```
import numpy as np
import pandas as pd

import requests
import xml.etree.ElementTree as ET
from tqdm import tqdm

# Plotting
import plotly.express as px



# Specifying the coordinates (longitude, latitude) of origin and destination
# first parameter is longitude, second parameter is latitude

source = (-83.920699, 35.96061) # Knoxville
dest  = (-73.973846, 40.71742)  # New York City


start = "{},{}".format(source[0], source[1])
end = "{},{}".format(dest[0], dest[1])
# Service - 'route', mode of transportation - 'driving', without alternatives
url = 'http://router.project-osrm.org/route/v1/driving/{};{}?alternatives=false&annotations=nodes'.format(start, end)


headers = { 'Content-type': 'application/json'}
r = requests.get(url, headers = headers)
print("Calling API ...:", r.status_code) # Status Code 200 is success


routejson = r.json()
route_nodes = routejson['routes'][0]['legs'][0]['annotation']['nodes']
```

```
Calling API ...: 200
```

```
### keeping every third element in the node list to optimise time
route_list = []
for i in range(0, len(route_nodes)):
    if i % 3==1:
        route_list.append(route_nodes[i])

coordinates = []

for node in tqdm(route_list):
    try:
        url = 'https://api.openstreetmap.org/api/0.6/node/' + str(node)
        r = requests.get(url, headers = headers)
        myroot = ET.fromstring(r.text)
        for child in myroot:
            lat, long = child.attrib['lat'], child.attrib['lon']
        coordinates.append((lat, long))
    except:
        continue
print(coordinates[:10])
```

```
100%|██████████| 4021/4021 [32:39<00:00,  2.05it/s]
[('35.9608412', '-83.9209346'), ('35.9618154', '-83.9215042'), ('35.9626906', '-83.9220218'), ('35.9632296', '-83.9223378'),
```

```
df_out = pd.DataFrame({'Node': np.arange(len(coordinates))})
df_out['coordinates'] = coordinates
df_out[['lat', 'long']] = pd.DataFrame(df_out['coordinates'].tolist())

# Converting Latitude and Longitude into float
df_out['lat'] = df_out['lat'].astype(float)
df_out['long'] = df_out['long'].astype(float)

# Plotting the coordinates on map
color_scale = [(0, 'red'), (1,'green')]
fig = px.scatter_mapbox(df_out,
                        lat="lat",
                        lon="long",
                        zoom=8,
                        height=600,
                        width=900)
```

```
fig.update_layout(mapbox_style="open-street-map")
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
fig.show()
```