In [1]:
```
from telethon.sync import TelegramClient
from telethon.tl.functions.messages import GetDialogsRequest
from telethon.tl.types import InputPeerEmpty, InputPeerChannel, InputPeerUser
from telethon.errors.rpcerrorlist import PeerFloodError, UserPrivacyRestrictedError
from telethon.tl.functions.channels import InviteToChannelRequest
import sys
import csv
import traceback
import time
import random
import datetime
import time
import asyncio
import telethon
from telethon.sessions import StringSession
from telethon.sync import TelegramClient, events
import nest_asyncio as nta
from os import system
import random
import os
import pandas as pd
```

In [2]:
```
apis = list()
```

In [3]:
```
client1 = TelegramClient('+923054129775', 12234459, '55e3da6f149bf094b5949b458eb4483c')
await client1.connect()
```

In [4]:
```
#Qasim
api_id1 = 12234459
api_hash1 = '55e3da6f149bf094b5949b458eb4483c'
phone1 = '+923054129775'
string_session=''
async with TelegramClient(StringSession(), api_id1, api_hash1) as client1:
    string_session=client1.session.save()
    apis.append([api_id1,api_hash1,phone1,string_session])
```

```
Please enter your phone (or bot token): +923054129775
Please enter the code you received: 41814
Signed in successfully as Sheikh Muhammad Qasim
```

In [5]:
```
#MyJazzNumber
api_id2 = 11193976
api_hash2 = 'eb4e970ebaab983e9028fd0eb976cf65'
phone2 = '+923051193900'
string_session=''
async with TelegramClient(StringSession(), api_id2, api_hash2) as client2:
    string_session=client2.session.save()
    apis.append([api_id2,api_hash2,phone2,string_session])
```

```
Please enter your phone (or bot token): +923051193900
Please enter the code you received: 27325
Signed in successfully as Qasim Sheikh
```

In [6]:

```
#UfoneNumberFadiBhai
api_id3 = 17185732
api_hash3 = 'c04784edd8365d21a45e4790d3a6a8f2'
phone3 = '+923354887361'
string_session=''
async with TelegramClient(StringSession(), api_id3, api_hash3) as client3:
    string_session=client3.session.save()
    apis.append([api_id3,api_hash3,phone3,string_session])
```

```
Please enter your phone (or bot token): +923354887361
Please enter the code you received: 25357
Signed in successfully as Degen Trade Helper
```

In [8]:

```
#Nanna
api_id4 = 12482314
api_hash4 = '5ce14b20db65a8b5ab6b9df22e54f21e'
phone4 = '+923334454010'
string_session=''
async with TelegramClient(StringSession(), api_id4, api_hash4) as client4:
    string_session=client4.session.save()
    apis.append([api_id4,api_hash4,phone4,string_session])
```

```
Please enter your phone (or bot token): +923334454010
Please enter the code you received: 63503
Signed in successfully as Afifa Shaikh
```

In [9]:

```
#Mom
api_id5 = 11568994
api_hash5 = '70b00b5f0ac8fb32fa9c9076be7d5ced'
phone5 = '+923244727765'
string_session=''
async with TelegramClient(StringSession(), api_id5, api_hash5) as client5:
    string_session=client5.session.save()
    apis.append([api_id5,api_hash5,phone5,string_session])
```

```
Please enter your phone (or bot token): +923244727765
Please enter the code you received: 21031
Signed in successfully as Tamseela Dawood
```

In [10]:

```
#Rehan Malik 1
api_id6 = 13488635
api_hash6 = 'c7de1f05de911f86f20a4bc4ed0b1fd9'
phone6 = '+923248443576'
string_session=''
async with TelegramClient(StringSession(), api_id6, api_hash6) as client6:
    string_session=client6.session.save()
    apis.append([api_id6,api_hash6,phone6,string_session])
```

```
Please enter your phone (or bot token): +923248443576
Please enter the code you received: 81403
Signed in successfully as Rehan Malik
```

In [11]:

```
#Rehan Malik 2
api_id7 = 11222160
api_hash7 = 'e7cea589feeebe381f6d1edacdeddc7a'
phone7 = '+923064604589'
string_session=''
async with TelegramClient(StringSession(), api_id7, api_hash7) as client7:
```

```
    string_session=client7.session.save()
    apis.append([api_id7,api_hash7,phone7,string_session])
```

```
Please enter your phone (or bot token): +923064604589
Please enter the code you received: 91001
Signed in successfully as Rehan Malik
```

In [12]:

```
##Saad Mir
api_id8 = 18941092
api_hash8 = 'd0a18fa6f72bf5ce724765ff2efaf8d3'
phone8 = '+923214778907'
string_session=''
async with TelegramClient(StringSession(), api_id8, api_hash8) as client8:
    string_session=client8.session.save()
    apis.append([api_id8,api_hash8,phone8,string_session])
```

```
Please enter your phone (or bot token): +923214778907
Please enter the code you received: 54527
Please enter your password: ........
Signed in successfully as Saad Mir
```

In [13]:

```
#Saad Mir 2
api_id9 = 19353011
api_hash9 = 'f4317b278f57d9ade86beeb04a756fc2'
phone9 = '+923241445069'
string_session=''
async with TelegramClient(StringSession(), api_id9, api_hash9) as client9:
    string_session=client9.session.save()
    apis.append([api_id9,api_hash9,phone9,string_session])
```

```
Please enter your phone (or bot token): +923241445069
Please enter the code you received: 75379
Signed in successfully as Hunter Mind
```

In [14]:

```
#Dad
api_id10 = 9698509
api_hash10 = '1b87426580520a3b21073b52ba91c916'
phone10 = '+923004279085'
string_session=''
async with TelegramClient(StringSession(), api_id10, api_hash10) as client10:
    string_session=client10.session.save()
    apis.append([api_id10,api_hash10,phone10,string_session])
```

```
Please enter your phone (or bot token): +923004279085
Please enter the code you received: 63098
Signed in successfully as Dawood Sheikh
```

In [15]:

```
#Bhabi
api_id11 = 12936152
api_hash11 = 'fdb9f06dfa4f3e1890bcef78455bf8b0'
phone11 = '+923215154360'
string_session=''
async with TelegramClient(StringSession(), api_id11, api_hash11) as client11:
    string_session=client11.session.save()
    apis.append([api_id11,api_hash11,phone11,string_session])
```

```
Please enter your phone (or bot token): +923215154360
Please enter the code you received: 27317
```

```
Signed in successfully as Shumaila Shumaila
```

In [123…

```
#ZongNumber
api_id11 = 11844702
api_hash11 = '5d524bf56c9c53476d2a56de283542f1'
phone11 = '+923160495794'
string_session=''
async with TelegramClient(StringSession(), api_id11, api_hash11) as client11:
    string_session=client11.session.save()
    apis.append([api_id11,api_hash11,phone11,string_session])
```

```
Please enter your phone (or bot token): +923160495794
Please enter the code you received: 35519
Signed in successfully as Degen Trader
```

In [121…

```
#Junaid
api_id13 = 10026132
api_hash13 = '01c503ac975d16c512a989897831c7d0'
phone13 = '+923154195171'
string_session=''
async with TelegramClient(StringSession(), api_id13, api_hash13) as client13:
    string_session=client13.session.save()
    apis.append([api_id13,api_hash13,phone13,string_session])
```

```
Please enter your phone (or bot token): +923154195171
Please enter the code you received: 63572
Signed in successfully as Junaid Ejaz
```

In [19]:

```
#FadiBhaiKSA
api_id14 = 14492955
api_hash14 = '4bc85d9b3f2431fcbb3026b0fbb56ff4'
phone14 = '+966531009131'
string_session=''
async with TelegramClient(StringSession(), api_id14, api_hash14) as client14:
    string_session=client14.session.save()
    apis.append([api_id14,api_hash14,phone14,string_session])
```

```
Please enter your phone (or bot token): +966531009131
Please enter the code you received: 30877
Signed in successfully as No name
```

In [20]:

```
#Akhnus
api_id15 = 8334330
api_hash15 = '8c659b1ba125cca04156f1bc31449051'
phone15 = '+923234605838'
string_session=''
async with TelegramClient(StringSession(), api_id15, api_hash15) as client15:
    string_session=client15.session.save()
    apis.append([api_id15,api_hash15,phone15,string_session])
```

```
Please enter your phone (or bot token): +923234605838
Please enter the code you received: 26075
Signed in successfully as Muhammad Aghnus Jamil
```

In [157…

```
a = list()
```

In [158…

```
file1 = open('DegenProTraders2.txt', 'w')
for i in range(len(apis)):
    for j in range(4):
        file1.writelines(str(apis[i][j]))
        file1.writelines('\n')
file1.close()
```

In [159…

```
df = pd.DataFrame(columns = ['Api_Id','Api_Hash', 'Number','Session_String'])
df
```

Out[159…

Api_Id	Api_Hash	Number	Session_String

In [160…

```
with open("DegenProTraders2.txt", "r") as f:
    while True:
        file_eof = f.readline()
        a.append(file_eof)
        if file_eof == '':
            break
```

In [161…

```
i = 0
for j in a:
    try:
        df.loc[-1] = [int(a[i].strip()),a[i+1].strip(),a[i+2].strip(),a[i+3].strip()]
        df.index = df.index + 1  # shifting index
        df = df.sort_index()  # sorting by index
        i = i+4
    except:
        break
```

In [162…

```
df.tail(5)
```

Out[162…

	Api_Id	Api_Hash	Number	Se
10	11568994	70b00b5f0ac8fb32fa9c9076be7d5ced	+923244727765	1BJWap1sBu7qO3K4sQuKj7tg 8
11	12482314	5ce14b20db65a8b5ab6b9df22e54f21e	+923334454010	1BJWap1sBu0o_ImjpHAiooEI4TJ8oSneWaj(
12	17185732	c04784edd8365d21a45e4790d3a6a8f2	+923354887361	1BJWap1sBuyBWUuD8SMUrrr PoH
13	11193976	eb4e970ebaab983e9028fd0eb976cf65	+923051193900	1BJWap1sBu0pUxytIYNl_ZLjc3-L-fBfRJLx
14	12234459	55e3da6f149bf094b5949b458eb4483c	+923054129775	1 c0XIEgWS6bwXMvRNzd6vr9TA

In [ ]:

```
chats = []
last_date = None
chunk_size = 200
groups=[]

result = await client(GetDialogsRequest(
```

```
            offset_date=last_date,
            offset_id=0,
            offset_peer=InputPeerEmpty(),
            limit=chunk_size,
            hash = 0
        ))
```

In [ ]:
```
chats.extend(result.chats)
```

In [ ]:
```
for chat in chats:
    try:
        if chat.megagroup== True:
            groups.append(chat)
    except:
        continue
```

In [ ]:
```
print('Choose a group to scrape members from:')
i=0
for g in groups:
    print(str(i) + '- ' + g.title)
    i+=1
```

In [ ]:
```
g_index = input("Enter a Number: ")
target_group=groups[int(g_index)]

print('Fetching Members...')
all_participants = []
all_participants = client.get_participants(target_group, aggressive=True)
```

In [ ]:
```
print('Saving In file...')
with open("members.csv","w",encoding='UTF-8') as f:
    writer = csv.writer(f,delimiter=",",lineterminator="\n")
    writer.writerow(['username','user id', 'access hash','name','group', 'group id'])
    partic = await all_participants
    for user in partic:
        if user.username:
            username= user.username
        else:
            username= ""
        if user.first_name:
            first_name= user.first_name
        else:
            first_name= ""
        if user.last_name:
            last_name= user.last_name
        else:
            last_name= ""
        name= (first_name + ' ' + last_name).strip()
        writer.writerow([username,user.id,user.access_hash,name,target_group.title, tar
print('Members scraped successfully.')
```

In [ ]:
```
input_file = 'members.csv'
```

```
users = []
with open(input_file, encoding='UTF-8') as f:
    rows = csv.reader(f,delimiter=",",lineterminator="\n")
    next(rows, None)
    for row in rows:
        user = {}
        user['username'] = row[0]
        user['id'] = int(row[1])
        user['access_hash'] = int(row[2])
        user['name'] = row[3]
        users.append(user)
```

In [ ]:

```
channel = 'https://t.me/membersadding007'
```

In [ ]:

```
nuser
```

In [ ]:

```
n=0
mode=2
for user in users:
    n += 1
    if n % 50 == 0:

        try:
            print ("Adding {}".format(user['id']))
            if mode == 1:
                if user['username'] == "":
                    continue
                user_to_add = client.get_input_entity(user['username'])
            elif mode == 2:
                user_to_add = InputPeerUser(user['id'], user['access_hash'])
            else:
                sys.exit("Invalid Mode Selected. Please Try Again.")
            await client(InviteToChannelRequest(channel,[user_to_add]))
            print("Waiting for 60-180 Seconds...")
            time.sleep(random.randrange(60, 180))
        except PeerFloodError:
            print("Getting Flood Error from telegram. Script is stopping now. Please tr
        except UserPrivacyRestrictedError:
            print("The user's privacy settings do not allow you to do this. Skipping.")
        except:
            traceback.print_exc()
            print("Unexpected Error")
            continue
        time.sleep(900)
```

In [ ]:

```
users
```

In [ ]:

```

```

In [ ]:

```
await client(InviteToChannelRequest(channel,['ForwardOptimism']))
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:
```
from telethon.sync import TelegramClient
from telethon.tl.functions.messages import GetDialogsRequest
from telethon.tl.types import InputPeerEmpty, InputPeerChannel, InputPeerUser
from telethon.errors.rpcerrorlist import PeerFloodError, UserPrivacyRestrictedError
from telethon.tl.functions.channels import InviteToChannelRequest
from telethon.sessions import StringSession

import sys
import csv
import traceback
import time
import random
```

In [ ]:
```
id = 12234459
hash = '55e3da6f149bf094b5949b458eb4483c'
sessionstring="1AZWarzUBu3c6-VSh4eV_poctwSjHoJfPbc4h4el6XYx-01Lk697ZD6jBZgJkvNQvPmWSeG-
chatid = 'https://t.me/membersadding007'
```

In [ ]:
```
input_file = 'members.csv'
users = []
with open(input_file, encoding='UTF-8') as f:
    rows = csv.reader(f,delimiter=",",lineterminator="\n")
    next(rows, None)
    for row in rows:
        user = {}
        user['username'] = row[0]
        user['id'] = int(row[1])
        user['access_hash'] = int(row[2])
        user['name'] = row[3]
        users.append(user)
```

In [ ]:
```
nuser=[]
for user in users:
    if user['username']!='':
        nuser.append(user['username'])
```

In [ ]:
```
def load_data():
    users = open('users.txt', 'r')
```

```
    Users = users.readlines()
    userdata=[]
    count=0
    for user in Users:
        if user[-1]=='\n':
            user=user[:-1]

        if count%2==0:
            pass
        else:
            userdata.append(user)
        count=count+1
    return userdata
```

In [ ]:
```
userssessions=load_data()
```

In [ ]:
```
len(userssessions)
```

In [ ]:
```
async def add_user_to_chat(id,hash,sessionstring,chatid,uid):
    async with TelegramClient(StringSession(sessionstring), id, hash) as client:
        await client(InviteToChannelRequest(chatid,[uid]))
```

In [ ]:
```

```

In [ ]:
```

```

In [ ]:
```
userssessions.pop(0)
userssessions.pop(1)
```

In [ ]:
```
len(userssessions)
```

In [ ]:
```
len(nuser)
```

In [ ]:
```
count=0
for sessionstring in userssessions:
    while count<len(nuser):
        try:
            await add_user_to_chat(id,hash,sessionstring,chatid,nuser[count])
            nuser.pop(count)
            count=count+1
            time.sleep(random.randint(1, 10))
            print("User Added")
        except PeerFloodError:
            print('Peer Flood Error')
            time.sleep(random.randint(1, 10))
            break
        except Exception as ex:
            print(ex)
```

```
            nuser.pop(count)
            count=count+1
            print(len(nuser))
```

In [ ]:
```
count
```

In [ ]:
```

```

In [ ]:
```

```

In [ ]:
```

```

In [ ]:
```
import datetime
import time
import asyncio
import telethon
from telethon.sessions import StringSession
from telethon.sync import TelegramClient, events
import nest_asyncio as nta
from os import system
import random

import os


class TelegramBot():
    def __init__(self):
        self.id=8055482
        self.hash='2f92ec1401f0a9d6ba10a7ccadcca439'
        self.usernames=[]
        self.users=[]

    async def create_new_user(self):
        try:
            if len(self.usernames)<1:
                self.load_data()
            while(1):
                username=input("Enter Unique Username :")
                if username in self.usernames:
                    print("This user already exists")
                else:
                    break

            string_session=''
            async with TelegramClient(StringSession(), self.id, self.hash) as client:
                string_session=client.session.save()

            datacreation=[]
            for i in range(len(self.users)):
                datacreation.append(self.usernames[i])
                datacreation.append(self.users[i])
```

```
            saveusers=datacreation+[username,string_session]
            for x in range(len(saveusers)):
                saveusers[x]=saveusers[x]+'\n'

            file1 = open('users.txt', 'w')
            file1.writelines(saveusers)
            file1.close()
            self.usernames.append(username)
            self.users.append(string_session)
            print("New User Added")

        except:
            print("Exception Occurred while creating new user")


    def index(self,username):
        for i in range(len(self.usernames)):
            if self.usernames[i]==username:
                return i

    def delete_user(self):
        if len(self.usernames)<1:
            self.load_data()

        username=input("Enter Username of user you want to delete ")
        if username in self.usernames:
            user_index=self.index(username)
            self.usernames.pop(user_index)
            self.users.pop(user_index)

            datacreation=[]
            for i in range(len(self.users)):
                datacreation.append(self.usernames[i])
                datacreation.append(self.users[i])

            saveusers=datacreation
            for x in range(len(saveusers)):
                saveusers[x]=saveusers[x]+'\n'
            file1 = open('users.txt', 'w')
            file1.writelines(saveusers)
            file1.close()
        else:
            print("User Do not Exists")


    async def add_user_to_chat(self,id,hash,sessionstring,chatid):
        async with TelegramClient(StringSession(sessionstring), id, hash) as client:
            await client.send_message(chatid,msg,parse_mode='html')


    def load_data(self):
        users = open('users.txt', 'r')
        Users = users.readlines()
        count=0
        for user in Users:
            if user[-1]=='\n':
                user=user[:-1]

            if count%2==0:
                self.usernames.append(user)
```

```
            else:
                self.users.append(user)
            count=count+1


    async def RunBot(self):
        while True:
            print("Enter 1 for Bulk Addition")
            print("Enter 2 to Enter a New Super User")
            print("Enter 3 to Delete an existing super user")
            print("Enter 4 to Exit")
            option=0
            while True:
                try:
                    option=int(input("Enter your choice "))
                    if (option==1) | (option==2) | (option==3) | (option==4):
                        break
                    else:
                        print("Select from above 4 options ")
                except Exception as ex:
                    print(ex)
            if option==1:
                pass
            elif option==2:
                await self.create_new_user()
            elif option==3:
                self.delete_user()
            elif option ==4:
                break
    async def Run(self):
        await self.RunBot()

TB=TelegramBot()
await TB.Run()
```

In [ ]: