

بسمه تعالی

پروژه مهندسی اینترنت

برنامه نویسی سوکت

زبان پایتون

احمد صالحی

۹۷۱۷۰۲۳۱۶۸

برنامه به دو قسمت سرور و کلاینت تقسیم شده که گد هر کدام را بررسی میکنیم :

گد سرور :

برنامه از یک کلاس سرور تشکیل شده که شامل توابع مختلفی است :

```
def start_server(self): # شروع سرور
    self.s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

    host = socket.gethostname(socket.gethostname())
    port = 5050
    self.clients = []

    self.s.bind((host, port)) # بایند سوکت به آی پی و پورت
    self.s.listen(100)

    print('Running on host: '+str(host)) # نمایش هاست و پورت
    print('Running on port: '+str(port))

    self.username_lookup = {}

    while True:
        c, addr = self.s.accept()

        username = c.recv(1024).decode()
        if(username != ""): # بررسی خالی نبود یا لاگین نبودن
            print('New connection. Username: '+str(username))
            self.broadcast('New person joined the room. Username: '+username)
            self.username_lookup[c] = username
            self.clients.append(c)
            threading.Thread(target=self.handle_client, args=(c, addr,)).start()
```

تابع start_server که سرور را با استفاده از سوکت مورد نظر اجرا می کند و سوکت را به هاست و پورت مورد نظر بایند می کند و آن را نمایش می دهد

سپس از سوکت و سرور ، یوزر نیم کاربر را که در سمت کلاینت وارد می شود دریافت کرده و آن را دیکود می کند ، در صورتی که یوزرنیم خالی نباشد (یعنی با موفقیت لاگین کرده باشیم) در سرور و کلاینت نمایش می دهد که یوزری با یوزرنیم خود وارد چت روم شده و در سرور آن را broadcast می کند و آن را به لیست client ها اضافه می کند (append) و سپس تدری را میسازد و تابع handle_client را به آن می دهد که در ادامه کد آن را میبینیم :

```
def handle_client(self,c,addr): # هندل کردن یوزر ها در سرور
    while True:
        try:
            msg = c.recv(1024) # دریافت مسیج ها
        except:
            c.shutdown(socket.SHUT_RDWR)
            self.clients.remove(c)
            print(str(self.username_lookup[c])+' left the room.')
            self.broadcast(str(self.username_lookup[c])+' has left the room.')
            break

        if msg.decode() != '': # بررسی خالی نبودن پیام
            print('New message: '+str(msg.decode())) # پرینت پیام جدید
            for connection in self.clients:
                if connection != c:
                    connection.send(msg)
```

در این تابع ، مسیج ها از سمت کلاینت ها دریافت می شود و در صورتی که مسیج آن خالی نباشد آن را ارسال می کند ، اگر اکسپشنی هم رخ دهد ، کانکشن سرور را قطع کرده و آن کلاینت را از لیست خارج می کند و با استفاده از تابع broadcast به همه نمایش می دهد که یوزرنیم آن شخص از چت روم خارج شده.

در آخر به تابع broadcast می رسیم :

```
def broadcast(self,msg): # ارسال پیام ها
    for connection in self.clients:
        connection.send(msg.encode())
```

این تابع در کانکشن های کلاینت ها میچرخد و مسیج های دریافت شده آن ها را به صورت انکود شده ، نمایش می دهد.

کد کلاینت :

```
def create_connection(self): # ساختن کانکشن
    self.s = socket.socket(socket.AF_INET,socket.SOCK_STREAM) # ساختن سوکت

    while 1:
        try: # کانکت شدن به کانکشن سوکت
            host = '192.168.42.69'
            port = 5050
            self.s.connect((host,port))
            break
        except:
            print("Couldn't connect to server")
```

در تابع اصلی که کانکشن ها را می سازد ، ابتدا یک سوکت می سازیم و با استفاده از هاست (آی پی) و پورت مورد نظر به آن متصل می شویم و در صورتی که موفق نشویم که متصل شویم پیام Couldn't Connect to Server را نمایش می دهیم.

```
read_db() # خواندن از پایگاه داده
print("if you want to login type 1")
print("if you want to sign up type 2")
menu = input()
if(menu == '1'): # ورود به برنامه
    self.username = input('Enter username --> ')
    self.password = input('Enter pass --> ')
    if(check_login(self.username,self.password) == True):
        self.s.send(self.username.encode())

        message_handler = threading.Thread(target=self.handle_messages,args=())
        message_handler.start()

        input_handler = threading.Thread(target=self.input_handler,args=())
        input_handler.start()
    else:
        print("Username or Password Incorrect")
```

در ادامه به تابع `read_db` می‌رسیم که در گد آن را در قسمت‌های بعد می‌بینیم

در شروع برنامه از کاربر می‌خواهیم که آیا می‌خواهد لاگین کند یا به عنوان یک یوزر جدید ثبت نام کند

در صورت لاگین ، یوزرنیم و پسورد کاربر به تابع `check_login` داده می‌شود و آن را بررسی می‌کند که در فایل‌های ما چنین یوزر و پسوردهای وجود دارد یا خیر

سپس دو ترد برای نمایش چت‌ها و مسیج‌ها ساخته می‌شود و اگر لاگین موفق نباشد پیام غلط بودن یوزر یا پسورد را به کاربر نمایش می‌دهد

```
usarray = [""]*50
pwdarray = [""]*50

def read_db(): # خواندن پایگاه داده
    dbuser = open("user.txt", "r")
    dbpass = open("pass.txt", "r")

    userdata = dbuser.readlines()
    passdata = dbpass.readlines()

    usercount = 0
    passcount = 0

    for i in userdata: # ریختن اطلاعات در یک آرایه
        usarray[usercount] = i.strip()
        usercount += 1

    for i in passdata: # ریختن اطلاعات در یک آرایه
        pwdarray[passcount] = i.strip()
        passcount += 1

    dbuser.close()
    dbpass.close()
```

حال به تابع `read_db` می‌رسیم که اطلاعات یوزرها و پسوردها را می‌خواند و آن را در دو آرایه می‌ریزد (که آن را راحت‌تر از آرایه بررسی کنیم و مجبور نباشیم که فایل را دوباره باز کنیم)

کد تابع `check_login` به صورت زیر است :

```
def check_login(username,password): # بررسی درست بودن لاگین
    for i in range(0,50):
        if (usarray[i] == username and cryptocode.decrypt(pwdarray[i],"ahmad") == password):
            return True
```

این تابع یوزر و پسورد را از آرایه‌هایی که ساختیم می‌گیرد و آن‌ها را دیگد (آن هاش) می‌کند و با یوزر و پسوردی که کاربر داده مقایسه می‌کند ، در صورتی که درست بود مقدار `true` را بر می‌گرداند.

حال به قسمت ثبت نام می‌رسیم :

```
elif(menu == '2'): # برای ثبت نام
    username = input("enter your desired username : ")
    if(check_username(username) == True):
        print("username is already taken")
    else:
        password = input("enter your password : ")
        hashed = cryptocode.encrypt(password , "ahmad")
        dbuser = open("user.txt","a")
        dbpass = open("pass.txt","a")
        dbuser.write(username + '\n')
        dbpass.write(hashed + '\n')
```

برای ثبت نام اول از کاربر یوزرنیم مورد نظر را می‌خواهیم سپس یوزر نیم را به تابع `check_username` پاس می‌دهیم تا بررسی کند که آیا تکراری است یا خیر.

در صورت تکراری نبود ، یوزرنیم را وارد فایل دیتابیس می کند و پسورد را هش کرده و پسورد هش شده را در فایل دیتابیس پسورد ها می نویسد (برای هش و آن هش از افزونه CryptoCode در پایتون استفاده کردم)

کُد تابع check_username به صورت زیر است که اول بررسی می کند که یوزرنیم در آرایه ای که از پایگاه داده یوزر ها ساختیم وجود دارد یا خیر و در صورت وجود آن مقدار true را بر می گرداند :

```
def check_username(username): # بررسی بودن یوزر تیم در دیتابیس
    for i in range(0,50):
        if (usarray[i] == username):
            return True
```

و در آخر نیز به دو تابع زیر می رسیم :

```
def handle_messages(self): # ارسال مسیج ها
    while 1:
        print(self.s.recv(1204).decode())

def input_handler(self): # نمایش پیام چت
    while 1:
        self.s.send((self.username+' : '+input()).encode())
```

تابع اول ، مسیج های مشتری های دیگر را هندل می کند و به ما نمایش می دهد
تابع دوم نیز پیام های ما را در چت روم به صورت username : message به باقی کاربر ها نمایش می دهد.

در آخر عکس هایی را از اجرای برنامه میبینیم :

```
C:\Program Files (x86)\Python38-32\python.exe
Running on host: 192.168.42.69
Running on port: 5050
New connection. Username: ahmad
New message: ahmad : hi
```

```
C:\Program Files (x86)\Python38-32\python.exe
if you want to login type 1
if you want to sign up type 2
1
Enter username --> ahmad
Enter pass --> 123456
hi
_
```

<pre>C:\Program Files (x86)\Python38-32\python.exe Running on host: 192.168.42.69 Running on port: 5050 New connection. Username: hamed New message: hamed : hello New connection. Username: ahmad New message: ahmad : hello guys !</pre>	<pre>C:\Program Files (x86)\Python38-32\python.exe if you want to login type 1 if you want to sign up type 2 1 Enter username --> ahmad Enter pass --> 123456 hello guys !</pre>	<pre>C:\Program Files (x86)\Python38-32\python.exe if you want to login type 1 if you want to sign up type 2 1 Enter username --> hamed Enter pass --> 654321 hello New person joined the room. Username: ahmad ahmad : hello guys ! _</pre>
--	--	--

با تشکر از توجه شما