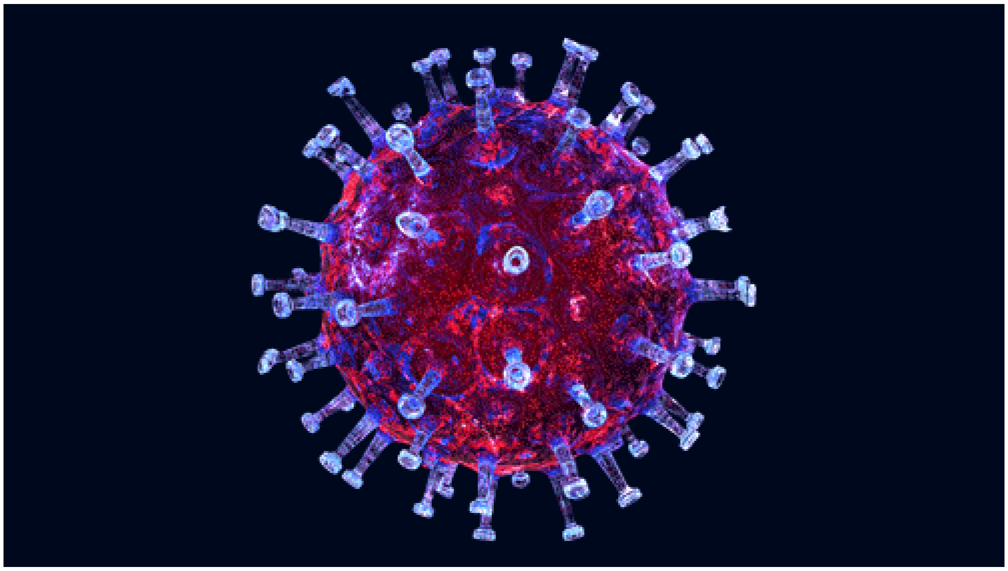
# Capstone Project : Week 2



## Covid 19 case study in New York City

**Coronaviruses are a large family of viruses which may cause illness in animals or humans. In humans, several coronaviruses are known to cause respiratory infections ranging from the common cold to more severe diseases such as Middle East Respiratory Syndrome (MERS) and Severe Acute Respiratory Syndrome (SARS). The most recently discovered coronavirus causes coronavirus disease COVID-19.COVID-19 is the infectious disease caused by the most recently discovered coronavirus. This new virus and disease were unknown before the outbreak began in Wuhan, China, in December 2019.The new york city has highest number of Covid 19 pandemic as per city consideration**

- Source (https://www.who.int/news-room/q-a-detail/q-a-coronaviruses)

```
In [13]: %matplotlib inline
         import matplotlib.pyplot as plt
```

### Import necessary Libraries

In [2]:
```
import requests # library to handle requests
import pandas as pd # library for data analsysis
import numpy as np # library to handle data in a vectorized manner
import random # library for random number generation

#!conda install -c conda-forge geopy --yes
from geopy.geocoders import Nominatim # module to convert an address into latitude and longitude values

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

# tranforming json file into a pandas dataframe library
from pandas.io.json import json_normalize

#!conda install -c conda-forge folium=0.5.0 --yes
import folium # plotting library

print('Folium installed')
print('Libraries imported.')
```

```
Folium installed
Libraries imported.
```

## Define Foursquare Credentials and Version

#### ***Make sure that you have created a Foursquare developer account and have your credentials handy***

In [3]:
```
CLIENT_ID = 'HUUNKGQOTOFMI2G4AWIYYIDAPWCLE15AEXOPNN2ZM3RE4BCB' # your Foursquare ID
CLIENT_SECRET = 'OWS5CWXLIMKN1MOTEOI3YFUEH5PNCJ3LJ5XZ1DJTBWO0OIME' # your Foursquare Secret
VERSION = '20180604'
LIMIT = 30
print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: HUUNKGQOTOFMI2G4AWIYYIDAPWCLE15AEXOPNN2ZM3RE4BCB
CLIENT_SECRET:OWS5CWXLIMKN1MOTEOI3YFUEH5PNCJ3LJ5XZ1DJTBWO0OIME
```

#### Let's again assume that you are at the NewYork-Presbyterian / Weill Cornell Medical Center. So let's start by converting the hospital address to its latitude and longitude coordinates.

In order to define an instance of the geocoder, we need to define a user_agent. We will name our agent *foursquare_agent*, as shown below.

In [4]:
```
address = "525 East 68th Street New York, NY"

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

```
40.764702099999994 -73.9540033012178
```

# 1. Search for a specific venue category

> https://api.foursquare.com/v2/venues/**search**?client_id=**CLIENT_ID**&client_secret=**CLIENT_SECRET**&ll=**LATITUDE**,**LONGITUDE**&v=**VERSION**

### Now, let's assume that it is lunch time, and you are craving Italian food. So, let's define a query to search for Italian food that is within 500 metres from the Conrad Hotel.

In [5]:
```
search_query = 'Covid19 hospital '
radius = 10000
print(search_query + ' .... OK!')
```

```
Covid19 hospital  .... OK!
```

### Define the corresponding URL

In [6]:
```
url = 'https://api.foursquare.com/v2/venues/search?client_id={}&client_secret=
{}&ll={},{}&v={}&query={}&radius={}&limit={}'.format(CLIENT_ID, CLIENT_SECRET,
latitude, longitude, VERSION, search_query, radius, LIMIT)
url
```

Out[6]:
```
'https://api.foursquare.com/v2/venues/search?client_id=HUUNKGQOTOFMI2G4AWIYYI
DAPWCLE15AEXOPNN2ZM3RE4BCB&client_secret=OWS5CWXLIMKN1MOTEOI3YFUEH5PNCJ3LJ5XZ
1DJTBWO0OIME&ll=40.764702099999994,-73.9540033012178&v=20180604&query=Covid19
hospital &radius=10000&limit=30'
```

### Send the GET Request and examine the results

```
In [7]: results = requests.get(url).json()
        results
```

```
Out[7]: {'meta': {'code': 500,
          'errorType': 'server_error',
          'errorDetail': 'Foursquare servers are experiencing problems. Please retry
        and check status.foursquare.com for updates.'},
         'response': {}}
```

### Get relevant part of JSON and transform it into a *pandas* dataframe

```
In [8]: # assign relevant part of JSON to venues
        venues = results['response']['venues']

        # tranform venues into a dataframe
        dataframe = json_normalize(venues)
        dataframe.head()
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-8-5acf500bf9ad> in <module>()
      1 # assign relevant part of JSON to venues
----> 2 venues = results['response']['venues']
      3 
      4 # tranform venues into a dataframe
      5 dataframe = json_normalize(venues)

KeyError: 'venues'
```

### Define information of interest and filter dataframe

In [47]:

```
# keep only columns that include venue name, and anything that is associated w
ith location
filtered_columns = ['name', 'categories'] + [col for col in dataframe.columns
if col.startswith('location.')] + ['id']
dataframe_filtered = dataframe.loc[:, filtered_columns]

# 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']

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

# clean column names by keeping only last term
dataframe_filtered.columns = [column.split('.')[-1] for column in dataframe_fi
ltered.columns]

dataframe_filtered
```

Out[47]:

| | name | categories | address | cc | city | country | crossStreet | distance | format |
|---|---|---|---|---|---|---|---|---|---|
| **0** | New York Presbyterian Hospital Weill Cornell M... | Hospital | 525 E 68th St Fl 6 | US | New York | United States | at York Ave | 11 | [525 E (at Yor York, N |
| **1** | Hospital for Special Surgery -- Caspary Resear... | Hospital | 541 E 71st St | US | New York | United States | NaN | 38 | [541 E New Y 10021, Sta... |
| **2** | NewYork-Presbyterian Hospital - Emergency Depa... | Emergency Room | 525 E 68th St | US | New York | United States | at York Ave. | 46 | [525 E York A York, N |
| **3** | Starr Pavilion New York Hospital urology | Medical Center | NaN | US | New York | United States | NaN | 48 | [New Y United |
| **4** | Walgreens at New York Presbyterian Hospital | Pharmacy | 525 E 68th St Ste F01-170 | US | New York | United States | NaN | 56 | [525 E F01-17 York, N |
| **5** | New York Presbyterian Hospital | Hospital | 505 E 70th St | US | New York | United States | NaN | 72 | [505 E New Y 10021, Sta... |
| **6** | Hospital for Special Surgery | Hospital | 535 E 70th St | US | New York | United States | at York Ave | 132 | [535 E York A York, N |
| **7** | Hospital for Special Surgery | Hospital | 535 E 70th St | US | New York | United States | NaN | 151 | [535 E New Y 10021, Sta... |
| **8** | Hospital for Special Surgery Interventional Ra... | Hospital | 535 E 70th St | US | New York | United States | NaN | 116 | [535 E New Y 10021, Sta... |
| **9** | New York Presbyterian Hospital Weill Cornell M... | Medical Center | 1305 York Ave | US | New York | United States | 70th St | 120 | [1305 (70th S York, N |

| | name | categories | address | cc | city | country | crossStreet | distance | format |
|---|---|---|---|---|---|---|---|---|---|
| **10** | NewYork-Presbyterian Hospital Spine Center | Hospital | NaN | US | New York | United States | NaN | 123 | [New Y United |
| **11** | Hospital for Special Surgery Sports Performanc... | Building | 519 E 72nd St | US | New York | United States | NaN | 149 | [519 E New Y 10021, Sta... |
| **12** | NewYork-Presbyterian Hospital/William Randolph... | Hospital | 525 E 68th St | US | New York | United States | York Avenue | 136 | [525 E (York A New Y 100... |
| **13** | Hospital for Special Surgery Pharmacy | Pharmacy | NaN | US | New York | United States | NaN | 149 | [New Y United |
| **14** | MSKCC Surgical Day Hospital | Hospital | NaN | US | New York | United States | NaN | 166 | [New Y 10021, States] |
| **15** | Hospital for Special Surgery - East River Prof... | Doctor's Office | 523 E 72nd St | US | New York | United States | NaN | 268 | [523 E New Y 10021, Sta... |
| **16** | Lenox Hill Hospital | Hospital | 100 E 77th St | US | New York | United States | at Lexington Ave | 1184 | [100 E Lexing New Y |
| **17** | NewYork-Presbyterian Hospital-NeuroScience ICU | Hospital Ward | 2SW | US | New York | United States | NaN | 188 | [2SW, NY 100 States] |
| **18** | The Mount Sinai Hospital | Hospital | 1468 Madison Avenue | US | New York | United States | at E 101st St | 2817 | [1468 I Avenu St), Ne |
| **19** | NewYork-Presbyterian Alexandra Cohen Hospital ... | Doctor's Office | 1283 York Avenue, 14th Floor | US | New York | United States | NaN | 188 | [1283 Y Avenu Floor, I NY 10. |
| **20** | NYC EMS Hospital 14 | Emergency Room | 70th St | US | New York | United States | York Ave | 214 | [70th S New Y United |

| | name | categories | address | cc | city | country | crossStreet | distance | format |
|---|---|---|---|---|---|---|---|---|---|
| **21** | Manhattan Eye, Ear, Nose & Throat Hospital | Hospital | 210 E 64th St | US | New York | United States | NaN | 804 | [210 E<br>New Y<br>10065,<br>Sta... |
| **22** | Hospital for Special Surgery -- Dana Center | Hospital | 510 E 73rd St | US | New York | United States | NaN | 257 | [510 E<br>New Y<br>10021,<br>Sta... |
| **23** | Bellevue Hospital Center | Hospital | 462 1st Ave | US | New York | United States | between E 26th & E 29th St. | 3373 | [462 1s<br>(betwe<br>E 29th |
| **24** | Memorial Sloan-Kettering Hospital Donor Room | Medical Center | 1250 1st Ave | US | New York | United States | NaN | 319 | [1250<br>New Y<br>10065,<br>States |
| **25** | Coler Hospital, Roosevelt Island, New York | Hospital | 900 Main St | US | New York | United States | NaN | 1105 | [900 M<br>York, N<br>United |
| **26** | Hospital For Special Surgery Ambulatory Surger... | Hospital | 1239 2nd Ave | US | New York | United States | 65th Street | 672 | [1239<br>(65th S<br>York, N |
| **27** | Hospital For Special Surgery Foot and Ankle Ce... | Doctor's Office | 523 E 72nd St | US | New York | United States | York Ave. | 261 | [523 E<br>(York A<br>York, N |
| **28** | Jake's Computer Hospital | Other Repair Shop | NaN | US | New York | United States | NaN | 290 | [New Y<br>10128,<br>States |
| **29** | Hospital for Special Surgery - Special Procedu... | Hospital Ward | 429 E 75th St | US | New York | United States | NaN | 501 | [429 E<br>New Y<br>10021,<br>Sta... |

### Let's visualize the Italian restaurants that are nearby

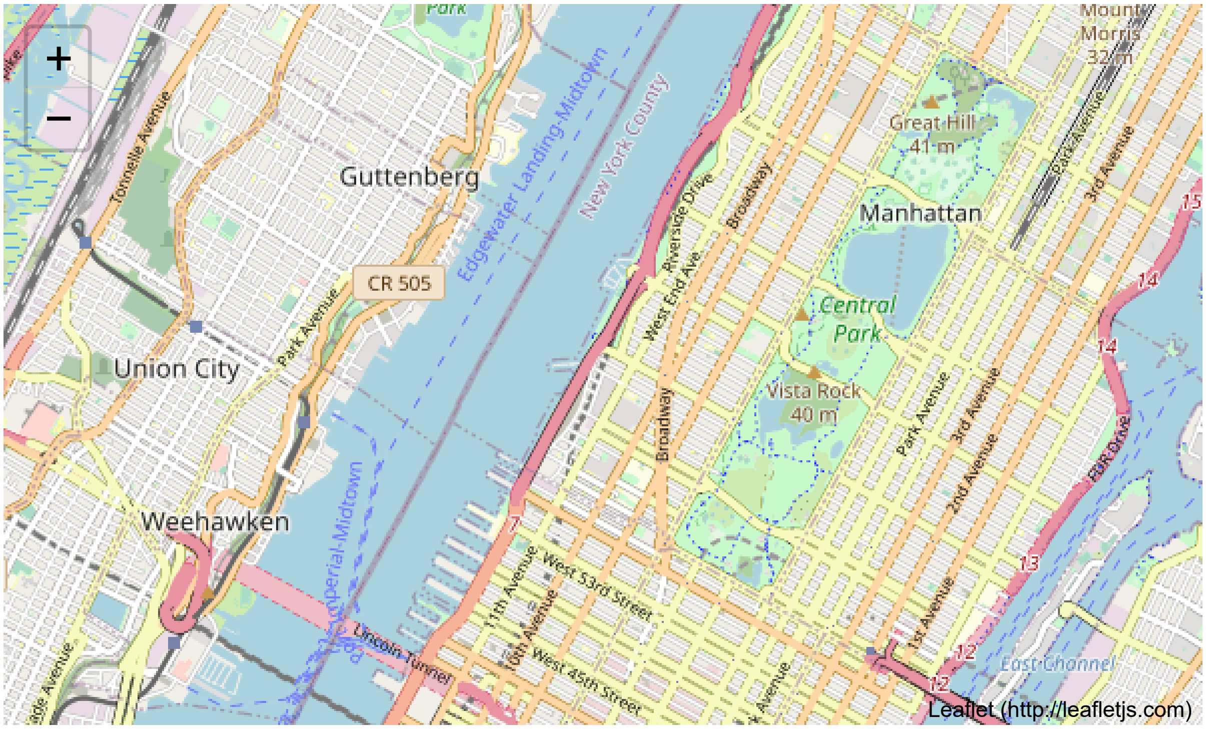
```
In [48]: dataframe_filtered.name
```

```
Out[48]: 0     New York Presbyterian Hospital Weill Cornell M...
         1     Hospital for Special Surgery -- Caspary Resear...
         2     NewYork-Presbyterian Hospital - Emergency Depa...
         3              Starr Pavilion New York Hospital urology
         4           Walgreens at New York Presbyterian Hospital
         5                        New York Presbyterian Hospital
         6                          Hospital for Special Surgery
         7                          Hospital for Special Surgery
         8     Hospital for Special Surgery Interventional Ra...
         9     New York Presbyterian Hospital Weill Cornell M...
         10           NewYork-Presbyterian Hospital Spine Center
         11    Hospital for Special Surgery Sports Performanc...
         12    NewYork-Presbyterian Hospital/William Randolph...
         13                Hospital for Special Surgery Pharmacy
         14                          MSKCC Surgical Day Hospital
         15    Hospital for Special Surgery - East River Prof...
         16                                  Lenox Hill Hospital
         17      NewYork-Presbyterian Hospital- NeuroScience ICU
         18                             The Mount Sinai Hospital
         19    NewYork-Presbyterian Alexandra Cohen Hospital ...
         20                                  NYC EMS Hospital 14
         21           Manhattan Eye, Ear, Nose & Throat Hospital
         22          Hospital for Special Surgery -- Dana Center
         23                             Bellevue Hospital Center
         24         Memorial Sloan-Kettering Hospital Donor Room
         25           Coler Hospital, Roosevelt Island, New York
         26    Hospital For Special Surgery Ambulatory Surger...
         27    Hospital For Special Surgery Foot and Ankle Ce...
         28                             Jake's Computer Hospital
         29    Hospital for Special Surgery - Special Procedu...
         Name: name, dtype: object
```

In [60]:

```
venues_map = folium.Map(location=[latitude, longitude], zoom_start=13) # generate map

venues_map
```

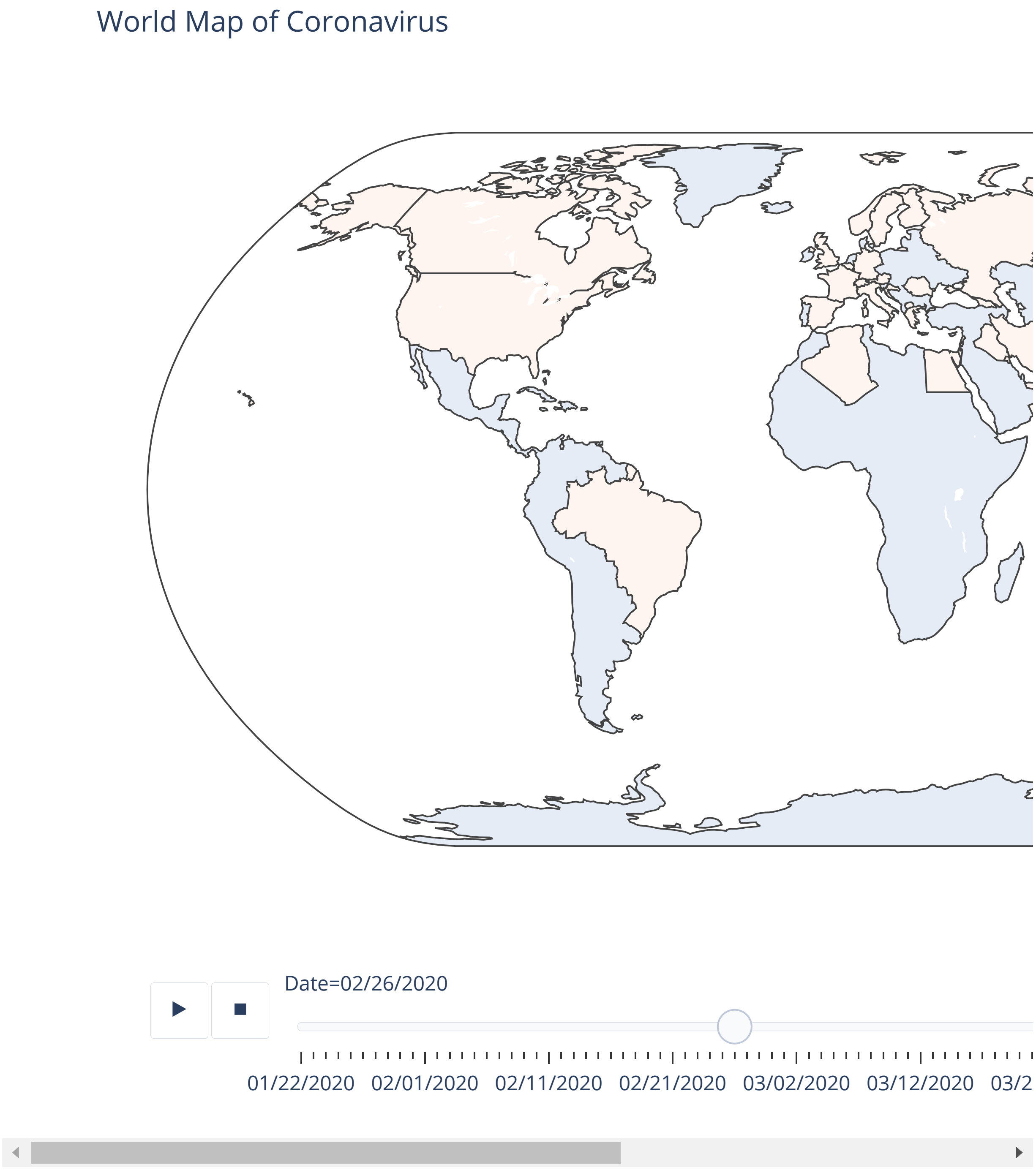
Out[60]:



# Corona Virus in the world

In [10]:

```
cases = pd.read_csv("covid_19_data.csv")
import plotly.offline as py
import plotly.express as px


py.init_notebook_mode(connected=True)

grp = cases.groupby(['ObservationDate', 'Country/Region'])['Confirmed', 'Death
s', 'Recovered'].max()
grp = grp.reset_index()
grp['Date'] = pd.to_datetime(grp['ObservationDate'])
grp['Date'] = grp['Date'].dt.strftime('%m/%d/%Y')
grp['Active'] = grp['Confirmed'] - grp['Recovered'] - grp['Deaths']
grp['Country'] =  grp['Country/Region']

fig = px.choropleth(grp, locations="Country", locationmode='country names',
                     color="Confirmed", hover_name="Country/Region",hover_data
= [grp.Recovered,grp.Deaths,grp.Active],projection="natural earth",
                     animation_frame="Date",width=1000, height=700,
                     color_continuous_scale='Reds',
                     range_color=[1000,50000],

                     title='World Map of Coronavirus')

fig.update(layout_coloraxis_showscale=True)
py.offline.iplot(fig)
```



## Read the csv file that contain covid19 cases are grouped according to their ages

In [14]:

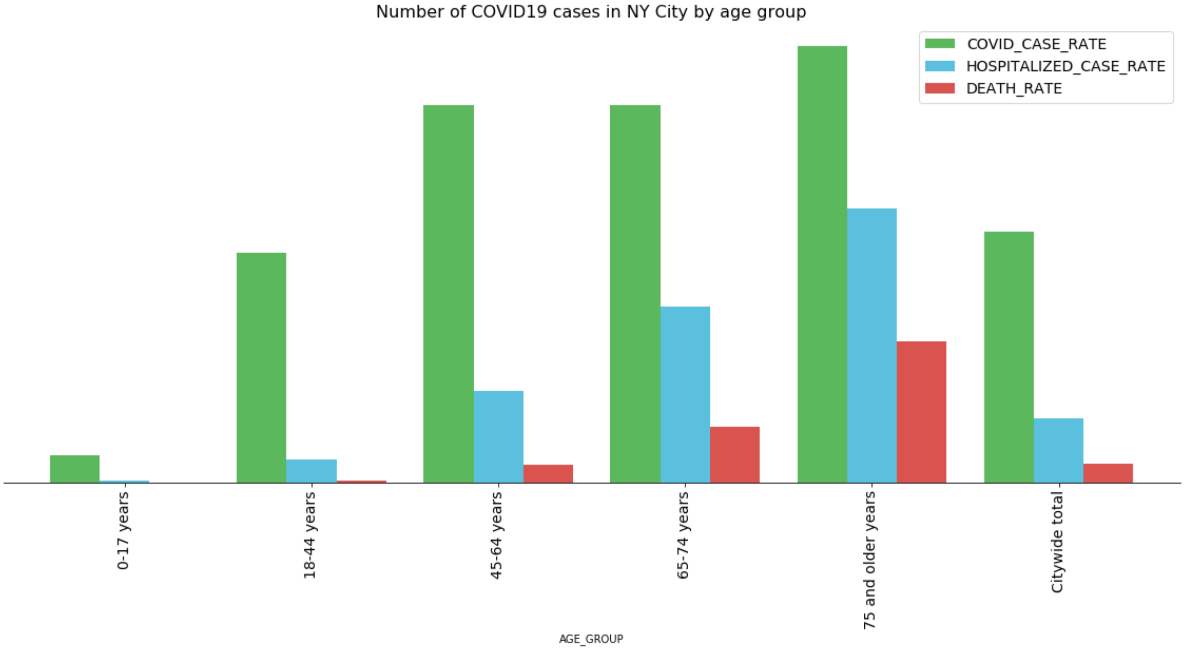
```
df=pd.read_csv(r"by-age.csv",index_col=0)
df.head()
```

Out[14]:

| | COVID_CASE_RATE | HOSPITALIZED_CASE_RATE | DEATH_RATE |
|---|---|---|---|
| **AGE_GROUP** | | | |
| **0-17 years** | 205.86 | 15.81 | 0.29 |
| **18-44 years** | 1722.48 | 169.01 | 14.30 |
| **45-64 years** | 2827.90 | 688.51 | 128.84 |
| **65-74 years** | 2824.85 | 1321.67 | 416.24 |
| **75 and older years** | 3274.19 | 2049.82 | 1055.26 |

# Visualizing Data using Matplotlib

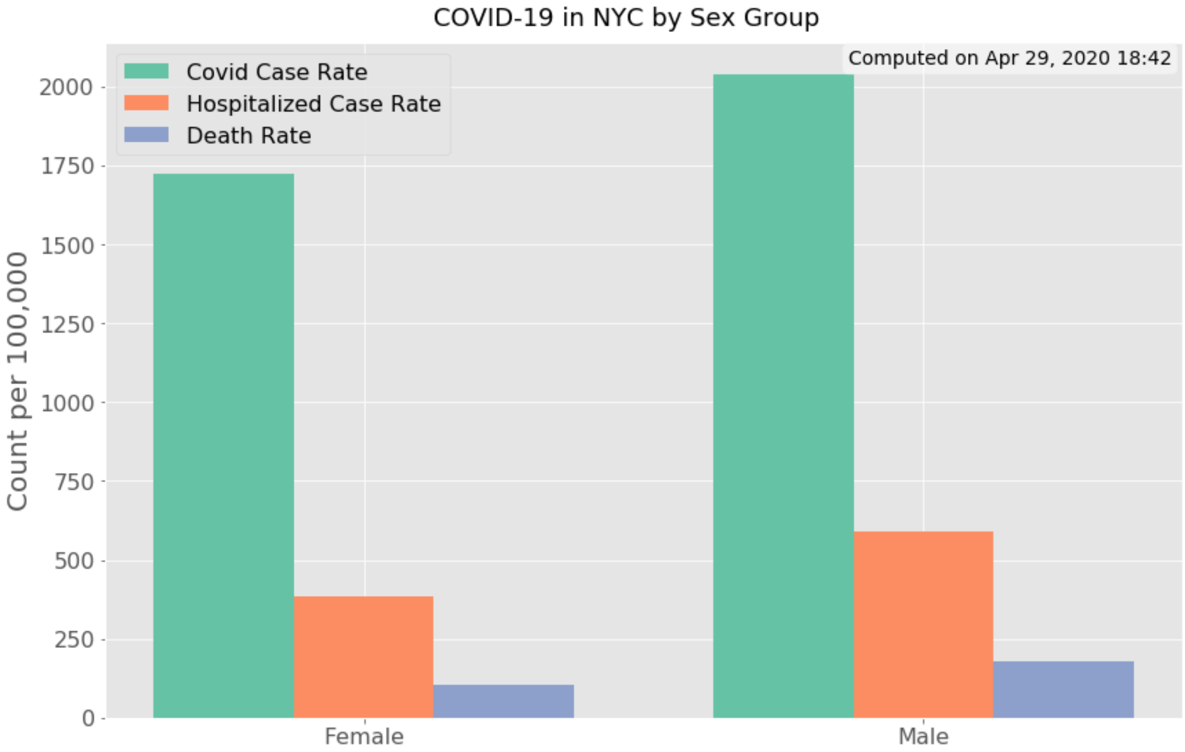
In [15]:
```
ax1 = df.plot(kind = 'bar',
                figsize = (20,8),
                width = 0.8,
                color = ('#5cb85c', '#5bc0de', '#d9534f'),
                fontsize = 14)
ax1.set_title("Number of COVID19 cases in NY City by age group",
             fontsize = 16)
ax1.legend(fontsize = 14)

#ax1.set_facecolor((1.0, 1.0, 1.0))

y_axis = ax1.axes.get_yaxis()
y_axis.set_visible(False)
ax1.set_ylabel('Count per 100,000',fontsize=20)

ax1.spines['top'].set_visible(False)

ax1.spines['right'].set_visible(False)
ax1.spines['left'].set_visible(False)
ax1.set_yticks([])
plt.show()
```



# Following data show Covid 19 cases according to sex group

In [16]:

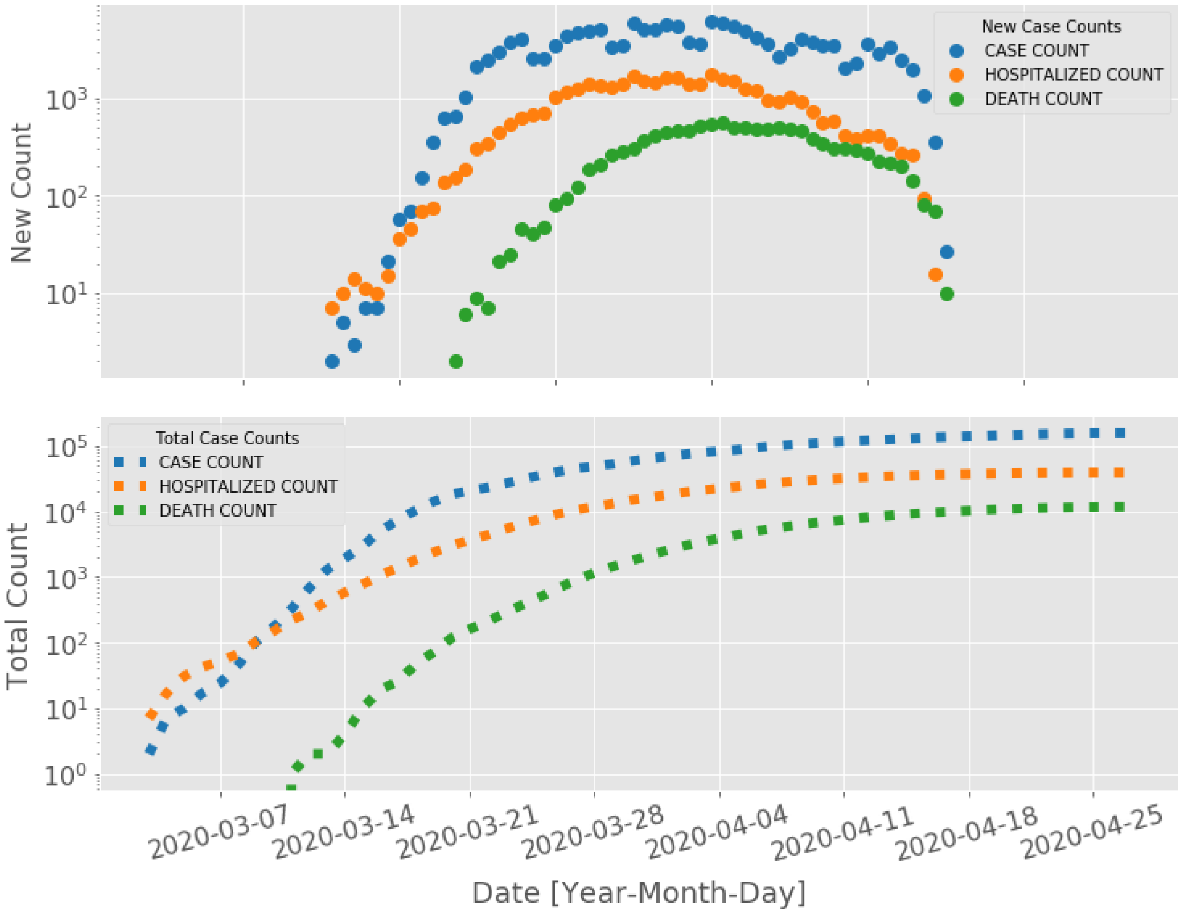
```
import requests,datetime,os

plt.style.use('ggplot') # ggplot formatting

# COVID-19 Datasets
github_url = 'https://raw.githubusercontent.com/nychealth/coronavirus-data/mas
ter/' # nyc data repository
data_file_urls = ['boro.csv','by-age.csv','by-sex.csv','case-hosp-death.csv',
                  'summary.csv','tests-by-zcta.csv'] # the .csv files to read
 where data exists

# read borough data file first and plot
r = requests.get(github_url+data_file_urls[2])
txt = r.content.decode('utf-8-sig').split('\r\n') # this vector contains all t
he data
header = txt[0].split(',')

fig,ax = plt.subplots(figsize=(14,9))
spacer = -0.25
cii = 0
for plot_indx in range(1,len(header)):
    data_to_plot,x_range = [],[]
    for jj in range(1,len(txt)-1):
        x_range.append(txt[jj].split(',')[0])
        data_to_plot.append(float(txt[jj].split(',')[plot_indx]))
    x_plot = np.arange(0,len(x_range))+spacer
    hist = ax.bar(x_plot,data_to_plot,label=header[plot_indx].replace('_',' ')
.title(),width=0.25,color=plt.cm.Set2(cii))
    spacer+=0.25
    cii+=1


ax.set_ylabel('Count per 100,000',fontsize=20)
ax.set_xticks(np.arange(0,len(x_range)))
ax.set_xticklabels(x_range)
ax.legend(fontsize=16)
ax.tick_params('both',labelsize=16)
fig.suptitle('COVID-19 in NYC by '+header[0].replace('_',' ').title(),y=0.92,f
ontsize=18)
# textbox showing the date the data was processed
txtbox = ax.text(0.0, 0.975, 'Computed on '+datetime.datetime.now().strftime(
'%b %d, %Y %H:%M'), transform=ax.transAxes, fontsize=14,
        verticalalignment='center', bbox=dict(boxstyle='round', facecolor='w',
alpha=0.5))
txtbox.set_x(1.04-(txtbox.figure.bbox.bounds[2]-(txtbox.clipbox.bounds[2]-txtb
ox.clipbox.bounds[0]))/txtbox.figure.bbox.bounds[2])
fig.savefig(header[0]+'_in_nyc.png',dpi=300,facecolor='#FCFCFC',bbox_inches =
'tight')
plt.show()
```



## Time series analysis of Covid 19 Cases

In [17]:

```
import requests,datetime,os

plt.style.use('ggplot') # ggplot formatting

# COVID-19 Datasets
github_url = 'https://raw.githubusercontent.com/nychealth/coronavirus-data/mas
ter/' # nyc data repository
data_file_urls = ['boro.csv','by-age.csv','by-sex.csv','case-hosp-death.csv',
                  'summary.csv','tests-by-zcta.csv'] # the .csv files to read
 where data exists

# read borough data file first and plot
r = requests.get(github_url+data_file_urls[3])
txt = r.content.decode('utf-8-sig').split('\r\n') # this vector contains all t
he data

header = txt[0].split(',')
dates = [datetime.datetime.strptime(ii.split(',')[0],'%m/%d/%y') for ii in txt
[1:]]

fig,axs = plt.subplots(2,1,figsize=(12,9))
cii = 0
for jj in range(1,len(txt[0].split(','))):
    vals = []
    for ii in range(0,len(txt[1:])):
        val = (txt[1:])[ii].split(',')[jj]
        if val=='':
            val = np.nan
        else:
            val = float(val)
        vals.append(val)

    axs[0].scatter(dates,vals,label=txt[0].split(',')[jj].replace('_',' '),col
or=plt.cm.tab10(cii),linewidth=3.0)
    axs[1].plot(dates,np.nancumsum(vals),label=(txt[0].split(',')[jj]).replace
('_',' ').replace('NEW','TOTAL'),
                                                                 linewidth=6.0,
color=plt.cm.tab10(cii),linestyle=':')
    cii+=1

axs[0].legend(title='New Case Counts')
axs[0].tick_params(axis='x', rotation=15)
axs[0].set_ylabel('New Count',fontsize=16)
axs[0].set_yscale('log')
axs[0].tick_params('both',labelsize=16)
axs[0].set_xticklabels([])
axs[1].legend(title='Total Case Counts')
axs[1].set_yscale('log')
axs[1].tick_params(axis='x', rotation=15)
axs[1].set_xlabel('Date [Year-Month-Day]',fontsize=18,labelpad=10)
axs[1].set_ylabel('Total Count',fontsize=18)
axs[1].tick_params('both',labelsize=16)
fig.subplots_adjust(hspace=0.1)
fig.savefig(header[0]+'_in_NYC_COVID19.png',dpi=300,facecolor='#FCFCFC')
plt.show()
```



Now, depending on when you run the above code, you might get different venues since the venues with the highest foot traffic are fetched live.