyn	Kensington	40.642382	-73.980421
58	Brooklyn	Windsor Terrace	40.656946	-73.980073
59	Brooklyn	Prospect Heights	40.676822	-73.964859
60	Brooklyn	Brownsville	40.663950	-73.910235
61	Brooklyn	Williamsburg	40.707144	-73.958115
62	Brooklyn	Bushwick	40.698116	-73.925258
63	Brooklyn	Bedford Stuyvesant	40.687232	-73.941785
64	Brooklyn	Brooklyn Heights	40.695864	-73.993782
65	Brooklyn	Cobble Hill	40.687920	-73.998561
66	Brooklyn	Carroll Gardens	40.680540	-73.994654
67	Brooklyn	Red Hook	40.676253	-74.012759
68	Brooklyn	Gowanus	40.673931	-73.994441
69	Brooklyn	Fort Greene	40.688527	-73.972906

	Borough	Neighborhood	Latitude	Longitude
70	Brooklyn	Park Slope	40.672321	-73.977050
71	Brooklyn	Cypress Hills	40.682391	-73.876616
72	Brooklyn	East New York	40.669926	-73.880699
73	Brooklyn	Starrett City	40.647589	-73.879370
74	Brooklyn	Canarsie	40.635564	-73.902093
75	Brooklyn	Flatlands	40.630446	-73.929113
76	Brooklyn	Mill Island	40.606336	-73.908186
77	Brooklyn	Manhattan Beach	40.577914	-73.943537
78	Brooklyn	Coney Island	40.574293	-73.988683
79	Brooklyn	Bath Beach	40.599519	-73.998752
80	Brooklyn	Borough Park	40.633131	-73.990498
81	Brooklyn	Dyker Heights	40.619219	-74.019314
82	Brooklyn	Gerritsen Beach	40.590848	-73.930102
83	Brooklyn	Marine Park	40.609748	-73.931344
84	Brooklyn	Clinton Hill	40.693229	-73.967843
85	Brooklyn	Sea Gate	40.576375	-74.007873
86	Brooklyn	Downtown	40.690844	-73.983463
87	Brooklyn	Boerum Hill	40.685683	-73.983748
88	Brooklyn	Prospect Lefferts Gardens	40.658420	-73.954899
89	Brooklyn	Ocean Hill	40.678403	-73.913068
90	Brooklyn	City Line	40.678570	-73.867976
91	Brooklyn	Bergen Beach	40.615150	-73.898556
92	Brooklyn	Midwood	40.625596	-73.957595
93	Brooklyn	Prospect Park South	40.647009	-73.962613
94	Brooklyn	Georgetown	40.623845	-73.916075
95	Brooklyn	East Williamsburg	40.708492	-73.938858
96	Brooklyn	North Side	40.714823	-73.958809
97	Brooklyn	South Side	40.710861	-73.958001
98	Brooklyn	Ocean Parkway	40.613060	-73.968367
99	Brooklyn	Fort Hamilton	40.614768	-74.031979
100	Manhattan	Chinatown	40.715618	-73.994279
101	Manhattan	Washington Heights	40.851903	-73.936900
102	Manhattan	Inwood	40.867684	-73.921210
103	Manhattan	Hamilton Heights	40.823604	-73.949688
104	Manhattan	Manhattanville	40.816934	-73.957385
105	Manhattan	Central Harlem	40.815976	-73.943211

Borough	Neighborhood	Latitude	Longitude
106	Manhattan	40.792249	-73.944182
107	Manhattan	40.775639	-73.960508
108	Manhattan	40.775930	-73.947118
109	Manhattan	40.768113	-73.958860
110	Manhattan	40.762160	-73.949168
111	Manhattan	40.787658	-73.977059
112	Manhattan	40.773529	-73.985338
113	Manhattan	40.759101	-73.996119
114	Manhattan	40.754691	-73.981669
115	Manhattan	40.748303	-73.978332
116	Manhattan	40.744035	-74.003116
117	Manhattan	40.726933	-73.999914
118	Manhattan	40.727847	-73.982226
119	Manhattan	40.717807	-73.980890
120	Manhattan	40.721522	-74.010683
121	Manhattan	40.719324	-73.997305
122	Manhattan	40.722184	-74.000657
123	Manhattan	40.734434	-74.006180
124	Manhattan	40.797307	-73.964286
125	Manhattan	40.808000	-73.963896
126	Manhattan	40.737210	-73.981376
127	Manhattan	40.711932	-74.016869
128	Manhattan	40.707107	-74.010665
129	Queens	40.768509	-73.915654
130	Queens	40.746349	-73.901842
131	Queens	40.751981	-73.882821
132	Queens	40.744049	-73.881656
133	Queens	40.654225	-73.838138
134	Queens	40.742382	-73.856825
135	Queens	40.725264	-73.844475
136	Queens	40.705179	-73.829819
137	Queens	40.697947	-73.831833
138	Queens	40.764454	-73.831773
139	Queens	40.750217	-73.939202
140	Queens	40.740176	-73.926916
141	Queens	40.764073	-73.867041

Borough	Neighborhood	Latitude	Longitude	
142	Queens	Maspeth	40.725427	-73.896217
143	Queens	Ridgewood	40.708323	-73.901435
144	Queens	Glendale	40.702762	-73.870742
145	Queens	Rego Park	40.728974	-73.857827
146	Queens	Woodhaven	40.689887	-73.858110
147	Queens	Ozone Park	40.680708	-73.843203
148	Queens	South Ozone Park	40.668550	-73.809865
149	Queens	College Point	40.784903	-73.843045
150	Queens	Whitestone	40.781291	-73.814202
151	Queens	Bayside	40.766041	-73.774274
152	Queens	Auburndale	40.761730	-73.791762
153	Queens	Little Neck	40.770826	-73.738898
154	Queens	Douglaston	40.766846	-73.742498
155	Queens	Glen Oaks	40.749441	-73.715481
156	Queens	Bellerose	40.728573	-73.720128
157	Queens	Kew Gardens Hills	40.722578	-73.820878
158	Queens	Fresh Meadows	40.734394	-73.782713
159	Queens	Briarwood	40.710935	-73.811748
160	Queens	Jamaica Center	40.704657	-73.796902
161	Queens	Oakland Gardens	40.745619	-73.754950
162	Queens	Queens Village	40.718893	-73.738715
163	Queens	Hollis	40.711243	-73.759250
164	Queens	South Jamaica	40.696911	-73.790426
165	Queens	St. Albans	40.694445	-73.758676
166	Queens	Rochdale	40.675211	-73.772588
167	Queens	Springfield Gardens	40.666230	-73.760421
168	Queens	Cambria Heights	40.692775	-73.735269
169	Queens	Rosedale	40.659816	-73.735261
170	Queens	Far Rockaway	40.603134	-73.754980
171	Queens	Broad Channel	40.603027	-73.820055
172	Queens	Breezy Point	40.557401	-73.925512
173	Queens	Steinway	40.775923	-73.902290
174	Queens	Beechhurst	40.792781	-73.804365
175	Queens	Bay Terrace	40.782843	-73.776802
176	Queens	Edgemere	40.595642	-73.776133
177	Queens	Arverne	40.589144	-73.791992

Borough	Neighborhood	Latitude	Longitude
178	Queens	Rockaway Beach	40.582802 -73.822361
179	Queens	Neponsit	40.572037 -73.857547
180	Queens	Murray Hill	40.764126 -73.812763
181	Queens	Floral Park	40.741378 -73.708847
182	Queens	Holliswood	40.720957 -73.767142
183	Queens	Jamaica Estates	40.716805 -73.787227
184	Queens	Queensboro Hill	40.744572 -73.825809
185	Queens	Hillcrest	40.723825 -73.797603
186	Queens	Ravenswood	40.761705 -73.931575
187	Queens	Lindenwood	40.663918 -73.849638
188	Queens	Laurelton	40.667884 -73.740256
189	Queens	Lefrak City	40.736075 -73.862525
190	Queens	Belle Harbor	40.576156 -73.854018
191	Queens	Rockaway Park	40.580343 -73.841534
192	Queens	Somerville	40.597711 -73.796648
193	Queens	Brookville	40.660003 -73.751753
194	Queens	Bellaire	40.733014 -73.738892
195	Queens	North Corona	40.754071 -73.857518
196	Queens	Forest Hills Gardens	40.714611 -73.841022
197	Staten Island	St. George	40.644982 -74.079353
198	Staten Island	New Brighton	40.640615 -74.087017
199	Staten Island	Stapleton	40.626928 -74.077902
200	Staten Island	Rosebank	40.615305 -74.069805
201	Staten Island	West Brighton	40.631879 -74.107182
202	Staten Island	Grymes Hill	40.624185 -74.087248
203	Staten Island	Todt Hill	40.597069 -74.111329
204	Staten Island	South Beach	40.580247 -74.079553
205	Staten Island	Port Richmond	40.633669 -74.129434
206	Staten Island	Mariner's Harbor	40.632546 -74.150085
207	Staten Island	Port Ivory	40.639683 -74.174645
208	Staten Island	Castleton Corners	40.613336 -74.119181
209	Staten Island	New Springville	40.594252 -74.164960
210	Staten Island	Travis	40.586314 -74.190737
211	Staten Island	New Dorp	40.572572 -74.116479
212	Staten Island	Oakwood	40.558462 -74.121566
213	Staten Island	Great Kills	40.549480 -74.149324

Borough	Neighborhood	Latitude	Longitude	
214	Staten Island	Eltingville	40.542231	-74.164331
215	Staten Island	Annadale	40.538114	-74.178549
216	Staten Island	Woodrow	40.541968	-74.205246
217	Staten Island	Tottenville	40.505334	-74.246569
218	Staten Island	Tompkinsville	40.637316	-74.080554
219	Staten Island	Silver Lake	40.619193	-74.096290
220	Staten Island	Sunnyside	40.612760	-74.097126
221	Brooklyn	Ditmas Park	40.643675	-73.961013
222	Brooklyn	Wingate	40.660947	-73.937187
223	Brooklyn	Rugby	40.655572	-73.926882
224	Staten Island	Park Hill	40.609190	-74.080157
225	Staten Island	Westerleigh	40.621090	-74.133041
226	Staten Island	Graniteville	40.620172	-74.153152
227	Staten Island	Arlington	40.635325	-74.165104
228	Staten Island	Arrochar	40.596313	-74.067124
229	Staten Island	Grasmere	40.598268	-74.076674
230	Staten Island	Old Town	40.596329	-74.087511
231	Staten Island	Dongan Hills	40.588673	-74.096399
232	Staten Island	Midland Beach	40.573527	-74.093483
233	Staten Island	Grant City	40.576216	-74.105856
234	Staten Island	New Dorp Beach	40.564255	-74.104327
235	Staten Island	Bay Terrace	40.553988	-74.139166
236	Staten Island	Huguenot	40.531912	-74.191741
237	Staten Island	Pleasant Plains	40.524699	-74.219831
238	Staten Island	Butler Manor	40.506082	-74.229504
239	Staten Island	Charleston	40.530531	-74.232158
240	Staten Island	Rossville	40.549404	-74.215729
241	Staten Island	Arden Heights	40.549286	-74.185887
242	Staten Island	Greenridge	40.555295	-74.170794
243	Staten Island	Heartland Village	40.589139	-74.159022
244	Staten Island	Chelsea	40.594726	-74.189560
245	Staten Island	Bloomfield	40.605779	-74.187256
246	Staten Island	Bulls Head	40.609592	-74.159409
247	Manhattan	Carnegie Hill	40.782683	-73.953256
248	Manhattan	Noho	40.723259	-73.988434
249	Manhattan	Civic Center	40.715229	-74.005415

Borough	Neighborhood	Latitude	Longitude
250	Manhattan	Midtown South	40.748510 -73.988713
251	Staten Island	Richmond Town	40.569606 -74.134057
252	Staten Island	Shore Acres	40.609719 -74.066678
253	Staten Island	Clifton	40.619178 -74.072642
254	Staten Island	Concord	40.604473 -74.084024
255	Staten Island	Emerson Hill	40.606794 -74.097762
256	Staten Island	Randall Manor	40.635630 -74.098051
257	Staten Island	Howland Hook	40.638433 -74.186223
258	Staten Island	Elm Park	40.630147 -74.141817
259	Brooklyn	Remsen Village	40.652117 -73.916653
260	Brooklyn	New Lots	40.662744 -73.885118
261	Brooklyn	Paerdegat Basin	40.631318 -73.902335
262	Brooklyn	Mill Basin	40.615974 -73.915154
263	Queens	Jamaica Hills	40.711460 -73.796465
264	Queens	Utopia	40.733500 -73.796717
265	Queens	Pomonok	40.734936 -73.804861
266	Queens	Astoria Heights	40.770317 -73.894680
267	Bronx	Clarendon Village	40.831428 -73.901199
268	Bronx	Concourse Village	40.824780 -73.915847
269	Bronx	Mount Eden	40.843826 -73.916556
270	Bronx	Mount Hope	40.848842 -73.908299
271	Manhattan	Sutton Place	40.760280 -73.963556
272	Queens	Hunters Point	40.743414 -73.953868
273	Manhattan	Turtle Bay	40.752042 -73.967708
274	Manhattan	Tudor City	40.746917 -73.971219
275	Manhattan	Stuyvesant Town	40.731000 -73.974052
276	Manhattan	Flatiron	40.739673 -73.990947
277	Queens	Sunnyside Gardens	40.745652 -73.918193
278	Queens	Blissville	40.737251 -73.932442
279	Brooklyn	Fulton Ferry	40.703281 -73.995508
280	Brooklyn	Vinegar Hill	40.703321 -73.981116
281	Brooklyn	Weeksville	40.675040 -73.930531
282	Brooklyn	Broadway Junction	40.677861 -73.903317
283	Brooklyn	Dumbo	40.703176 -73.988753
284	Staten Island	Manor Heights	40.601810 -74.120594
285	Staten Island	Willowbrook	40.603707 -74.132084

	Borough	Neighborhood	Latitude	Longitude
286	Staten Island	Sandy Ground	40.541140	-74.217766
287	Staten Island	Egbertville	40.579119	-74.127272
288	Queens	Roxbury	40.567376	-73.892138
289	Brooklyn	Homecrest	40.598525	-73.959185
290	Queens	Middle Village	40.716415	-73.881143
291	Staten Island	Prince's Bay	40.526264	-74.201526
292	Staten Island	Lighthouse Hill	40.576506	-74.137927
293	Staten Island	Richmond Valley	40.519541	-74.229571
294	Queens	Malba	40.790602	-73.826678
295	Brooklyn	Highland Park	40.681999	-73.890346
296	Brooklyn	Madison	40.609378	-73.948415
297	Bronx	Bronxdale	40.852723	-73.861726
298	Bronx	Allerton	40.865788	-73.859319
299	Bronx	Kingsbridge Heights	40.870392	-73.901523
300	Brooklyn	Erasmus	40.646926	-73.948177
301	Manhattan	Hudson Yards	40.756658	-74.000111
302	Queens	Hammels	40.587338	-73.805530
303	Queens	Bayswater	40.611322	-73.765968
304	Queens	Queensbridge	40.756091	-73.945631
305	Staten Island	Fox Hills	40.617311	-74.081740

In [189]:

```
neighborhoods.head()
```

Out[189]:

	Borough	Neighborhood	Latitude	Longitude
0	Bronx	Wakefield	40.894705	-73.847201
1	Bronx	Co-op City	40.874294	-73.829939
2	Bronx	Eastchester	40.887556	-73.827806
3	Bronx	Fieldston	40.895437	-73.905643
4	Bronx	Riverdale	40.890834	-73.912585

In [190]:

```
print('The dataframe has {} boroughs and {} neighborhoods.'.format(
    len(neighborhoods['Borough'].unique()),
    neighborhoods.shape[0]
)
)
```

The dataframe has 5 boroughs and 306 neighborhoods.

Finding out Boroughs and their neighborhoods

In [191]:

```
manhattan_data = neighborhoods[neighborhoods['Borough'] == 'Manhattan'].reset_index(drop=True)
manhattan_data
```

Out[191]:

	Borough	Neighborhood	Latitude	Longitude
0	Manhattan	Marble Hill	40.876551	-73.910660
1	Manhattan	Chinatown	40.715618	-73.994279
2	Manhattan	Washington Heights	40.851903	-73.936900
3	Manhattan	Inwood	40.867684	-73.921210
4	Manhattan	Hamilton Heights	40.823604	-73.949688
5	Manhattan	Manhattanville	40.816934	-73.957385
6	Manhattan	Central Harlem	40.815976	-73.943211
7	Manhattan	East Harlem	40.792249	-73.944182
8	Manhattan	Upper East Side	40.775639	-73.960508
9	Manhattan	Yorkville	40.775930	-73.947118
10	Manhattan	Lenox Hill	40.768113	-73.958860
11	Manhattan	Roosevelt Island	40.762160	-73.949168
12	Manhattan	Upper West Side	40.787658	-73.977059
13	Manhattan	Lincoln Square	40.773529	-73.985338
14	Manhattan	Clinton	40.759101	-73.996119
15	Manhattan	Midtown	40.754691	-73.981669
16	Manhattan	Murray Hill	40.748303	-73.978332
17	Manhattan	Chelsea	40.744035	-74.003116
18	Manhattan	Greenwich Village	40.726933	-73.999914
19	Manhattan	East Village	40.727847	-73.982226
20	Manhattan	Lower East Side	40.717807	-73.980890
21	Manhattan	Tribeca	40.721522	-74.010683
22	Manhattan	Little Italy	40.719324	-73.997305
23	Manhattan	Soho	40.722184	-74.000657
24	Manhattan	West Village	40.734434	-74.006180
25	Manhattan	Manhattan Valley	40.797307	-73.964286
26	Manhattan	Morningside Heights	40.808000	-73.963896
27	Manhattan	Gramercy	40.737210	-73.981376
28	Manhattan	Battery Park City	40.711932	-74.016869
29	Manhattan	Financial District	40.707107	-74.010665
30	Manhattan	Carnegie Hill	40.782683	-73.953256
31	Manhattan	Noho	40.723259	-73.988434
32	Manhattan	Civic Center	40.715229	-74.005415
33	Manhattan	Midtown South	40.748510	-73.988713

Borough	Neighborhood	Latitude	Longitude	
34	Manhattan	Sutton Place	40.760280	-73.963556
35	Manhattan	Turtle Bay	40.752042	-73.967708
36	Manhattan	Tudor City	40.746917	-73.971219
37	Manhattan	Stuyvesant Town	40.731000	-73.974052
38	Manhattan	Flatiron	40.739673	-73.990947
39	Manhattan	Hudson Yards	40.756658	-74.000111

In [192]:

```
brooklyn_data = neighborhoods[neighborhoods['Borough'] == 'Brooklyn'].reset_index(drop=True)
brooklyn_data
```

Out[192]:

Borough	Neighborhood	Latitude	Longitude
0	Brooklyn	Bay Ridge	40.625801
1	Brooklyn	Bensonhurst	40.611009
2	Brooklyn	Sunset Park	40.645103
3	Brooklyn	Greenpoint	40.730201
4	Brooklyn	Gravesend	40.595260
5	Brooklyn	Brighton Beach	40.576825
6	Brooklyn	Sheepshead Bay	40.586890
7	Brooklyn	Manhattan Terrace	40.614433
8	Brooklyn	Flatbush	40.636326
9	Brooklyn	Crown Heights	40.670829
10	Brooklyn	East Flatbush	40.641718
11	Brooklyn	Kensington	40.642382
12	Brooklyn	Windsor Terrace	40.656946
13	Brooklyn	Prospect Heights	40.676822
14	Brooklyn	Brownsville	40.663950
15	Brooklyn	Williamsburg	40.707144
16	Brooklyn	Bushwick	40.698116
17	Brooklyn	Bedford Stuyvesant	40.687232
18	Brooklyn	Brooklyn Heights	40.695864
19	Brooklyn	Cobble Hill	40.687920
20	Brooklyn	Carroll Gardens	40.680540
21	Brooklyn	Red Hook	40.676253
22	Brooklyn	Gowanus	40.673931
23	Brooklyn	Fort Greene	40.688527
24	Brooklyn	Park Slope	40.672321
25	Brooklyn	Cypress Hills	40.682391
26	Brooklyn	East New York	40.669926
27	Brooklyn	Starrett City	40.647589
28	Brooklyn	Canarsie	40.635564
29	Brooklyn	Flatlands	40.630446
30	Brooklyn	Mill Island	40.606336
31	Brooklyn	Manhattan Beach	40.577914
32	Brooklyn	Coney Island	40.574293
33	Brooklyn	Bath Beach	40.599519

Borough	Neighborhood	Latitude	Longitude	
34	Brooklyn	Borough Park	40.633131	-73.990498
35	Brooklyn	Dyker Heights	40.619219	-74.019314
36	Brooklyn	Gerritsen Beach	40.590848	-73.930102
37	Brooklyn	Marine Park	40.609748	-73.931344
38	Brooklyn	Clinton Hill	40.693229	-73.967843
39	Brooklyn	Sea Gate	40.576375	-74.007873
40	Brooklyn	Downtown	40.690844	-73.983463
41	Brooklyn	Boerum Hill	40.685683	-73.983748
42	Brooklyn	Prospect Lefferts Gardens	40.658420	-73.954899
43	Brooklyn	Ocean Hill	40.678403	-73.913068
44	Brooklyn	City Line	40.678570	-73.867976
45	Brooklyn	Bergen Beach	40.615150	-73.898556
46	Brooklyn	Midwood	40.625596	-73.957595
47	Brooklyn	Prospect Park South	40.647009	-73.962613
48	Brooklyn	Georgetown	40.623845	-73.916075
49	Brooklyn	East Williamsburg	40.708492	-73.938858
50	Brooklyn	North Side	40.714823	-73.958809
51	Brooklyn	South Side	40.710861	-73.958001
52	Brooklyn	Ocean Parkway	40.613060	-73.968367
53	Brooklyn	Fort Hamilton	40.614768	-74.031979
54	Brooklyn	Ditmas Park	40.643675	-73.961013
55	Brooklyn	Wingate	40.660947	-73.937187
56	Brooklyn	Rugby	40.655572	-73.926882
57	Brooklyn	Remsen Village	40.652117	-73.916653
58	Brooklyn	New Lots	40.662744	-73.885118
59	Brooklyn	Padergat Basin	40.631318	-73.902335
60	Brooklyn	Mill Basin	40.615974	-73.915154
61	Brooklyn	Fulton Ferry	40.703281	-73.995508
62	Brooklyn	Vinegar Hill	40.703321	-73.981116
63	Brooklyn	Weeksville	40.675040	-73.930531
64	Brooklyn	Broadway Junction	40.677861	-73.903317
65	Brooklyn	Dumbo	40.703176	-73.988753
66	Brooklyn	Homecrest	40.598525	-73.959185
67	Brooklyn	Highland Park	40.681999	-73.890346
68	Brooklyn	Madison	40.609378	-73.948415
69	Brooklyn	Erasmus	40.646926	-73.948177

In [193]:

```
bronx_data = neighborhoods[neighborhoods['Borough'] == 'Bronx'].reset_index(drop=True)  
bronx_data
```

Out[193]:

	Borough	Neighborhood	Latitude	Longitude
0	Bronx	Wakefield	40.894705	-73.847201
1	Bronx	Co-op City	40.874294	-73.829939
2	Bronx	Eastchester	40.887556	-73.827806
3	Bronx	Fieldston	40.895437	-73.905643
4	Bronx	Riverdale	40.890834	-73.912585
5	Bronx	Kingsbridge	40.881687	-73.902818
6	Bronx	Woodlawn	40.898273	-73.867315
7	Bronx	Norwood	40.877224	-73.879391
8	Bronx	Williamsbridge	40.881039	-73.857446
9	Bronx	Baychester	40.866858	-73.835798
10	Bronx	Pelham Parkway	40.857413	-73.854756
11	Bronx	City Island	40.847247	-73.786488
12	Bronx	Bedford Park	40.870185	-73.885512
13	Bronx	University Heights	40.855727	-73.910416
14	Bronx	Morris Heights	40.847898	-73.919672
15	Bronx	Fordham	40.860997	-73.896427
16	Bronx	East Tremont	40.842696	-73.887356
17	Bronx	West Farms	40.839475	-73.877745
18	Bronx	High Bridge	40.836623	-73.926102
19	Bronx	Melrose	40.819754	-73.909422
20	Bronx	Mott Haven	40.806239	-73.916100
21	Bronx	Port Morris	40.801664	-73.913221
22	Bronx	Longwood	40.815099	-73.895788
23	Bronx	Hunts Point	40.809730	-73.883315
24	Bronx	Morrisania	40.823592	-73.901506
25	Bronx	Soundview	40.821012	-73.865746
26	Bronx	Clason Point	40.806551	-73.854144
27	Bronx	Throgs Neck	40.815109	-73.816350
28	Bronx	Country Club	40.844246	-73.824099
29	Bronx	Parkchester	40.837938	-73.856003
30	Bronx	Westchester Square	40.840619	-73.842194
31	Bronx	Van Nest	40.843608	-73.866299
32	Bronx	Morris Park	40.847549	-73.850402
33	Bronx	Belmont	40.857277	-73.888452

Borough	Neighborhood	Latitude	Longitude
34	Bronx	Spuyten Duyvil	40.881395 -73.917190
35	Bronx	North Riverdale	40.908543 -73.904531
36	Bronx	Pelham Bay	40.850641 -73.832074
37	Bronx	Schuylerville	40.826580 -73.826203
38	Bronx	Edgewater Park	40.821986 -73.813885
39	Bronx	Castle Hill	40.819014 -73.848027
40	Bronx	Olinville	40.871371 -73.863324
41	Bronx	Pelham Gardens	40.862966 -73.841612
42	Bronx	Concourse	40.834284 -73.915589
43	Bronx	Unionport	40.829774 -73.850535
44	Bronx	Edenwald	40.884561 -73.848083
45	Bronx	Claremont Village	40.831428 -73.901199
46	Bronx	Concourse Village	40.824780 -73.915847
47	Bronx	Mount Eden	40.843826 -73.916556
48	Bronx	Mount Hope	40.848842 -73.908299
49	Bronx	Bronxdale	40.852723 -73.861726
50	Bronx	Allerton	40.865788 -73.859319
51	Bronx	Kingsbridge Heights	40.870392 -73.901523

In [194]:

```
queens_data = neighborhoods[neighborhoods['Borough'] == 'Queens'].reset_index(drop=True)
queens_data
```

Out[194]:

Borough	Neighborhood	Latitude	Longitude
0	Queens	Astoria	40.768509
1	Queens	Woodside	40.746349
2	Queens	Jackson Heights	40.751981
3	Queens	Elmhurst	40.744049
4	Queens	Howard Beach	40.654225
5	Queens	Corona	40.742382
6	Queens	Forest Hills	40.725264
7	Queens	Kew Gardens	40.705179
8	Queens	Richmond Hill	40.697947
9	Queens	Flushing	40.764454
10	Queens	Long Island City	40.750217
11	Queens	Sunnyside	40.740176
12	Queens	East Elmhurst	40.764073
13	Queens	Maspeth	40.725427
14	Queens	Ridgewood	40.708323
15	Queens	Glendale	40.702762
16	Queens	Rego Park	40.728974
17	Queens	Woodhaven	40.689887
18	Queens	Ozone Park	40.680708
19	Queens	South Ozone Park	40.668550
20	Queens	College Point	40.784903
21	Queens	Whitestone	40.781291
22	Queens	Bayside	40.766041
23	Queens	Auburndale	40.761730
24	Queens	Little Neck	40.770826
25	Queens	Douglaston	40.766846
26	Queens	Glen Oaks	40.749441
27	Queens	Bellerose	40.728573
28	Queens	Kew Gardens Hills	40.722578
29	Queens	Fresh Meadows	40.734394
30	Queens	Briarwood	40.710935
31	Queens	Jamaica Center	40.704657
32	Queens	Oakland Gardens	40.745619
33	Queens	Queens Village	40.718893

Borough	Neighborhood	Latitude	Longitude
34	Queens	Hollis	40.711243 -73.759250
35	Queens	South Jamaica	40.696911 -73.790426
36	Queens	St. Albans	40.694445 -73.758676
37	Queens	Rochdale	40.675211 -73.772588
38	Queens	Springfield Gardens	40.666230 -73.760421
39	Queens	Cambria Heights	40.692775 -73.735269
40	Queens	Rosedale	40.659816 -73.735261
41	Queens	Far Rockaway	40.603134 -73.754980
42	Queens	Broad Channel	40.603027 -73.820055
43	Queens	Breezy Point	40.557401 -73.925512
44	Queens	Steinway	40.775923 -73.902290
45	Queens	Beechhurst	40.792781 -73.804365
46	Queens	Bay Terrace	40.782843 -73.776802
47	Queens	Edgemere	40.595642 -73.776133
48	Queens	Arverne	40.589144 -73.791992
49	Queens	Rockaway Beach	40.582802 -73.822361
50	Queens	Neponsit	40.572037 -73.857547
51	Queens	Murray Hill	40.764126 -73.812763
52	Queens	Floral Park	40.741378 -73.708847
53	Queens	Holliswood	40.720957 -73.767142
54	Queens	Jamaica Estates	40.716805 -73.787227
55	Queens	Queensboro Hill	40.744572 -73.825809
56	Queens	Hillcrest	40.723825 -73.797603
57	Queens	Ravenswood	40.761705 -73.931575
58	Queens	Lindenwood	40.663918 -73.849638
59	Queens	Laurelton	40.667884 -73.740256
60	Queens	Lefrak City	40.736075 -73.862525
61	Queens	Belle Harbor	40.576156 -73.854018
62	Queens	Rockaway Park	40.580343 -73.841534
63	Queens	Somerville	40.597711 -73.796648
64	Queens	Brookville	40.660003 -73.751753
65	Queens	Bellaire	40.733014 -73.738892
66	Queens	North Corona	40.754071 -73.857518
67	Queens	Forest Hills Gardens	40.714611 -73.841022
68	Queens	Jamaica Hills	40.711460 -73.796465
69	Queens	Utopia	40.733500 -73.796717

Borough	Neighborhood	Latitude	Longitude	
70	Queens	Pomonok	40.734936	-73.804861
71	Queens	Astoria Heights	40.770317	-73.894680
72	Queens	Hunters Point	40.743414	-73.953868
73	Queens	Sunnyside Gardens	40.745652	-73.918193
74	Queens	Blissville	40.737251	-73.932442
75	Queens	Roxbury	40.567376	-73.892138
76	Queens	Middle Village	40.716415	-73.881143
77	Queens	Malba	40.790602	-73.826678
78	Queens	Hammels	40.587338	-73.805530
79	Queens	Bayswater	40.611322	-73.765968
80	Queens	Queensbridge	40.756091	-73.945631

In [195]:

```
statenisland_data = neighborhoods[neighborhoods['Borough'] == 'Staten Island'].reset_index  
(drop=True)  
statenisland_data
```

Out[195]:

	Borough	Neighborhood	Latitude	Longitude
0	Staten Island	St. George	40.644982	-74.079353
1	Staten Island	New Brighton	40.640615	-74.087017
2	Staten Island	Stapleton	40.626928	-74.077902
3	Staten Island	Rosebank	40.615305	-74.069805
4	Staten Island	West Brighton	40.631879	-74.107182
5	Staten Island	Grymes Hill	40.624185	-74.087248
6	Staten Island	Todt Hill	40.597069	-74.111329
7	Staten Island	South Beach	40.580247	-74.079553
8	Staten Island	Port Richmond	40.633669	-74.129434
9	Staten Island	Mariner's Harbor	40.632546	-74.150085
10	Staten Island	Port Ivory	40.639683	-74.174645
11	Staten Island	Castleton Corners	40.613336	-74.119181
12	Staten Island	New Springville	40.594252	-74.164960
13	Staten Island	Travis	40.586314	-74.190737
14	Staten Island	New Dorp	40.572572	-74.116479
15	Staten Island	Oakwood	40.558462	-74.121566
16	Staten Island	Great Kills	40.549480	-74.149324
17	Staten Island	Eltingville	40.542231	-74.164331
18	Staten Island	Annadale	40.538114	-74.178549
19	Staten Island	Woodrow	40.541968	-74.205246
20	Staten Island	Tottenville	40.505334	-74.246569
21	Staten Island	Tompkinsville	40.637316	-74.080554
22	Staten Island	Silver Lake	40.619193	-74.096290
23	Staten Island	Sunnyside	40.612760	-74.097126
24	Staten Island	Park Hill	40.609190	-74.080157
25	Staten Island	Westerleigh	40.621090	-74.133041
26	Staten Island	Graniteville	40.620172	-74.153152
27	Staten Island	Arlington	40.635325	-74.165104
28	Staten Island	Arrochar	40.596313	-74.067124
29	Staten Island	Grasmere	40.598268	-74.076674
30	Staten Island	Old Town	40.596329	-74.087511
31	Staten Island	Dongan Hills	40.588673	-74.096399
32	Staten Island	Midland Beach	40.573527	-74.093483
33	Staten Island	Grant City	40.576216	-74.105856

	Borough	Neighborhood	Latitude	Longitude
34	Staten Island	New Dorp Beach	40.564255	-74.104327
35	Staten Island	Bay Terrace	40.553988	-74.139166
36	Staten Island	Huguenot	40.531912	-74.191741
37	Staten Island	Pleasant Plains	40.524699	-74.219831
38	Staten Island	Butler Manor	40.506082	-74.229504
39	Staten Island	Charleston	40.530531	-74.232158
40	Staten Island	Rossville	40.549404	-74.215729
41	Staten Island	Arden Heights	40.549286	-74.185887
42	Staten Island	Greenridge	40.555295	-74.170794
43	Staten Island	Heartland Village	40.589139	-74.159022
44	Staten Island	Chelsea	40.594726	-74.189560
45	Staten Island	Bloomfield	40.605779	-74.187256
46	Staten Island	Bulls Head	40.609592	-74.159409
47	Staten Island	Richmond Town	40.569606	-74.134057
48	Staten Island	Shore Acres	40.609719	-74.066678
49	Staten Island	Clifton	40.619178	-74.072642
50	Staten Island	Concord	40.604473	-74.084024
51	Staten Island	Emerson Hill	40.606794	-74.097762
52	Staten Island	Randall Manor	40.635630	-74.098051
53	Staten Island	Howland Hook	40.638433	-74.186223
54	Staten Island	Elm Park	40.630147	-74.141817
55	Staten Island	Manor Heights	40.601810	-74.120594
56	Staten Island	Willowbrook	40.603707	-74.132084
57	Staten Island	Sandy Ground	40.541140	-74.217766
58	Staten Island	Egbertville	40.579119	-74.127272
59	Staten Island	Prince's Bay	40.526264	-74.201526
60	Staten Island	Lighthouse Hill	40.576506	-74.137927
61	Staten Island	Richmond Valley	40.519541	-74.229571
62	Staten Island	Fox Hills	40.617311	-74.081740

In [196]:

```
print('Manhattan dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(manhattan_data['Borough'].unique()),
    manhattan_data.shape[0]
)
)
```

Manhattan dataframe has 1 boroughs and 40 neighbourhoods.

In [197]:

```
print('Brooklyn dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(brooklyn_data['Borough'].unique()),
    brooklyn_data.shape[0]
)
)
```

Brooklyn dataframe has 1 boroughs and 70 neighbourhoods.

In [198]:

```
print('Bronx dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(bronx_data['Borough'].unique()),
    bronx_data.shape[0]
)
)
```

Bronx dataframe has 1 boroughs and 52 neighbourhoods.

In [199]:

```
print('Queens dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(queens_data['Borough'].unique()),
    queens_data.shape[0]
)
)
```

Queens dataframe has 1 boroughs and 81 neighbourhoods.

In [200]:

```
print('Staten Island dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(statenisland_data['Borough'].unique()),
    statenisland_data.shape[0]
)
)
```

Staten Island dataframe has 1 boroughs and 63 neighbourhoods.

We will explore Manhattan and its neighbourhood

Create a map of New York with neighborhoods superimposed on top.

In [201]:

```
address = 'New York City, NY'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geographical coordinate of New York City are {}, {}'.format(latitude, longitude))
```

The geographical coordinate of New York City are 40.7127281, -74.0060152.

In [202]:

```
# create map of New York using latitude and longitude values
map_newyork = folium.Map(location=[latitude, longitude], zoom_start=10)

# add markers to map
for lat, lng, borough, neighborhood in zip(neighborhoods['Latitude'], neighborhoods['Longitude'], neighborhoods['Borough'], neighborhoods['Neighborhood']):
    label = '{}, {}'.format(neighborhood, borough)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_newyork)

map_newyork
```

Out[202]:

Make this Notebook Trusted to load map: File -> Trust Notebook

In [203]:

```
manhattan_data = neighborhoods[neighborhoods['Borough'] == 'Manhattan'].reset_index(drop=True)
manhattan_data.head()
```

Out[203]:

	Borough	Neighborhood	Latitude	Longitude
0	Manhattan	Marble Hill	40.876551	-73.910660
1	Manhattan	Chinatown	40.715618	-73.994279
2	Manhattan	Washington Heights	40.851903	-73.936900
3	Manhattan	Inwood	40.867684	-73.921210
4	Manhattan	Hamilton Heights	40.823604	-73.949688

In [204]:

```
print('Manhattan dataframe has {} boroughs and {} neighbourhoods.'.format(
    len(manhattan_data['Borough'].unique()),
    manhattan_data.shape[0]
)
)
```

Manhattan dataframe has 1 boroughs and 40 neighbourhoods.

In [205]:

```
address = 'Manhattan, NY'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geographical coordinate of Manhattan are {}, {}.'.format(latitude, longitude))
```

The geographical coordinate of Manhattan are 40.7896239, -73.9598939.

In [206]:

```
# create map of Manhattan using Latitude and Longitude values
map_manhattan = folium.Map(location=[latitude, longitude], zoom_start=11)

# add markers to map
for lat, lng, label in zip(manhattan_data['Latitude'], manhattan_data['Longitude'], manhattan_data['Neighborhood']):
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_manhattan)

map_manhattan
```

Out[206]:

Make this Notebook Trusted to load map: File -> Trust Notebook

Define Foursquare Credentials and Version

In [207]:

```
CLIENT_ID = 'KW0JV4RJ1HGQL2P4RVDTXMOC4HBRFMQA3GLD0CWFMB4GTEMC' # your Foursquare ID
CLIENT_SECRET = 'QMYKCS1PKC4WAINWPUAUQ1ALE25NPDKSQVV0Y025PEETYX2' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version
LIMIT = 100 # A default Foursquare API limit value

print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

Your credentails:

```
CLIENT_ID: KW0JV4RJ1HGQL2P4RVDTXMOC4HBRFMQA3GLD0CWFMB4GTEMC
CLIENT_SECRET:QMYKCS1PKC4WAINWPUAUQ1ALE25NPDKSQVV0Y025PEETYX2
```

Let's explore the first neighborhood in our dataframe.

In [208]:

```
manhattan_data.loc[0, 'Neighborhood']
```

Out[208]:

```
'Marble Hill'
```

In [209]:

```
neighborhood_latitude = manhattan_data.loc[0, 'Latitude'] # neighborhood Latitude value
neighborhood_longitude = manhattan_data.loc[0, 'Longitude'] # neighborhood Longitude value

neighborhood_name = manhattan_data.loc[0, 'Neighborhood'] # neighborhood name

print('Latitude and longitude values of {} are {}, {}.'.format(neighborhood_name,
                                                               neighborhood_latitude,
                                                               neighborhood_longitude))
```

Latitude and longitude values of Marble Hill are 40.87655077879964, -73.91065965862981.

Now, let's get the top 100 venues that are in Marble Hill within a radius of 500 meters.

In [210]:

```
# type your answer here

LIMIT = 100
radius = 500
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={} ,{}&radius={} &limit={}'.format(CLIENT_ID,
CLIENT_SECRET,
VERSION,
neighborhood_latitude,
neighborhood_longitude,
radius,
LIMIT)
url
```

Out[210]:

```
'https://api.foursquare.com/v2/venues/explore?&client_id=KW0JV4RJ1HGQL2P4RVDT
XMOC4HBRFMQA3GLD0CWFMB4GTEMC&client_secret=QMYKCS1PKC4WAINWPUAUQ1ALE25NPKDKSQ
VV0Y025PEETYX2&v=20180605&ll=40.87655077879964,-73.91065965862981&radius=500&
limit=100'
```

In [211]:

```
results = requests.get(url).json()  
results
```

Out[211]:

```
{'meta': {'code': 200, 'requestId': '601e62572aaae823b6bd50ee'},  
 'response': {'suggestedFilters': {'header': 'Tap to show:',  
   'filters': [{'name': 'Open now', 'key': 'openNow'}]}},  
 'headerLocation': 'Marble Hill',  
 'headerFullLocation': 'Marble Hill, New York',  
 'headerLocationGranularity': 'neighborhood',  
 'totalResults': 23,  
 'suggestedBounds': {'ne': {'lat': 40.88105078329964,  
   'lng': -73.90471933917806},  
   'sw': {'lat': 40.87205077429964, 'lng': -73.91659997808156}},  
 'groups': [{}],  
   'type': 'Recommended Places',  
   'name': 'recommended',  
   'items': [{}],  
     'reasons': {'count': 0,  
       'items': [{}],  
         'summary': 'This spot is popular',  
         'type': 'general',  
         'reasonName': 'globalInteractionReason'}]},  
 'venue': {'id': '4b4429abf964a52037f225e3',  
   'name': "Arturo's",  
   'location': {'address': '5198 Broadway',  
     'crossStreet': 'at 225th St.',  
     'lat': 40.87441177110231,  
     'lng': -73.91027100981574,  
     'labeledLatLngs': [{}],  
       'label': 'display',  
       'lat': 40.87441177110231,  
       'lng': -73.91027100981574},  
       'label': 'entrance', 'lat': 40.874401, 'lng': -73.910339}],  
   'distance': 240,  
   'postalCode': '10463',  
   'cc': 'US',  
   'city': 'New York',  
   'state': 'NY',  
   'country': 'United States',  
   'formattedAddress': ['5198 Broadway (at 225th St.)',  
     'New York, NY 10463',  
     'United States']},  
   'categories': [{}],  
     'id': '4bf58dd8d48988d1ca941735',  
     'name': 'Pizza Place',  
     'pluralName': 'Pizza Places',  
     'shortName': 'Pizza',  
     'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/pizz  
a_',  
       'suffix': '.png'},  
       'primary': True}],  
     'delivery': {'id': '72548',  
       'url': 'https://www.seamless.com/menu/arturos-pizza-5189-broadway-ave  
-new-york/72548?affiliate=1131&utm_source=foursquare-affiliate-network&utm_me  
dium=affiliate&utm_campaign=1131&utm_content=72548',  
       'provider': {'name': 'seamless',  
         'icon': {'prefix': 'https://fastly.4sqi.net/img/general/cap/'},  
         'sizes': [40, 50],  
         'name': '/delivery_provider_seamless_20180129.png'}},  
       'photos': {'count': 0, 'groups': []},  
       'referralId': 'e-0-4b4429abf964a52037f225e3-0'},  
     {'reasons': {'count': 0,  
       'items': [{}],  
         'summary': 'This spot is popular',  
         'type': 'general',  
         'reasonName': 'globalInteractionReason'}}]
```

```
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4baf59e8f964a520a6f93be3',
'name': 'Bikram Yoga',
'location': {'address': '5500 Broadway',
'crossStreet': '230th Street',
'lat': 40.876843690797934,
'lng': -73.90620384419528,
'labeledLatLngs': [{"label": "display",
'lat': 40.876843690797934,
'lng': -73.90620384419528}],
'distance': 376,
'postalCode': '10463',
'cc': 'US',
'city': 'Bronx',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['5500 Broadway (230th Street)',
'Bronx, NY 10463',
'United States']},
'categories': [{"id": '4bf58dd8d48988d102941735',
'name': 'Yoga Studio',
'pluralName': 'Yoga Studios',
'shortName': 'Yoga Studio',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/gym_yogastudio_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'reerralId': 'e-0-4baf59e8f964a520a6f93be3-1'},
{'reasons': {'count': 0,
'items': [{"summary": 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4b79cc46f964a520c5122fe3',
'name': 'Tibbett Diner',
'location': {'address': '3033 Tibbett Ave',
'crossStreet': 'btwn 230th & 231st',
'lat': 40.8804044222466,
'lng': -73.90893738006402,
'labeledLatLngs': [{"label": "display",
'lat': 40.8804044222466,
'lng': -73.90893738006402},
{'label': 'entrance', 'lat': 40.880388, 'lng': -73.90896}],
'distance': 452,
'postalCode': '10463',
'cc': 'US',
'city': 'Bronx',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['3033 Tibbett Ave (btwn 230th & 231st)',
'Bronx, NY 10463',
'United States']},
'categories': [{"id": '4bf58dd8d48988d147941735',
'name': 'Diner',
'pluralName': 'Diners',
'shortName': 'Diner',
```

```
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/dine  
r_',  
    'suffix': '.png'},  
    'primary': True}],  
    'delivery': {'id': '2539986',  
        'url': 'https://www.seamless.com/menu/tibbett-diner-3033-tibbett-ave-  
bronx/2539986?affiliate=1131&utm_source=foursquare-affiliate-network&utm_medi  
um=affiliate&utm_campaign=1131&utm_content=2539986',  
        'provider': {'name': 'seamless',  
            'icon': {'prefix': 'https://fastly.4sqi.net/img/general/cap/'},  
            'sizes': [40, 50],  
            'name': '/delivery_provider_seamless_20180129.png'}}},  
    'photos': {'count': 0, 'groups': []}},  
    'referralId': 'e-0-4b79cc46f964a520c5122fe3-2'},  
{'reasons': {'count': 0,  
    'items': [{summary: 'This spot is popular',  
        'type': 'general',  
        'reasonName': 'globalInteractionReason'}]}},  
'venue': {'id': '55f81cd2498ee903149fcc64',  
    'name': 'Starbucks',  
    'location': {'address': '171 W 230th St',  
        'crossStreet': 'Kimberly Pl',  
        'lat': 40.87753134921497,  
        'lng': -73.90558216359267,  
        'labeledLatLngs': [{label: 'display',  
            'lat': 40.87753134921497,  
            'lng': -73.90558216359267}],  
        'distance': 441,  
        'postalCode': '10463',  
        'cc': 'US',  
        'city': 'Bronx',  
        'state': 'NY',  
        'country': 'United States',  
        'formattedAddress': ['171 W 230th St (Kimberly Pl)',  
            'Bronx, NY 10463',  
            'United States']},  
    'categories': [{id: '4bf58dd8d48988d1e0931735',  
        'name': 'Coffee Shop',  
        'pluralName': 'Coffee Shops',  
        'shortName': 'Coffee Shop',  
        'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coff  
eeshop_',  
            'suffix': '.png'},  
            'primary': True}],  
        'photos': {'count': 0, 'groups': []}},  
        'referralId': 'e-0-55f81cd2498ee903149fcc64-3'},  
    {'reasons': {'count': 0,  
        'items': [{summary: 'This spot is popular',  
            'type': 'general',  
            'reasonName': 'globalInteractionReason'}]}},  
    'venue': {'id': '4b5357adf964a520319827e3',  
        'name': "Dunkin'",  
        'location': {'address': '5501 Broadway',  
            'crossStreet': 'W 230th St',  
            'lat': 40.87713584201589,  
            'lng': -73.90666550701411,  
            'labeledLatLngs': [{label: 'display',  
                'lat': 40.87713584201589,  
                'lng': -73.90666550701411}]}]
```

```
'lat': 40.87713584201589,
'lng': -73.90666550701411}],
'distance': 342,
'postalCode': '10463',
'cc': 'US',
'city': 'Bronx',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['5501 Broadway (W 230th St)',
'Bronx, NY 10463',
'United States']],
'categories': [{id: '4bf58dd8d48988d148941735',
'name': 'Donut Shop',
'pluralName': 'Donut Shops',
'shortName': 'Donuts',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/donu
ts_',
'suffix': '.png'},
'primary': True}],
'delivery': {id: '2200464',
'url': 'https://www.seamless.com/menu/dunkin-5501-broadway-bronx/2200
464?affiliate=1131&utm_source=foursquare-affiliate-network&utm_medium=affilia
te&utm_campaign=1131&utm_content=2200464',
'provider': {'name': 'seamless',
'icon': {'prefix': 'https://fastly.4sqi.net/img/general/cap/'},
'sizes': [40, 50],
'name': '/delivery_provider_seamless_20180129.png'}}},
'photos': {'count': 0, 'groups': []},
'referralId': 'e-0-4b5357adf964a520319827e3-4'},
{'reasons': {'count': 0,
'items': [{summary: 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {id: '4a725fa1f964a520f6da1fe3',
'name': 'TCR The Club of Riverdale',
'location': {'address': '2600 Netherland Ave',
'lat': 40.8786283,
'lng': -73.9145678,
'labeledLatLngs': [{'label': 'display',
'lat': 40.8786283,
'lng': -73.9145678}],
'distance': 402,
'postalCode': '10463',
'cc': 'US',
'city': 'Bronx',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['2600 Netherland Ave',
'Bronx, NY 10463',
'United States']},
'categories': [{id: '4e39a891bd410d7aed40cbc2',
'name': 'Tennis Stadium',
'pluralName': 'Tennis Stadiums',
'shortName': 'Tennis',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/arts_ent
ertainment/stadium_tennis_',
'suffix': '.png'}},
```

```
'primary': True}],
'photos': {'count': 0, 'groups': []},
'venuePage': {'id': '40358759'}},
'referralId': 'e-0-4a725fa1f964a520f6da1fe3-5'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '55f751ca498eacc0307d1cfe',
'name': 'Blink Fitness',
'location': {'address': '5520 Broadway',
'crossStreet': 'at W 230th St',
'lat': 40.877271495944626,
'lng': -73.90559491338075,
'labeledLatLngs': [{'label': 'display',
'lat': 40.877271495944626,
'lng': -73.90559491338075}],
'distance': 433,
'postalCode': '10463',
'cc': 'US',
'neighborhood': 'Kingsbridge',
'city': 'Bronx',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['5520 Broadway (at W 230th St)',
'Bronx, NY 10463',
'United States']},
'categories': [{id: '4bf58dd8d48988d176941735',
'name': 'Gym',
'pluralName': 'Gyms',
'shortName': 'Gym',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/gym_',
'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-55f751ca498eacc0307d1cfe-6'},
{'reasons': {'count': 0,
'items': [{summary': 'This spot is popular',
'type': 'general',
'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '546d31ca498e561c698a0320',
'name': 'T.J. Maxx',
'location': {'lat': 40.87723198343352,
'lng': -73.90504239962168,
'labeledLatLngs': [{'label': 'display',
'lat': 40.87723198343352,
'lng': -73.90504239962168}],
'distance': 478,
'cc': 'US',
'city': 'Bronx',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['Bronx, NY', 'United States']},
'categories': [{id: '4bf58dd8d48988d1f6941735',
'name': 'Department Store',
'pluralName': 'Department Stores',
```

```
'shortName': 'Department Store',
  'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/departmentstore_',
    'suffix': '.png'},
  'primary': True}],
  'photos': {'count': 0, 'groups': []}},
  'referralId': 'e-0-546d31ca498e561c698a0320-7'},
  {'reasons': {'count': 0,
    'items': [{summary: 'This spot is popular',
      'type': 'general',
      'reasonName': 'globalInteractionReason'}]}},
  'venue': {'id': '4cf6ae55d3a8a1cd71a9d243',
    'name': 'Astral Fitness & Wellness Center',
    'location': {'address': '5500 Broadway',
      'crossStreet': '230th St',
      'lat': 40.87670526507199,
      'lng': -73.90637207670373,
      'labeledLatLngs': [{'label': 'display',
        'lat': 40.87670526507199,
        'lng': -73.90637207670373},
       {'label': 'entrance', 'lat': 40.87685, 'lng': -73.906058}],
      'distance': 361,
      'postalCode': '10463',
      'cc': 'US',
      'city': 'Bronx',
      'state': 'NY',
      'country': 'United States',
      'formattedAddress': ['5500 Broadway (230th St)',
        'Bronx, NY 10463',
        'United States']}},
    'categories': [{'id': '4bf58dd8d48988d176941735',
      'name': 'Gym',
      'pluralName': 'Gyms',
      'shortName': 'Gym',
      'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/gym_',
        'suffix': '.png'},
      'primary': True}],
      'photos': {'count': 0, 'groups': []},
      'venuePage': {'id': '75803748'}},
      'referralId': 'e-0-4cf6ae55d3a8a1cd71a9d243-8'},
      {'reasons': {'count': 0,
        'items': [{summary: 'This spot is popular',
          'type': 'general',
          'reasonName': 'globalInteractionReason'}]}},
      'venue': {'id': '57655be738faa66160da7527',
        'name': 'Starbucks',
        'location': {'address': '50 W 225th St',
          'lat': 40.873754554218515,
          'lng': -73.90861305343668,
          'labeledLatLngs': [{'label': 'display',
            'lat': 40.873754554218515,
            'lng': -73.90861305343668},
           {'label': 'entrance', 'lat': 40.873502, 'lng': -73.907986}],
          'distance': 355,
          'postalCode': '10463',
          'cc': 'US',
```

```
'city': 'New York',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['50 W 225th St',
  'New York, NY 10463',
  'United States']],
'categories': [{id: '4bf58dd8d48988d1e0931735',
  name: 'Coffee Shop',
  pluralName: 'Coffee Shops',
  shortName: 'Coffee Shop',
  icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
    'suffix': '.png'},
  primary}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-57655be738faa66160da7527-9'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '5631194e498e2de074de661c',
  'name': 'Vitamin Shoppe',
  'location': {'address': '5510 Broadway',
    'lat': 40.87716,
    'lng': -73.905632,
    'labeledLatLngs': [{"label": 'display',
      'lat': 40.87716,
      'lng': -73.905632},
     {"label": 'entrance', 'lat': 40.877165, 'lng': -73.905673}],
    'distance': 428,
    'postalCode': '10463',
    'cc': 'US',
    'city': 'Bronx',
    'state': 'NY',
    'country': 'United States',
    'formattedAddress': ['5510 Broadway',
      'Bronx, NY 10463',
      'United States']}],
  'categories': [{id: '5744ccdf4b0c0459246b4cd',
    name: 'Supplement Shop',
    pluralName: 'Supplement Shops',
    shortName: 'Supplement Shop',
    icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/education/lab_',
      'suffix': '.png'},
    primary}],
  'photos': {'count': 0, 'groups': []}},
  'referralId': 'e-0-5631194e498e2de074de661c-10'},
  {'reasons': {'count': 0,
    'items': [{'summary': 'This spot is popular',
      'type': 'general',
      'reasonName': 'globalInteractionReason'}]}},
  'venue': {'id': '4b9c9c6af964a520b27236e3',
    'name': 'Land & Sea Restaurant',
    'location': {'address': '5535 Broadway',
      'crossStreet': '231st St',
      'lat': 40.87788463309788,
```

```
'lng': -73.90587282193539,
'labeledLatLngs': [{label': 'display',
  'lat': 40.87788463309788,
  'lng': -73.90587282193539},
 {'label': 'entrance', 'lat': 40.877906, 'lng': -73.905722}],
'distance': 429,
'postalCode': '10463',
'cc': 'US',
'city': 'Bronx',
'state': 'NY',
'country': 'United States',
'formattedAddress': ['5535 Broadway (231st St)',
 'Bronx, NY 10463',
 'United States']],
'categories': [{id': '4bf58dd8d48988d1ce941735',
  'name': 'Seafood Restaurant',
  'pluralName': 'Seafood Restaurants',
  'shortName': 'Seafood',
  'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/seaf
ood_',
    'suffix': '.png'},
  'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4b9c9c6af964a520b27236e3-11'},
{'reasons': {'count': 0,
  'items': [{summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4b9c9c43f964a520ac7236e3',
  'name': 'Lot Less Closeouts',
  'location': {'address': '5545 Broadway',
    'crossStreet': '231st St',
    'lat': 40.878270422202085,
    'lng': -73.9052646742604,
    'labeledLatLngs': [{label': 'display',
      'lat': 40.878270422202085,
      'lng': -73.9052646742604},
     {'label': 'entrance', 'lat': 40.87828, 'lng': -73.905466}],
    'distance': 492,
    'postalCode': '10463',
    'cc': 'US',
    'city': 'Bronx',
    'state': 'NY',
    'country': 'United States',
    'formattedAddress': ['5545 Broadway (231st St)',
     'Bronx, NY 10463',
     'United States']},
   'categories': [{id': '52dea92d3cf9994f4e043dbb',
     'name': 'Discount Store',
     'pluralName': 'Discount Stores',
     'shortName': 'Discount Store',
     'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/dis
countstore_',
       'suffix': '.png'},
     'primary': True}],
    'photos': {'count': 0, 'groups': []}},
   'referralId': 'e-0-4b9c9c43f964a520ac7236e3-12'}},
```

```
{'reasons': {'count': 0,
  'items': [{{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}}]},
  'venue': {'id': '4b88e053f964a5208a1132e3',
    'name': 'Rite Aid',
    'location': {'address': '5237 Broadway',
      'crossStreet': '228th Street',
      'lat': 40.875466574434704,
      'lng': -73.90890629016033,
      'labeledLatLngs': [{{'label': 'display',
        'lat': 40.875466574434704,
        'lng': -73.90890629016033}],
      'distance': 190,
      'postalCode': '10463',
      'cc': 'US',
      'city': 'Bronx',
      'state': 'NY',
      'country': 'United States',
      'formattedAddress': ['5237 Broadway (228th Street)',
        'Bronx, NY 10463',
        'United States']},
    'categories': [{{'id': '4bf58dd8d48988d10f951735',
      'name': 'Pharmacy',
      'pluralName': 'Pharmacies',
      'shortName': 'Pharmacy',
      'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/pharmacy_',
        'suffix': '.png'},
      'primary': True}],
      'photos': {'count': 0, 'groups': []}},
    'referralId': 'e-0-4b88e053f964a5208a1132e3-13'},
  {'reasons': {'count': 0,
    'items': [{{'summary': 'This spot is popular',
      'type': 'general',
      'reasonName': 'globalInteractionReason'}}]},
    'venue': {'id': '4e4e4517bd4101d0d7a67568',
      'name': 'Baskin-Robbins',
      'location': {'address': '5501 Broadway',
        'lat': 40.87714929478882,
        'lng': -73.90665810372622,
        'labeledLatLngs': [{{'label': 'display',
          'lat': 40.87714929478882,
          'lng': -73.90665810372622},
          {'label': 'entrance', 'lat': 40.877129, 'lng': -73.906608}],
        'distance': 343,
        'postalCode': '10463',
        'cc': 'US',
        'city': 'Bronx',
        'state': 'NY',
        'country': 'United States',
        'formattedAddress': ['5501 Broadway',
          'Bronx, NY 10463',
          'United States']},
      'categories': [{{'id': '4bf58dd8d48988d1c9941735',
        'name': 'Ice Cream Shop',
        'pluralName': 'Ice Cream Shops'}],
```

```
'shortName': 'Ice Cream',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/icecream_',
         'suffix': '.png'},
'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4e4e4517bd4101d0d7a67568-14'},
{'reasons': {'count': 0,
     'items': [{summary: 'This spot is popular',
                'type': 'general',
                'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4b4f7b65f964a5205a0827e3',
          'name': 'Subway',
          'location': {'address': '5209 Broadway',
                       'lat': 40.8746665302951,
                       'lng': -73.9095858429637,
                       'labeledLatLngs': [{'label': 'display',
                               'lat': 40.8746665302951,
                               'lng': -73.9095858429637}],
                       'distance': 228,
                       'postalCode': '10463',
                       'cc': 'US',
                       'city': 'Bronx',
                       'state': 'NY',
                       'country': 'United States',
                       'formattedAddress': ['5209 Broadway',
                                            'Bronx, NY 10463',
                                            'United States']}},
          'categories': [{'id': '4bf58dd8d48988d1c5941735',
                          'name': 'Sandwich Place',
                          'pluralName': 'Sandwich Places',
                          'shortName': 'Sandwiches',
                          'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/deli',
                                   'suffix': '.png'},
                          'primary': True}],
          'photos': {'count': 0, 'groups': []}},
          'referralId': 'e-0-4b4f7b65f964a5205a0827e3-15'},
          {'reasons': {'count': 0,
                      'items': [{summary: 'This spot is popular',
                                 'type': 'general',
                                 'reasonName': 'globalInteractionReason'}]}},
          'venue': {'id': '585c205665e7c70a2f1055ea',
                    'name': 'Boston Market',
                    'location': {'address': '5520 Broadway',
                                 'lat': 40.87743,
                                 'lng': -73.9054121,
                                 'labeledLatLngs': [{'label': 'display',
                                         'lat': 40.87743,
                                         'lng': -73.9054121}],
                                 'distance': 452,
                                 'postalCode': '10463',
                                 'cc': 'US',
                                 'city': 'Bronx',
                                 'state': 'NY',
                                 'country': 'United States',
                                 'formattedAddress': ['5520 Broadway',
                                                     'Bronx, NY 10463',
                                                     'United States']}},
                    'categories': [{'id': '4bf58dd8d48988d1c5941735',
                                    'name': 'Sandwich Place',
                                    'pluralName': 'Sandwich Places',
                                    'shortName': 'Sandwiches',
                                    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/deli',
                                             'suffix': '.png'},
                                    'primary': True}],
                    'photos': {'count': 0, 'groups': []}},
                    'referralId': 'e-0-4b4f7b65f964a5205a0827e3-15'}]
```

```
'Bronx, NY 10463',
'United States']}},
'categories': [{id: '4bf58dd8d48988d14e941735',
  name: 'American Restaurant',
  pluralName: 'American Restaurants',
  shortName: 'American',
  icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/default_',
    'suffix': '.png'},
  primary: True}],
'delivery': {id: '1436334',
  url: 'https://www.seamless.com/menu/boston-market-5520-broadway-brown/1436334?affiliate=1131&utm_source=foursquare-affiliate-network&utm_medium=affiliate&utm_campaign=1131&utm_content=1436334',
  provider: {'name': 'seamless',
    icon: {'prefix': 'https://fastly.4sqi.net/img/general/cap/'},
    sizes: [40, 50],
    name: '/delivery_provider_seamless_20180129.png'}},
  'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-585c205665e7c70a2f1055ea-16'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    type: 'general',
    reasonName: 'globalInteractionReason'}]},
  'venue': {id: '56229ff8498e2abb44b6f12b',
    name: 'Five Below',
    location: {'address': '171 W 230th St',
      'lat': 40.8772564,
      'lng': -73.9049384,
      'labeledLatLngs': [{'label': 'display',
        'lat': 40.8772564,
        'lng': -73.9049384}]},
    'distance': 487,
    'postalCode': '10463',
    'cc': 'US',
    'city': 'Bronx',
    'state': 'NY',
    'country': 'United States',
    'formattedAddress': ['171 W 230th St',
      'Bronx, NY 10463',
      'United States']}},
  'categories': [{id: '52dea92d3cf9994f4e043dbb',
    name: 'Discount Store',
    pluralName: 'Discount Stores',
    shortName: 'Discount Store',
    icon: {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/discountstore_',
      'suffix': '.png'},
    primary: True}],
    'photos': {'count': 0, 'groups': []}},
    'referralId': 'e-0-56229ff8498e2abb44b6f12b-17'},
    {'reasons': {'count': 0,
      'items': [{summary: 'This spot is popular',
        type: 'general',
        reasonName: 'globalInteractionReason'}]},
      'venue': {id: '4b9f030af964a520eb0f37e3',
        name: 'GameStop',
```

```
'location': {'address': '90 W 225th St Ste A-B',
  'crossStreet': 'btw Broadway & Exterior St.',
  'lat': 40.874266802124836,
  'lng': -73.90934218062803,
  'labeledLatLngs': [{'label': 'display',
    'lat': 40.874266802124836,
    'lng': -73.90934218062803}],
  'distance': 277,
  'postalCode': '10463',
  'cc': 'US',
  'city': 'Bronx',
  'state': 'NY',
  'country': 'United States',
  'formattedAddress': ['90 W 225th St Ste A-B (btw Broadway & Exterior
St.)',
    'Bronx, NY 10463',
    'United States']},
 'categories': [{id: '4bf58dd8d48988d10b951735',
  'name': 'Video Game Store',
  'pluralName': 'Video Game Stores',
  'shortName': 'Video Games',
  'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/vid
eogames_',
    'suffix': '.png'},
  'primary': true}],
  'photos': {'count': 0, 'groups': []}},
  'referralId': 'e-0-4b9f030af964a520eb0f37e3-18'},
  {'reasons': {'count': 0,
   'items': [{"summary": 'This spot is popular',
     'type': 'general',
     'reasonName': 'globalInteractionReason'}]}},
  'venue': {'id': '4e4ce4debd413c4cc66d05d0',
    'name': 'Subway',
    'location': {'address': '5549 Broadway',
      'lat': 40.878464979323944,
      'lng': -73.9055176422437,
      'labeledLatLngs': [{'label': 'display',
        'lat': 40.878464979323944,
        'lng': -73.9055176422437},
       {'label': 'entrance', 'lat': 40.878473, 'lng': -73.905329}],
      'distance': 482,
      'postalCode': '10463',
      'cc': 'US',
      'city': 'Bronx',
      'state': 'NY',
      'country': 'United States',
      'formattedAddress': ['5549 Broadway',
        'Bronx, NY 10463',
        'United States']},
    'categories': [{id: '4bf58dd8d48988d1c5941735',
      'name': 'Sandwich Place',
      'pluralName': 'Sandwich Places',
      'shortName': 'Sandwiches',
      'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/deli
_',
        'suffix': '.png'},
      'primary': true}]}]
```

```
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4e4ce4debd413c4cc66d05d0-19'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    type: 'general',
    reasonName: 'globalInteractionReason'}]}},
'venue': {'id': '4dfe40df8877333e195b68fc',
  'name': 'Parrilla Latina',
  'location': {'address': '230th St & Broadway',
    'lat': 40.87747294351472,
    'lng': -73.90607346968568,
    'labeledLatLngs': [{"label": 'display',
      'lat': 40.87747294351472,
      'lng': -73.90607346968568}],
    'distance': 399,
    'postalCode': '10463',
    'cc': 'US',
    'city': 'Bronx',
    'state': 'NY',
    'country': 'United States',
    'formattedAddress': ['230th St & Broadway',
      'Bronx, NY 10463',
      'United States']},
  'categories': [{"id": '4bf58dd8d48988d1cc941735',
    'name': 'Steakhouse',
    'pluralName': 'Steakhouses',
    'shortName': 'Steakhouse',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/steakhouse_',
      'suffix': '.png'},
    'primary': True}],
    'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4dfe40df8877333e195b68fc-20'},
{'reasons': {'count': 0,
  'items': [{summary: 'This spot is popular',
    type: 'general',
    reasonName: 'globalInteractionReason'}]}},
'venue': {'id': '4c852173dc018cfa2bc3e56c',
  'name': "The Children's Place",
  'location': {'address': '44 W 225th St',
    'lat': 40.873671591133125,
    'lng': -73.90815619608166,
    'labeledLatLngs': [{"label": 'display',
      'lat': 40.873671591133125,
      'lng': -73.90815619608166}],
    'distance': 383,
    'postalCode': '10463',
    'cc': 'US',
    'city': 'Bronx',
    'state': 'NY',
    'country': 'United States',
    'formattedAddress': ['44 W 225th St',
      'Bronx, NY 10463',
      'United States']},
  'categories': [{"id": '4bf58dd8d48988d105951735',
    'name': 'Kids Store',
    'pluralName': 'Kids Stores',
```

```

'shortName': 'Kids Store',
'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/app
are_l_kids_',
    'suffix': '.png'},
    'primary': True}],
'photos': {'count': 0, 'groups': []}},
'referralId': 'e-0-4c852173dc018cfab2bc3e56c-21'},
{'reasons': {'count': 0,
    'items': [{summary': 'This spot is popular',
        'type': 'general',
        'reasonName': 'globalInteractionReason'}]}},
'venue': {'id': '4ed7956b8b81b2bf28adc714',
    'name': 'Terrace View Delicatessen',
    'location': {'address': '135 Terrace View Ave.',
        'lat': 40.875995,
        'lng': -73.913151,
        'labeledLatLngs': [{'label': 'display',
            'lat': 40.875995,
            'lng': -73.913151},
            {'label': 'entrance', 'lat': 40.875964, 'lng': -73.912985}],
        'distance': 218,
        'postalCode': '10034',
        'cc': 'US',
        'city': 'New York',
        'state': 'NY',
        'country': 'United States',
        'formattedAddress': ['135 Terrace View Ave.',
            'New York, NY 10034',
            'United States']},
    'categories': [{id: '4bf58dd8d48988d146941735',
        'name': 'Deli / Bodega',
        'pluralName': 'Delis / Bodegas',
        'shortName': 'Deli / Bodega',
        'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/deli
        _',
            'suffix': '.png'},
            'primary': True}],
    'photos': {'count': 0, 'groups': []}}},
'referralId': 'e-0-4ed7956b8b81b2bf28adc714-22'}]}

```

In [212]:

```

# function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']

```

In [213]:

```

venues = results['response']['groups'][0]['items']

nearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.location.lng']
nearby_venues = nearby_venues.loc[:, filtered_columns]

# filter the category for each row
nearby_venues['venue.categories'] = nearby_venues.apply(get_category_type, axis=1)

# clean columns
nearby_venues.columns = [col.split(".")[-1] for col in nearby_venues.columns]

nearby_venues.head()

```

/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/ipykernel_launcher.py:3: FutureWarning: pandas.io.json.json_normalize is deprecated, use pandas.json_normalize instead

This is separate from the ipykernel package so we can avoid doing imports until

Out[213]:

	name	categories	lat	lng
0	Arturo's	Pizza Place	40.874412	-73.910271
1	Bikram Yoga	Yoga Studio	40.876844	-73.906204
2	Tibbett Diner	Diner	40.880404	-73.908937
3	Starbucks	Coffee Shop	40.877531	-73.905582
4	Dunkin'	Donut Shop	40.877136	-73.906666

In [214]:

```
print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))
```

23 venues were returned by Foursquare.

2. Explore Neighborhoods in Manhattan

In [215]:

```
def getNearbyVenues(names, latitudes, longitudes, radius=500):  
  
    venues_list=[]  
    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_list.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_list in venues_list for item in venue_list])  
    nearby_venues.columns = ['Neighborhood',  
                            'Neighborhood Latitude',  
                            'Neighborhood Longitude',  
                            'Venue',  
                            'Venue Latitude',  
                            'Venue Longitude',  
                            'Venue Category']  
  
    return(nearby_venues)
```

Now write the code to run the above function on each neighborhood and create a new dataframe called `_manhattanvenues`.

In [216]:

```
# type your answer here
manhattan_venues = getNearbyVenues(names=manhattan_data['Neighborhood'],latitudes=manhatta
n_data['Latitude'],longitudes=manhattan_data['Longitude'])
```

Marble Hill
Chinatown
Washington Heights
Inwood
Hamilton Heights
Manhattanville
Central Harlem
East Harlem
Upper East Side
Yorkville
Lenox Hill
Roosevelt Island
Upper West Side
Lincoln Square
Clinton
Midtown
Murray Hill
Chelsea
Greenwich Village
East Village
Lower East Side
Tribeca
Little Italy
Soho
West Village
Manhattan Valley
Morningside Heights
Gramercy
Battery Park City
Financial District
Carnegie Hill
Noho
Civic Center
Midtown South
Sutton Place
Turtle Bay
Tudor City
Stuyvesant Town
Flatiron
Hudson Yards

In [217]:

```
print(manhattan_venues.shape)
manhattan_venues.head()
```

(3171, 7)

Out[217]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Marble Hill	40.876551	-73.91066	Arturo's	40.874412	-73.910271	Pizza Place
1	Marble Hill	40.876551	-73.91066	Bikram Yoga	40.876844	-73.906204	Yoga Studio
2	Marble Hill	40.876551	-73.91066	Tibbett Diner	40.880404	-73.908937	Diner
3	Marble Hill	40.876551	-73.91066	Starbucks	40.877531	-73.905582	Coffee Shop
4	Marble Hill	40.876551	-73.91066	Dunkin'	40.877136	-73.906666	Donut Shop

In [218]:

```
manhattan_venues.groupby('Neighborhood').count()
```

Out[218]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Battery Park City	70	70	70	70	70	70
Carnegie Hill	86	86	86	86	86	86
Central Harlem	44	44	44	44	44	44
Chelsea	100	100	100	100	100	100
Chinatown	100	100	100	100	100	100
Civic Center	100	100	100	100	100	100
Clinton	100	100	100	100	100	100
East Harlem	39	39	39	39	39	39
East Village	100	100	100	100	100	100
Financial District	100	100	100	100	100	100
Flatiron	100	100	100	100	100	100
Gramercy	86	86	86	86	86	86
Greenwich Village	100	100	100	100	100	100
Hamilton Heights	63	63	63	63	63	63
Hudson Yards	65	65	65	65	65	65
Inwood	57	57	57	57	57	57
Lenox Hill	100	100	100	100	100	100
Lincoln Square	93	93	93	93	93	93
Little Italy	100	100	100	100	100	100
Lower East Side	50	50	50	50	50	50
Manhattan Valley	48	48	48	48	48	48
Manhattanville	45	45	45	45	45	45
Marble Hill	23	23	23	23	23	23
Midtown	100	100	100	100	100	100
Midtown South	100	100	100	100	100	100
Morningside Heights	40	40	40	40	40	40
Murray Hill	100	100	100	100	100	100
Noho	100	100	100	100	100	100
Roosevelt Island	23	23	23	23	23	23

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Neighborhood						
Soho	100	100	100	100	100	100
Stuyvesant Town	18	18	18	18	18	18
Sutton Place	100	100	100	100	100	100
Tribeca	82	82	82	82	82	82
Tudor City	76	76	76	76	76	76
Turtle Bay	100	100	100	100	100	100
Upper East Side	96	96	96	96	96	96
Upper West Side	84	84	84	84	84	84
Washington Heights	83	83	83	83	83	83
West Village	100	100	100	100	100	100
Yorkville	100	100	100	100	100	100

Let's find out how many unique categories can be curated from all the returned venues

In [219]:

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

```

There are 333 uniques categories.

Analyze Each Neighborhood

In [220]:

```
# one hot encoding
manhattan_onehot = pd.get_dummies(manhattan_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
manhattan_onehot['Neighborhood'] = manhattan_venues['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [manhattan_onehot.columns[-1]] + list(manhattan_onehot.columns[:-1])
manhattan_onehot = manhattan_onehot[fixed_columns]

manhattan_onehot.head()
```

Out[220]:

	Neighborhood	Accessories Store	Adult Boutique	Afghan Restaurant	African Restaurant	American Restaurant	Antique Shop	Arep Restaurant
0	Marble Hill	0	0	0	0	0	0	0
1	Marble Hill	0	0	0	0	0	0	0
2	Marble Hill	0	0	0	0	0	0	0
3	Marble Hill	0	0	0	0	0	0	0
4	Marble Hill	0	0	0	0	0	0	0

In [221]:

```
manhattan_onehot.shape
```

Out[221]:

(3171, 334)

Next, let's group rows by neighborhood and by taking the mean of the frequency of occurrence of each category

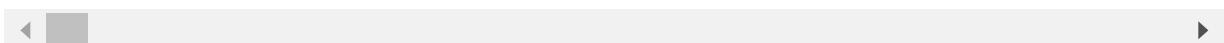
In [222]:

```
manhattan_grouped = manhattan_onehot.groupby('Neighborhood').mean().reset_index()
manhattan_grouped
```

Out[222]:

	Neighborhood	Accessories Store	Adult Boutique	Afghan Restaurant	African Restaurant	American Restaurant	Antique Shop	Are Restaura
0	Battery Park City	0.000000	0.000000	0.00	0.000000	0.000000	0.00	0.0000
1	Carnegie Hill	0.000000	0.000000	0.00	0.000000	0.011628	0.00	0.0000
2	Central Harlem	0.000000	0.000000	0.00	0.068182	0.045455	0.00	0.0000
3	Chelsea	0.000000	0.000000	0.00	0.000000	0.030000	0.00	0.0000
4	Chinatown	0.000000	0.000000	0.00	0.000000	0.040000	0.00	0.0000
5	Civic Center	0.000000	0.000000	0.00	0.000000	0.020000	0.01	0.0000
6	Clinton	0.000000	0.000000	0.00	0.000000	0.040000	0.00	0.0000
7	East Harlem	0.000000	0.000000	0.00	0.000000	0.000000	0.00	0.0000
8	East Village	0.000000	0.000000	0.00	0.000000	0.010000	0.00	0.0000
9	Financial District	0.000000	0.000000	0.00	0.000000	0.020000	0.00	0.0000
10	Flatiron	0.000000	0.000000	0.00	0.000000	0.040000	0.00	0.0000
11	Gramercy	0.000000	0.000000	0.00	0.000000	0.046512	0.00	0.0000
12	Greenwich Village	0.010000	0.000000	0.00	0.000000	0.030000	0.00	0.0000
13	Hamilton Heights	0.000000	0.015873	0.00	0.000000	0.000000	0.00	0.0000
14	Hudson Yards	0.000000	0.000000	0.00	0.000000	0.061538	0.00	0.0000
15	Inwood	0.000000	0.000000	0.00	0.000000	0.017544	0.00	0.0000
16	Lenox Hill	0.000000	0.000000	0.01	0.000000	0.000000	0.00	0.0000
17	Lincoln Square	0.000000	0.000000	0.00	0.000000	0.021505	0.00	0.0000
18	Little Italy	0.000000	0.000000	0.00	0.000000	0.000000	0.00	0.0000
19	Lower East Side	0.000000	0.000000	0.00	0.000000	0.020000	0.00	0.0000
20	Manhattan Valley	0.000000	0.000000	0.00	0.000000	0.020833	0.00	0.0000
21	Manhattanville	0.000000	0.000000	0.00	0.000000	0.022222	0.00	0.0000
22	Marble Hill	0.000000	0.000000	0.00	0.000000	0.043478	0.00	0.0000
23	Midtown	0.000000	0.000000	0.00	0.000000	0.020000	0.00	0.0000
24	Midtown South	0.000000	0.000000	0.00	0.000000	0.020000	0.00	0.0000
25	Morningside Heights	0.000000	0.000000	0.00	0.000000	0.075000	0.00	0.0000
26	Murray Hill	0.000000	0.000000	0.00	0.000000	0.040000	0.00	0.0000
27	Noho	0.000000	0.010000	0.00	0.000000	0.000000	0.00	0.0000

	Neighborhood	Accessories Store	Adult Boutique	Afghan Restaurant	African Restaurant	American Restaurant	Antique Shop	Are Restaura
28	Roosevelt Island	0.000000	0.000000	0.00	0.000000	0.000000	0.00	0.0000
29	Soho	0.000000	0.000000	0.00	0.000000	0.020000	0.00	0.0000
30	Stuyvesant Town	0.000000	0.000000	0.00	0.000000	0.000000	0.00	0.0000
31	Sutton Place	0.000000	0.010000	0.00	0.000000	0.020000	0.00	0.0000
32	Tribeca	0.000000	0.000000	0.00	0.000000	0.048780	0.00	0.0000
33	Tudor City	0.000000	0.000000	0.00	0.000000	0.013158	0.00	0.0000
34	Turtle Bay	0.000000	0.000000	0.00	0.000000	0.010000	0.00	0.0000
35	Upper East Side	0.000000	0.000000	0.00	0.000000	0.041667	0.00	0.0000
36	Upper West Side	0.000000	0.000000	0.00	0.000000	0.023810	0.00	0.0000
37	Washington Heights	0.012048	0.000000	0.00	0.000000	0.012048	0.00	0.0120
38	West Village	0.010000	0.010000	0.00	0.000000	0.050000	0.00	0.0000
39	Yorkville	0.000000	0.000000	0.00	0.000000	0.000000	0.00	0.0000



Let's confirm the new size

In [223]:

```
manhattan_grouped.shape
```

Out[223]:

(40, 334)

In []:



Let's print each neighborhood along with the top 5 most common venues

In [224]:

```
num_top_venues = 5

for hood in manhattan_grouped['Neighborhood']:

    temp = manhattan_grouped[manhattan_grouped['Neighborhood'] == hood].T.reset_index()
    temp.columns = ['venue', 'freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print("----"+hood+"----")
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_top_venues))
    print('\n')
```

----Battery Park City----

	venue	freq
0	Coffee Shop	0.09
1	Hotel	0.07
2	Park	0.06
3	Memorial Site	0.04
4	Gym	0.04

----Carnegie Hill----

	venue	freq
0	Coffee Shop	0.09
1	Café	0.05
2	Yoga Studio	0.03
3	Pizza Place	0.03
4	Cosmetics Shop	0.03

----Central Harlem----

	venue	freq
0	African Restaurant	0.07
1	Bar	0.05
2	Chinese Restaurant	0.05
3	American Restaurant	0.05
4	Seafood Restaurant	0.05

----Chelsea----

	venue	freq
0	Art Gallery	0.06
1	Coffee Shop	0.06
2	Bakery	0.05
3	French Restaurant	0.04
4	American Restaurant	0.03

----Chinatown----

	venue	freq
0	Chinese Restaurant	0.08
1	Bakery	0.07
2	American Restaurant	0.04
3	Hotpot Restaurant	0.04
4	Cocktail Bar	0.04

----Civic Center----

	venue	freq
0	Coffee Shop	0.07
1	Gym / Fitness Center	0.05
2	Spa	0.05
3	Cocktail Bar	0.04
4	Hotel	0.04

----Clinton----

	venue	freq
0	Italian Restaurant	0.06

```

1 Gym / Fitness Center 0.05
2 Coffee Shop 0.05
3 Theater 0.05
4 American Restaurant 0.04

```

----East Harlem----

	venue	freq
0	Mexican Restaurant	0.13
1	Bakery	0.10
2	Deli / Bodega	0.08
3	Thai Restaurant	0.08
4	Latin American Restaurant	0.05

----East Village----

	venue	freq
0	Bar	0.07
1	Pizza Place	0.05
2	Wine Bar	0.04
3	Mexican Restaurant	0.03
4	Korean Restaurant	0.03

----Financial District----

	venue	freq
0	Coffee Shop	0.10
1	Bar	0.05
2	Cocktail Bar	0.04
3	Italian Restaurant	0.04
4	Gym	0.03

----Flatiron----

	venue	freq
0	Italian Restaurant	0.05
1	Japanese Restaurant	0.04
2	Spa	0.04
3	New American Restaurant	0.04
4	Gym / Fitness Center	0.04

----Gramercy----

	venue	freq
0	Bagel Shop	0.06
1	Bar	0.06
2	Pizza Place	0.05
3	American Restaurant	0.05
4	Italian Restaurant	0.05

----Greenwich Village----

	venue	freq
0	Italian Restaurant	0.10
1	Clothing Store	0.06
2	Sushi Restaurant	0.05
3	Café	0.03

4 American Restaurant 0.03

----Hamilton Heights----

	venue	freq
0	Pizza Place	0.10
1	Coffee Shop	0.06
2	Café	0.06
3	Mexican Restaurant	0.05
4	Deli / Bodega	0.05

----Hudson Yards----

	venue	freq
0	Gym / Fitness Center	0.08
1	American Restaurant	0.06
2	Hotel	0.05
3	Italian Restaurant	0.05
4	Café	0.03

----Inwood----

	venue	freq
0	Café	0.07
1	Mexican Restaurant	0.07
2	Restaurant	0.05
3	Lounge	0.05
4	Chinese Restaurant	0.04

----Lenox Hill----

	venue	freq
0	Italian Restaurant	0.07
1	Sushi Restaurant	0.05
2	Cocktail Bar	0.05
3	Coffee Shop	0.05
4	Pizza Place	0.04

----Lincoln Square----

	venue	freq
0	Plaza	0.05
1	Gym / Fitness Center	0.05
2	Café	0.05
3	Italian Restaurant	0.04
4	Concert Hall	0.04

----Little Italy----

	venue	freq
0	Bakery	0.05
1	Café	0.04
2	Chinese Restaurant	0.04
3	Cocktail Bar	0.04
4	Bubble Tea Shop	0.03

----Lower East Side----

	venue	freq
0	Chinese Restaurant	0.06
1	Art Gallery	0.06
2	Japanese Restaurant	0.04
3	Café	0.04
4	Coffee Shop	0.04

----Manhattan Valley----

	venue	freq
0	Coffee Shop	0.06
1	Yoga Studio	0.04
2	Mexican Restaurant	0.04
3	Indian Restaurant	0.04
4	Thai Restaurant	0.04

----Manhattanville----

	venue	freq
0	Coffee Shop	0.09
1	Sushi Restaurant	0.07
2	Seafood Restaurant	0.07
3	Deli / Bodega	0.04
4	Italian Restaurant	0.04

----Marble Hill----

	venue	freq
0	Gym	0.09
1	Discount Store	0.09
2	Sandwich Place	0.09
3	Coffee Shop	0.09
4	Yoga Studio	0.04

----Midtown----

	venue	freq
0	Hotel	0.07
1	Clothing Store	0.06
2	Coffee Shop	0.06
3	Theater	0.04
4	Sporting Goods Shop	0.03

----Midtown South----

	venue	freq
0	Korean Restaurant	0.15
1	Hotel	0.07
2	Hotel Bar	0.05
3	Gym / Fitness Center	0.04
4	Salad Place	0.03

----Morningside Heights----

	venue	freq
0	Coffee Shop	0.10

1	American Restaurant	0.08
2	Park	0.08
3	Bookstore	0.08
4	Café	0.05

----Murray Hill----

	venue	freq
0	Coffee Shop	0.06
1	Sandwich Place	0.05
2	Bar	0.05
3	Hotel	0.04
4	American Restaurant	0.04

----Noho----

	venue	freq
0	Italian Restaurant	0.05
1	Coffee Shop	0.05
2	Art Gallery	0.05
3	Mexican Restaurant	0.04
4	Pizza Place	0.03

----Roosevelt Island----

	venue	freq
0	Park	0.13
1	Soccer Field	0.04
2	Food & Drink Shop	0.04
3	Coffee Shop	0.04
4	Metro Station	0.04

----Soho----

	venue	freq
0	Clothing Store	0.10
1	Italian Restaurant	0.08
2	Coffee Shop	0.05
3	Mediterranean Restaurant	0.04
4	Bakery	0.04

----Stuyvesant Town----

	venue	freq
0	Park	0.17
1	Boat or Ferry	0.11
2	Coffee Shop	0.11
3	Bar	0.11
4	Baseball Field	0.06

----Sutton Place----

	venue	freq
0	Italian Restaurant	0.06
1	Coffee Shop	0.06
2	Gym / Fitness Center	0.04
3	Park	0.04

4 Pizza Place 0.03

----Tribeca----

	venue	freq
0	Italian Restaurant	0.06
1	Park	0.06
2	American Restaurant	0.05
3	Wine Bar	0.05
4	Café	0.04

----Tudor City----

	venue	freq
0	Park	0.07
1	Café	0.05
2	Mexican Restaurant	0.05
3	Sushi Restaurant	0.04
4	Deli / Bodega	0.04

----Turtle Bay----

	venue	freq
0	Coffee Shop	0.06
1	Italian Restaurant	0.05
2	Sushi Restaurant	0.03
3	Deli / Bodega	0.03
4	Ramen Restaurant	0.03

----Upper East Side----

	venue	freq
0	Italian Restaurant	0.06
1	Coffee Shop	0.06
2	Exhibit	0.06
3	Gym / Fitness Center	0.05
4	Bakery	0.05

----Upper West Side----

	venue	freq
0	Bar	0.05
1	Wine Bar	0.05
2	Café	0.05
3	Bakery	0.05
4	Italian Restaurant	0.04

----Washington Heights----

	venue	freq
0	Café	0.07
1	Bakery	0.05
2	Mobile Phone Shop	0.04
3	Spanish Restaurant	0.04
4	Grocery Store	0.04

----West Village----

	venue	freq
0	Italian Restaurant	0.11
1	New American Restaurant	0.05
2	Cocktail Bar	0.05
3	American Restaurant	0.05
4	Park	0.04

----Yorkville----

	venue	freq
0	Italian Restaurant	0.08
1	Coffee Shop	0.06
2	Bar	0.05
3	Gym	0.05
4	Deli / Bodega	0.04

In [225]:

```
# Create dataframe for individual neighborhoods
df_batterypark = pd.DataFrame(np.array([[0, 'Coffee Shop', 0.09], [1, 'Hotel', 0.07], [2, 'Park', 0.06], [3, 'Memorial Site', 0.04], [4, 'Gym', 0.04]]), columns=['Battery Park City', 'venue', 'freq'])
df_batterypark
```

Out[225]:

Battery Park City	venue	freq
0	Coffee Shop	0.09
1	Hotel	0.07
2	Park	0.06
3	Memorial Site	0.04
4	Gym	0.04

In [226]:

```
df_carnegiehill = pd.DataFrame(np.array([[0, 'Coffee Shop', 0.09], [1, 'Café', 0.05], [2, 'Yoga Studio', 0.03], [3, 'Pizza Place', 0.03], [4, 'Cosmetics Shop', 0.03]]), columns=['Carnegie Hill', 'venue', 'freq'])
df_carnegiehill
```

Out[226]:

Carnegie Hill		venue	freq
0	0	Coffee Shop	0.09
1	1	Café	0.05
2	2	Yoga Studio	0.03
3	3	Pizza Place	0.03
4	4	Cosmetics Shop	0.03

In [227]:

```
df_central_harlem = pd.DataFrame(np.array([[0, 'African Restaurant', 0.07], [1, 'Bar', 0.05], [2, 'Chinese Restaurant', 0.05], [3, 'American Restaurant', 0.05], [4, 'Seafood Restaurant', 0.05]]), columns=['Central Harlem', 'venue', 'freq'])
df_central_harlem
```

Out[227]:

Central Harlem		venue	freq
0	0	African Restaurant	0.07
1	1	Bar	0.05
2	2	Chinese Restaurant	0.05
3	3	American Restaurant	0.05
4	4	Seafood Restaurant	0.05

In [228]:

```
df_chelsea = pd.DataFrame(np.array([[0, 'Art Gallery', 0.06], [1, 'Coffee Shop', 0.06], [2, 'Bakery', 0.05], [3, 'French Restaurant', 0.04], [4, 'American Restaurant', 0.03]]), columns=['Chelsea', 'venue', 'freq'])
df_chelsea
```

Out[228]:

	Chelsea	venue	freq
0	0	Art Gallery	0.06
1	1	Coffee Shop	0.06
2	2	Bakery	0.05
3	3	French Restaurant	0.04
4	4	American Restaurant	0.03

In [229]:

```
df_chinatown = pd.DataFrame(np.array([[0, 'Chinese Restaurant', 0.08], [1, 'Bakery', 0.07], [2, 'Bakery', 0.04], [3, 'Hotpot Restaurant', 0.04], [4, 'Cocktail Bar', 0.04]]), columns=['Chinatown', 'venue', 'freq'])
df_chinatown
```

Out[229]:

	Chinatown	venue	freq
0	0	Chinese Restaurant	0.08
1	1	Bakery	0.07
2	2	Bakery	0.04
3	3	Hotpot Restaurant	0.04
4	4	Cocktail Bar	0.04

In [230]:

```
df_batterypark['freq'] = df_batterypark['freq'].astype(float)
print (df_batterypark)
print (df_batterypark.dtypes)

df_carnegiehill['freq'] = df_carnegiehill['freq'].astype(float)
print (df_carnegiehill)
print (df_carnegiehill.dtypes)

df_central_harlem['freq'] = df_central_harlem['freq'].astype(float)
print (df_central_harlem)
print (df_central_harlem.dtypes)

df_chelsea['freq'] = df_chelsea['freq'].astype(float)
print (df_chelsea)
print (df_chelsea.dtypes)

df_chinatown['freq'] = df_chinatown['freq'].astype(float)
print (df_chinatown)
print (df_chinatown.dtypes)
```

```
Battery Park City      venue  freq
0                      0    Coffee Shop  0.09
1                      1        Hotel  0.07
2                      2        Park  0.06
3                      3 Memorial Site  0.04
4                      4        Gym  0.04
Battery Park City      object
venue                  object
freq                  float64
dtype: object
Carnegie Hill          venue  freq
0                      0    Coffee Shop  0.09
1                      1       Café  0.05
2                      2  Yoga Studio  0.03
3                      3 Pizza Place  0.03
4                      4 Cosmetics Shop  0.03
Carnegie Hill          object
venue                  object
freq                  float64
dtype: object
Central Harlem         venue  freq
0                      0 African Restaurant  0.07
1                      1           Bar  0.05
2                      2 Chinese Restaurant  0.05
3                      3 American Restaurant  0.05
4                      4 Seafood Restaurant  0.05
Central Harlem         object
venue                  object
freq                  float64
dtype: object
Chelsea                venue  freq
0                      0 Art Gallery  0.06
1                      1 Coffee Shop  0.06
2                      2       Bakery  0.05
3                      3 French Restaurant  0.04
4                      4 American Restaurant  0.03
Chelsea                object
venue                  object
freq                  float64
dtype: object
Chinatown               venue  freq
0                      0 Chinese Restaurant  0.08
1                      1       Bakery  0.07
2                      2       Bakery  0.04
3                      3 Hotpot Restaurant  0.04
4                      4 Cocktail Bar  0.04
Chinatown               object
venue                  object
freq                  float64
dtype: object
```

In [231]:

```
df_batterypark.dtypes
```

Out[231]:

```
Battery Park City      object
venue                  object
freq                   float64
dtype: object
```

In [232]:

```
df_batterypark
```

Out[232]:

	Battery Park City	venue	freq
0	0	Coffee Shop	0.09
1	1	Hotel	0.07
2	2	Park	0.06
3	3	Memorial Site	0.04
4	4	Gym	0.04

In [233]:

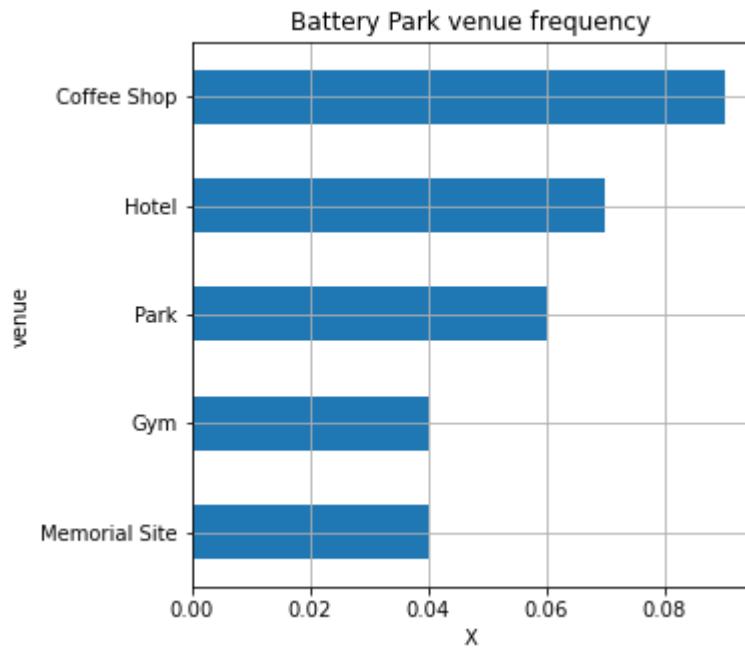
```
# Create dataframe for individual neighborhoods
df_batterypark1 = pd.DataFrame(np.array([[ 'Coffee Shop', 0.09],[ 'Hotel', 0.07],[ 'Park', 0.06],[ 'Memorial Site', 0.04],
[ 'Gym', 0.04]]),columns=[ 'venue', 'freq'])
df_batterypark1
```

Out[233]:

	venue	freq
0	Coffee Shop	0.09
1	Hotel	0.07
2	Park	0.06
3	Memorial Site	0.04
4	Gym	0.04

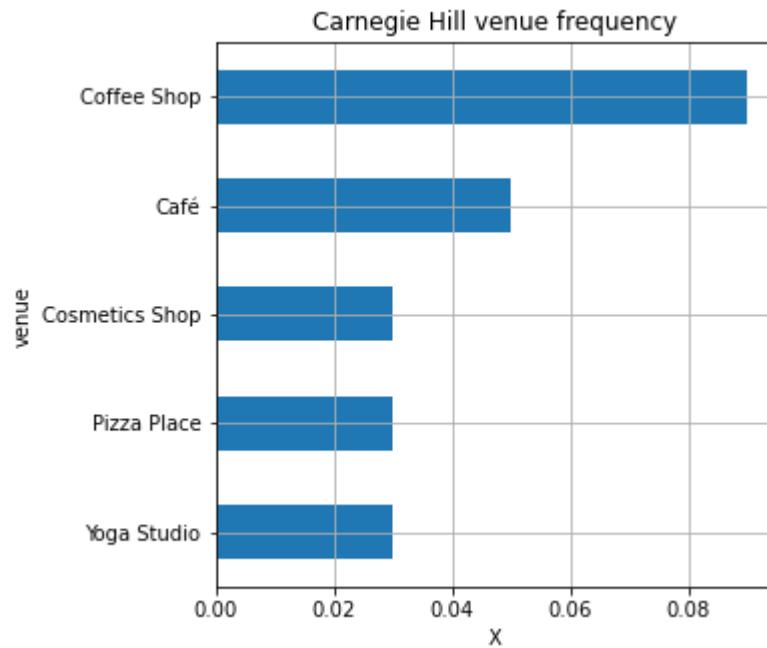
In [234]:

```
df_batterypark.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('X')
plt.title('Battery Park venue frequency')
plt.show()
```



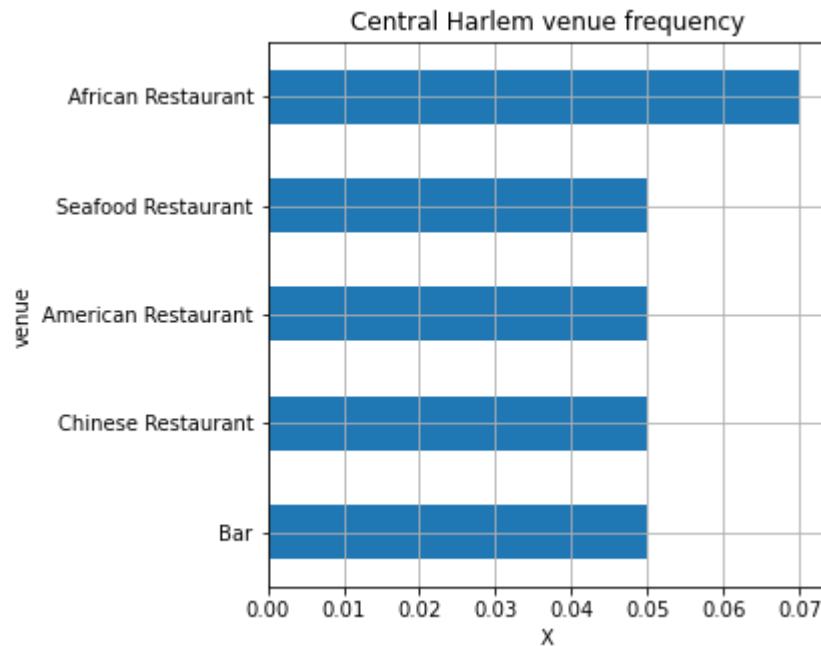
In [235]:

```
df_carnegiehill.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('X')
plt.title('Carnegie Hill venue frequency')
plt.show()
```



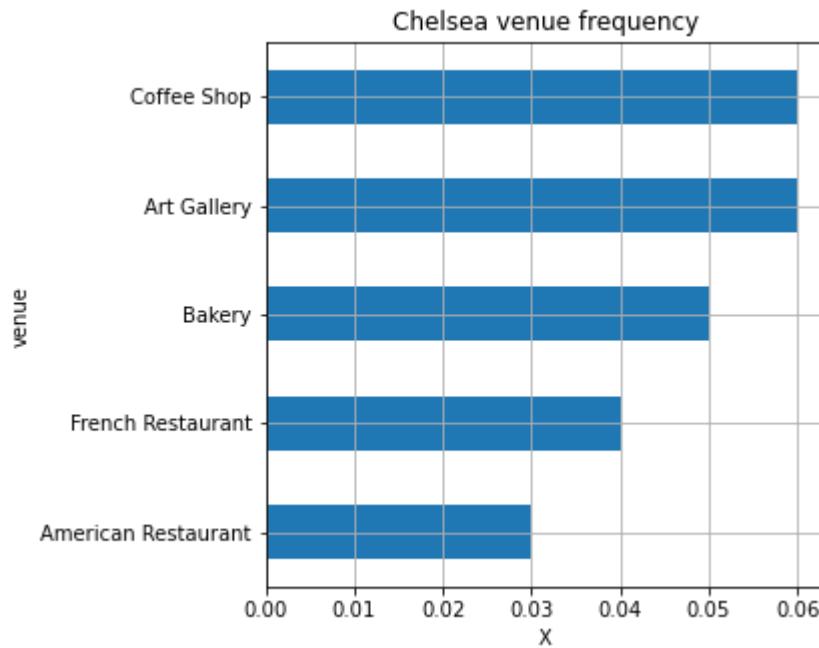
In [236]:

```
df_central_harlem.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('X')
plt.title('Central Harlem venue frequency')
plt.show()
```



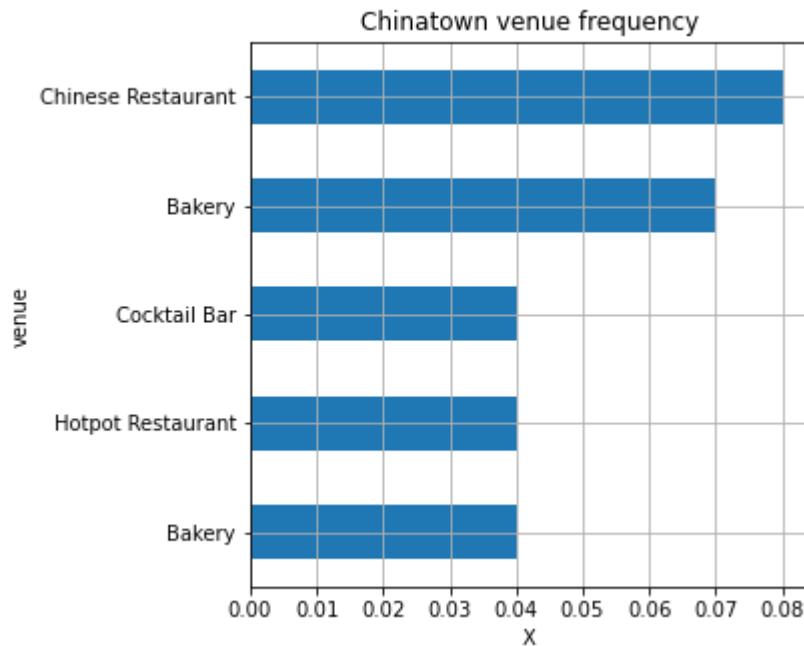
In [237]:

```
df_chelsea.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('X')
plt.title('Chelsea venue frequency')
plt.show()
```



In [238]:

```
df_chinatown.set_index('venue')['freq'].sort_values().plot(kind='barh', grid=True, figsize=(5,5))
plt.xlabel('X')
plt.title('Chinatown venue frequency')
plt.show()
```



In []:

Let's put that into a *pandas* dataframe

Writing a function to sort the venues in descending order.

In [239]:

```
def return_most_common_venues(row, num_top_venues):
    row_categories = row.iloc[1:]
    row_categories_sorted = row_categories.sort_values(ascending=False)

    return row_categories_sorted.index.values[0:num_top_venues]
```

In [240]:

```

num_top_venues = 10

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['Neighborhood']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{0}{1} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{0}th Most Common Venue'.format(ind+1))

# create a new dataframe
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = manhattan_grouped['Neighborhood']

for ind in np.arange(manhattan_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(manhattan_grouped.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted.head()

```

Out[240]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
0	Battery Park City	Coffee Shop	Hotel	Park	Clothing Store	Gym	Memorial Site	Food Court
1	Carnegie Hill	Coffee Shop	Café	Yoga Studio	Pizza Place	Wine Shop	Cosmetics Shop	Bookstore
2	Central Harlem	African Restaurant	Chinese Restaurant	French Restaurant	Gym / Fitness Center	American Restaurant	Bar	Cosmetics Shop
3	Chelsea	Coffee Shop	Art Gallery	Bakery	French Restaurant	Ice Cream Shop	Wine Shop	Seafood Restaurant
4	Chinatown	Chinese Restaurant	Bakery	American Restaurant	Hotpot Restaurant	Cocktail Bar	Salon / Barbershop	Desser Shop

Cluster Neighborhoods

Run k-means to cluster the neighborhood into 5 clusters.

In [241]:

```
# set number of clusters
kclusters = 5

manhattan_grouped_clustering = manhattan_grouped.drop('Neighborhood', 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(manhattan_grouped_clustering)

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:10]
```

Out[241]:

```
array([2, 2, 0, 2, 0, 2, 2, 3, 0, 2], dtype=int32)
```

Let's create a new dataframe that includes the cluster as well as the top 10 venues for each neighborhood.

In [242]:

```
# add clustering Labels
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

manhattan_merged = manhattan_data

# merge manhattan_grouped with manhattan_data to add Latitude/Longitude for each neighborhood
manhattan_merged = manhattan_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Neighborhood')

manhattan_merged.head() # check the last columns!
```

Out[242]:

	Borough	Neighborhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
0	Manhattan	Marble Hill	40.876551	-73.910660	2	Discount Store	Sandwich Place	Coffee Shop
1	Manhattan	Chinatown	40.715618	-73.994279	0	Chinese Restaurant	Bakery	American Restaurant
2	Manhattan	Washington Heights	40.851903	-73.936900	3	Café	Bakery	Spanish Restaurant
3	Manhattan	Inwood	40.867684	-73.921210	3	Café	Mexican Restaurant	Lounge
4	Manhattan	Hamilton Heights	40.823604	-73.949688	3	Pizza Place	Café	Coffee Shop

In [243]:

```
manhattan_merged.groupby('Neighborhood').count()
```

Out[243]:

Neighborhood	Borough	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Battery Park City	1	1	1	1	1	1	1	1	1
Carnegie Hill	1	1	1	1	1	1	1	1	1
Central Harlem	1	1	1	1	1	1	1	1	1
Chelsea	1	1	1	1	1	1	1	1	1
Chinatown	1	1	1	1	1	1	1	1	1
Civic Center	1	1	1	1	1	1	1	1	1
Clinton	1	1	1	1	1	1	1	1	1
East Harlem	1	1	1	1	1	1	1	1	1
East Village	1	1	1	1	1	1	1	1	1
Financial District	1	1	1	1	1	1	1	1	1
Flatiron	1	1	1	1	1	1	1	1	1
Gramercy	1	1	1	1	1	1	1	1	1
Greenwich Village	1	1	1	1	1	1	1	1	1
Hamilton Heights	1	1	1	1	1	1	1	1	1
Hudson Yards	1	1	1	1	1	1	1	1	1
Inwood	1	1	1	1	1	1	1	1	1
Lenox Hill	1	1	1	1	1	1	1	1	1
Lincoln Square	1	1	1	1	1	1	1	1	1
Little Italy	1	1	1	1	1	1	1	1	1
Lower East Side	1	1	1	1	1	1	1	1	1
Manhattan Valley	1	1	1	1	1	1	1	1	1
Manhattanville	1	1	1	1	1	1	1	1	1
Marble Hill	1	1	1	1	1	1	1	1	1
Midtown	1	1	1	1	1	1	1	1	1
Midtown South	1	1	1	1	1	1	1	1	1
Morningside Heights	1	1	1	1	1	1	1	1	1

Neighborhood	Borough	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Murray Hill	1	1	1	1	1	1	1	1	1
Noho	1	1	1	1	1	1	1	1	1
Roosevelt Island	1	1	1	1	1	1	1	1	1
Soho	1	1	1	1	1	1	1	1	1
Stuyvesant Town	1	1	1	1	1	1	1	1	1
Sutton Place	1	1	1	1	1	1	1	1	1
Tribeca	1	1	1	1	1	1	1	1	1
Tudor City	1	1	1	1	1	1	1	1	1
Turtle Bay	1	1	1	1	1	1	1	1	1
Upper East Side	1	1	1	1	1	1	1	1	1
Upper West Side	1	1	1	1	1	1	1	1	1
Washington Heights	1	1	1	1	1	1	1	1	1
West Village	1	1	1	1	1	1	1	1	1
Yorkville	1	1	1	1	1	1	1	1	1



Finally, let's visualize the resulting clusters

In [244]:

```
# create map
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(manhattan_merged['Latitude'], manhattan_merged['Longitude'], manhattan_merged['Neighborhood'], manhattan_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[244]:

Make this Notebook Trusted to load map: File -> Trust Notebook

Examine Clusters

Examining each cluster and determining the discriminating venue categories that distinguish each cluster.

Cluster 1

In [245]:

```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 0, manhattan_merged.columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

Out[245]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Comm Ven
1	Chinatown	Chinese Restaurant	Bakery	American Restaurant	Hotpot Restaurant	Cocktail Bar	Salon / Barbershop	Dess Sh
6	Central Harlem	African Restaurant	Chinese Restaurant	French Restaurant	Gym / Fitness Center	American Restaurant	Bar	Cosmet Sh
12	Upper West Side	Bakery	Café	Bar	Wine Bar	Italian Restaurant	Indian Restaurant	Pizza Pla
19	East Village	Bar	Pizza Place	Wine Bar	Mexican Restaurant	Korean Restaurant	Vietnamese Restaurant	Salc Barbersh
20	Lower East Side	Chinese Restaurant	Art Gallery	Latin American Restaurant	Bakery	Café	Coffee Shop	Japanes Restaur
22	Little Italy	Bakery	Chinese Restaurant	Cocktail Bar	Café	Italian Restaurant	Hotel	Sandw Pla
27	Gramercy	Bagel Shop	Bar	Pizza Place	Italian Restaurant	American Restaurant	Cocktail Bar	Mexic Restaur

◀ ▶

Cluster 2

In [246]:

```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 1, manhattan_merged.columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

Out[246]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Corr V
33	Midtown South	Korean Restaurant	Hotel	Hotel Bar	Gym / Fitness Center	Burger Joint	Coffee Shop	Café	Bi

◀ ▶

Cluster 3

In [247]:

```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 2, manhattan_merged.columns[[1]
+ list(range(5, manhattan_merged.shape[1]))]]
```

Out[247]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	(
0	Marble Hill	Discount Store	Sandwich Place	Coffee Shop	Gym	Tennis Stadium	Steakhouse	Do
8	Upper East Side	Italian Restaurant	Exhibit	Coffee Shop	Bakery	Gym / Fitness Center	American Restaurant	Yo
9	Yorkville	Italian Restaurant	Coffee Shop	Bar	Gym	Deli / Bodega	Sushi Restaurant	R
10	Lenox Hill	Italian Restaurant	Sushi Restaurant	Cocktail Bar	Coffee Shop	Café	Pizza Place	
13	Lincoln Square	Plaza	Gym / Fitness Center	Café	Performing Arts Venue	Theater	Italian Restaurant	Col
14	Clinton	Italian Restaurant	Theater	Gym / Fitness Center	Coffee Shop	Wine Shop	Gym	R
15	Midtown	Hotel	Coffee Shop	Clothing Store	Theater	Bookstore	Steakhouse	
16	Murray Hill	Coffee Shop	Sandwich Place	Bar	American Restaurant	Hotel	Gym / Fitness Center	R
17	Chelsea	Coffee Shop	Art Gallery	Bakery	French Restaurant	Ice Cream Shop	Wine Shop	R
18	Greenwich Village	Italian Restaurant	Clothing Store	Sushi Restaurant	Coffee Shop	Café	Gym	Dess
21	Tribeca	Italian Restaurant	Park	Wine Bar	American Restaurant	Spa	Café	Col
23	Soho	Clothing Store	Italian Restaurant	Coffee Shop	Bakery	Mediterranean Restaurant	Hotel	Ba
24	West Village	Italian Restaurant	American Restaurant	Cocktail Bar	New American Restaurant	Wine Bar	Park	C
25	Manhattan Valley	Coffee Shop	Yoga Studio	Pizza Place	Mexican Restaurant	Thai Restaurant	Indian Restaurant	
26	Morningside Heights	Coffee Shop	Park	American Restaurant	Bookstore	Burger Joint	Café	Medit R
28	Battery Park City	Coffee Shop	Hotel	Park	Clothing Store	Gym	Memorial Site	Fc
29	Financial District	Coffee Shop	Bar	Italian Restaurant	Cocktail Bar	Park	Mexican Restaurant	Gym
30	Carnegie Hill	Coffee Shop	Café	Yoga Studio	Pizza Place	Wine Shop	Cosmetics Shop	E
31	Noho	Coffee Shop	Italian Restaurant	Art Gallery	Mexican Restaurant	Hotel	Sushi Restaurant	Piz

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
32	Civic Center	Coffee Shop	Gym / Fitness Center	Spa	Hotel	Cocktail Bar	French Restaurant	Yoga Studio
34	Sutton Place	Italian Restaurant	Coffee Shop	Park	Gym / Fitness Center	Gym	Pizza Place	Fitness Studio
35	Turtle Bay	Coffee Shop	Italian Restaurant	Hotel	Seafood Restaurant	Deli / Bodega	Ramen Restaurant	Japanese Restaurant
38	Flatiron	Italian Restaurant	Japanese Restaurant	New American Restaurant	Spa	Gym / Fitness Center	American Restaurant	Colombian Restaurant
39	Hudson Yards	Gym / Fitness Center	American Restaurant	Italian Restaurant	Hotel	Burger Joint	Restaurant	Boat Bar

Cluster 4

In [248]:

```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 3, manhattan_merged.columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

Out[248]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
2	Washington Heights	Café	Bakery	Spanish Restaurant	Grocery Store	Mobile Phone Shop	Deli / Bodega	Sandwich Place
3	Inwood	Café	Mexican Restaurant	Lounge	Restaurant	Chinese Restaurant	Deli / Bodega	Wine Bar
4	Hamilton Heights	Pizza Place	Café	Coffee Shop	Mexican Restaurant	Deli / Bodega	Latin American Restaurant	Liquor Store
5	Manhattanville	Coffee Shop	Sushi Restaurant	Seafood Restaurant	Mexican Restaurant	Deli / Bodega	Italian Restaurant	Latin American Restaurant
7	East Harlem	Mexican Restaurant	Bakery	Thai Restaurant	Deli / Bodega	Latin American Restaurant	Sandwich Place	North American Restaurant
11	Roosevelt Island	Park	Gym	Dog Run	Plaza	Soccer Field	Farmers Market	Supermarket
36	Tudor City	Park	Café	Mexican Restaurant	Deli / Bodega	Greek Restaurant	Coffee Shop	Subway Restaurant

Cluster 5

In [249]:

```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 4, manhattan_merged.columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

Out[249]:

Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th M Comr Ve
37	Stuyvesant Town	Park	Bar	Boat or Ferry	Coffee Shop	Cocktail Bar	Heliport	Gym / Fitness Center

In []:

In []:

In []: