

# 1. Getting Location

- We can find the current location based on the IP address using selenium and python.

## what is selenium?

- It automates browser.Here,it's purpose is automating the process of granting permission to access location.
- It web-scrape our exact current geolocation coordinates.
- we will try to access a website like "<https://mycurrentlocation.net>" through chrome, it would ask us to allow location access.
- The command " — use-fake-ui-for-media-stream" will grant all permission for location, microphone,etc. automatically.
- GET call the webpage and wait 20secs.
- Get the XPath of the latitude and longitude elements mentioned on webpage and used selenium to find the element.

```
In [1]: from selenium import webdriver
from selenium.webdriver.chrome.options import Options
from selenium.webdriver.support.ui import WebDriverWait
import time

def getLocation():
    options = Options()
    options.add_argument("--use-fake-ui-for-media-stream")
    timeout = 20
    driver = webdriver.Chrome(executable_path = './chromedriver.exe', options=options)
    driver.get("https://mycurrentlocation.net/")
    wait = WebDriverWait(driver, timeout)
    time.sleep(3)
    longitude = driver.find_elements_by_xpath('//*[@id="longitude"]')
    longitude = [x.text for x in longitude]
    longitude = str(longitude[0])
    latitude = driver.find_elements_by_xpath('//*[@id="latitude"]')
    latitude = [x.text for x in latitude]
    latitude = str(latitude[0])
    driver.quit()
    return (latitude, longitude)

print(getLocation())

('28.68627', '77.22178')
```

# 2. Accessing Nearby Bluetooth devices

- **Bluetooth package** from python is used to get the nearby devices name.
- **discover\_devices(lookup\_names = True)** function is used for the mentioned purpose.

# 3.CreatingTrail:

- The below code will call the getLocation function after 5secs untill the no of required counts are completed using the **time.time()** function.
- Each time when the defLocation function is called discover\_devices function for scanning bluetooth device is used.
- Current location and nearby Bluetooth devices can be listed at each instance.
- The above data is saved in a csv file.

```
In [26]: import time
from datetime import date
import calendar
from bluetooth import *
import csv

next_time = time.time()
count=1
print("Trail for the day:")
fields = ['Location', 'Bluetooth devices']
mydata={}
with open('loc.csv', 'w') as csvfile:
    # creating a csv writer object
    csvwriter = csv.writer(csvfile)

    # writing the fields
    csvwriter.writerow(fields)
    csvfile.close()
while (count!=11):
    loc=getLocation()
    print(str(count)+"Your current location:"+str(loc))
    list1=[loc]
    count+=1

    nearby_devices = discover_devices(lookup_names = True)
    l=len(nearby_devices)

    name = [lis[1] for lis in nearby_devices]
    list2=[name]

    rows = zip(list1,list2)
    print ("Near by devices:"+str(name))
    with open("loc.csv", "a") as f:
        writer = csv.writer(f)
        for row in rows:
            writer.writerow(row)

    mydata[loc]=name

    next_time += 5
    time.sleep(max(0, next_time - time.time()))
```

Trail for the day:  
1.Your current location:('28.60345', '77.27146')  
Near by devices:['730', '1fb95b51']  
2.Your current location:('28.60345', '77.27146')  
Near by devices:['730', '6fed76dd', 'C1-01']  
3.Your current location:('28.60345', '77.27146')  
Near by devices:['730', '1fb95b51']  
4.Your current location:('28.60345', '77.27146')  
Near by devices:['730', 'C1-01']  
5.Your current location:('28.60345', '77.27146')  
Near by devices:['730', 'C1-01']  
6.Your current location:('', '')  
Near by devices:['', '730', 'C1-01', '6fed76dd']  
7.Your current location:('28.60345', '77.27146')  
Near by devices:['730', 'C1-01', '6fed76dd']  
8.Your current location:('28.60345', '77.27146')  
Near by devices:['730', '6fed76dd', 'C1-01']  
9.Your current location:('28.60345', '77.27146')  
Near by devices:['730', '6fed76dd', '1fb95b51']  
10.Your current location:('28.60345', '77.27146')  
Near by devices:['730', '6fed76dd']  
11.Your current location:('28.60345', '77.27146')

```
In [1]: # display csv file image
from IPython.display import Image
Image(filename='csv.png')
```

Out[1]: 

```
1 Location,Bluetooth devices
2
3 "('28.60345', '77.27146'),"['730', '1fb95b51']"
4
5 "('28.60345', '77.27146'),"['730', '6fed76dd', 'C1-01']"
6
7 "('28.60345', '77.27146'),"['730', '1fb95b51']"
8
9 "('28.60345', '77.27146'),"['730', 'C1-01']"
10
11 "('28.60345', '77.27146'),"['730', 'C1-01']"
12
13 "('28.60345', '77.27146'),"['730', 'C1-01', '6fed76dd']"
14
15 "('28.60345', '77.27146'),"['730', '6fed76dd', 'C1-01']"
16
17 "('28.60345', '77.27146'),"['730', '6fed76dd', '1fb95b51']"
18
19 "('28.60345', '77.27146'),"['730', '6fed76dd']"
20
21 "('28.60341', '77.27146'),"['730', '6fed76dd', '1fb95b51']"
22
23 "('28.603451', '77.27146'),"['730', 'C1-01', '6fed76dd']"
24
25 "('28.61737', '77.27499'),"['730']"
26
27 "('28.61737', '77.27499'),"['730']"
```

```
In [9]: #reading the saved data as a dataframe
import pandas as pd

df=pd.read_csv('loc.csv',error_bad_lines=False)
df.head(20)
```

Out[9]:

	Location	Bluetooth devices
0	('28.60345', '77.27146')	['730', '1fb95b51']
1	('28.60345', '77.27146')	['730', '6fed76dd', 'C1-01']
2	('28.60345', '77.27146')	['730', '1fb95b51']
3	('28.60345', '77.27146')	['730', 'C1-01']
4	('28.60345', '77.27146')	['730', 'C1-01']
5	('28.60345', '77.27146')	['730', 'C1-01', '6fed76dd']
6	('28.60345', '77.27146')	['730', '6fed76dd', 'C1-01']
7	('28.60345', '77.27146')	['730', '6fed76dd', '1fb95b51']
8	('28.60345', '77.27146')	['730', '6fed76dd']
9	('28.60341', '77.27146')	['730', '6fed76dd', '1fb95b51']
10	('28.603451', '77.27146')	['730', 'C1-01', '6fed76dd']
11	('28.61737', '77.27499')	['730']
12	('28.61737', '77.27499')	['730']
13	('28.60345', '77.27146')	['730', '1fb95b51']
14	('28.60345', '77.27146')	['730', '6fed76dd', '1fb95b51']
15	('28.60345', '77.27146')	['730', 'C1-01', '6fed76dd']
16	('28.61737', '77.27499')	['730']
17	('28.61737', '77.27499')	['730']
18	('28.60345', '77.27146')	['730', '1fb95b51']
19	('28.60345', '77.27146')	['730', '6fed76dd']

# 4. Graph creation:

- For creating graph Networkx package is used.
- The are many Locations that are repeated in the saved data.
- Using the **dataframe groupby** a dictionary is created in which all the nearby devices are listed for distinct location.
- Distinct Locations are keys and different bluetooth devices are the values in dictionary.
- In graph,a directed edge is created from **location to the grouped devices** traced at that location at different instances.

```
In [17]: #grouping all traced devices for distinct location
#creating a dictionary
d=df.groupby('Location')['Bluetooth devices'].apply(list).to_dict()
print(d)

{'('28.60341', '77.27146)': [ '['730', '6fed76dd', '1fb95b51']" ], '('28.60345', '77.27146)': [ '['730', '1fb95b51']" , '['730', '6fed76dd', 'C1-01']" , '['730', '1fb95b51']" , '['730', 'C1-01']" , '['730', 'C1-01']" , '['730', '6fed76dd']" , '['730', '6fed76dd', 'C1-01']" , '['730', '6fed76dd', '1fb95b51']" , '['730', 'C1-01', '6fed76dd']" , '['730', '6fed76dd', '1fb95b51']" , '['730', 'C1-01', '6fed76dd']" , '['730', 'C1-01', '6fed76dd', '1fb95b51']" ], '('28.603451', '77.27146)': [ '['730', 'C1-01', '6fed76dd']" ], '('28.61737', '77.27499)': [ '['730']" , '['730']" , '['730']" ] }
```

```
In [23]: import networkx as nx
import matplotlib.pyplot as plt
g = nx.DiGraph(d)
nx.draw(g,with_labels=True,node_color='#00ff00')
plt.draw()
```

