PES University, Bangalore

(Established under Karnataka Act No. 16 of 2013)

UE21CS252B- Computer Networks

Assignment 1 and Assignment 2

Semester & Section: 4th semester D section

Sl No.	Name of the Student	SRN
1.	Gaargi V	PES1UG21CS193
2.	Ganta Srilalitha	PES1UG21CS200
3.	Ishu Singh	PES1UG21CS242

Assignment 1

Problem Statement: Multilingual Chat Room. Each user will be able to see all messages in the language of his choice. At the server, all messages are in English. This, when forwarded to all clients is translated to their various languages at the client side.

Screenshot	Explanation
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PS C:\Users\Gaargi V\Desktop\PES\Sem 4\CN\CN Assignment 1> python server.py running server on 127.0.0.1 1234	First start the server. There is an acknowledgement message displayed.
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PS C:\Users\Gaargi V\Desktop\PES\Sem 4\CN\CN Assignment 1> python client.py successfully connected to server enter usernameabc Enter LanguageFrench Msg:Bonjour Msg:[abc]Pepitoooo Msg:Comment ca va Msg:[abc]Comment allez-vous Msg:[hjk]Bonjour	Client abc who types in French has joined the chat room.
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PS C:\Users\Gaargi V\Desktop\PES\Sem 4\CN\CN Assignment 1> python client.py successfully connected to server enter usernamepqr Enter LanguageKannada Msg:[abc]ನಮಸ್ತೆ [abc]ನಿಂ ಹೆಸರೇನಿ ಯಾರು [hjk]	Client pqr uses Kannada. He sees abc's message in Kannada.

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

PS C:\Users\Gaargi V\Desktop\PES\Sem 4\CN\CN Assignment 1> python client.py
successfully connected to server
enter usernamehjk
Enter LanguageGerman
Msg:Guten morgen
Msg:[hjk]Guten Morgen!
```

Client hjk uses German.

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

PS C:\Users\Gaargi V\Desktop\PES\Sem 4\CN\CN Assignment 1> python server.py
running server on 127.0.0.1 1234
successfully connected to client127.0.0.1 49952
successfully connected to client127.0.0.1 49954
Hello
How are you
successfully connected to client127.0.0.1 49969
Good morning
```

An acknowledgement is displayed at the server everytime a client joins. First abc, then pqr , then hjk have joined the chat.

Messages sent by all clients are seen in English at the server.

Code

```python
#Server Code
import socket
import threading

HOST='127.0.0.1'
PORT=1234
LISTENER_LIMIT=5
active_clients=[]
from translate import Translator

def listen_for_messages(client,username):
    while(1):
        message=client.recv(2048).decode('utf-8')
        if message !='':
            print (message)
            final_msg=username+'~'+message
            send_messages_to_all(final_msg)

        else:
            print(f'msg sent from client{username} is empty')

#fun to send msg to client serially which r there in a list
def send_message_to_client(client,message):
    client.sendall(message.encode())
```

```python
#server sends msg to all clients
def send_messages_to_all(message):
    for user in active_clients:
        send_message_to_client(user[1],message)

#func to handle client
def client_handler(client):
    #server will listen for client msg that contains username
    while(1):
        username=client.recv(2048).decode('utf-8')#2048-ms size
        if username !='':
            active_clients.append((username,client))
            break
        else:
            print('client is empty')
    threading.Thread(target=listen_for_messages,args=(client,username)).start()
def main():
    server=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
    print(f"running server on {HOST} {PORT}")
    try:
        server.bind((HOST,PORT))
    except:
        print(f'unable to bind to host{HOST} and port{PORT}')

    #set server limit
    server.listen(LISTENER_LIMIT)

    while(1):
        client,address=server.accept()
        print(f'successfully connected to client{address[0]} {address[1]}')
        threading.Thread(target=client_handler,args=(client,)).start() #start- tostart
thread
if __name__=='__main__':
    main()
```

```python
#Client Code
import socket
import threading
import tkinter as tk
from tkinter import scrolledtext
from tkinter import messagebox
HOST='127.0.0.1'
PORT=1234

from translate import Translator

#client to listen to server msg
def listen_for_messages_from_server(client,lang):
    while(1):
        message=client.recv(2048).decode('utf-8')
        if message!='':
```

```
            username=message.split("~")[0]
            content=message.split("~")[1]
            translator= Translator(to_lang=lang)
            translationreceived = translator.translate(content)
            print(f"[{username}]{translationreceived}")
        else:
            print('message recieved from client os empty')

def send_message_to_server(client,lang):
    while(1):
        message=input("Msg:")


        if(message!='\n'):

            translator1= Translator(to_lang="English",from_lang=lang)
            translation = translator1.translate(message)

            client.sendall(translation.encode())
        else:
            print('empty message')
            exit(0)

def communicate_to_server(client):
    username=input('enter username')
    lang=input("Enter Language")
    if username!='':
        client.sendall(username.encode())
    else:
        print('username cant be empty')
        exit(0)
    threading.Thread(target=listen_for_messages_from_server,args=(client,lang)).start()
    send_message_to_server(client,lang)
def main():
    client=socket.socket(socket.AF_INET,socket.SOCK_STREAM)

    #connect ro server
    try:
        client.connect((HOST,PORT))
        print('successfully connected to server')
    except:
        print(f'unable to bind to host{HOST} and port{PORT}')

    communicate_to_server(client)

if __name__=='__main__':
    main()
```

Assignment 2

Problem Statement: Port Scanner

Screenshot	Explanation
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL What you want to scan?: 204.79.197.222 Starting scan on host: 204.79.197.222 port 440 is closed port 441 is closed port 442 is closed port 443 is open port 444 is closed Time taken 63.21 seconds PS C:\Users\Gaargi V\Desktop\PES\Sem 4\CN\CN Assignment 2> * History restored PS C:\Users\Gaargi V\Desktop\PES\Sem 4\CN\CN Assignment 2>	The IP Address of the host to be scanned is given as the input. The program then scans ports from 440 to 445. This range is fixed in the code to minimize the time taken to find an open port. The port 443 is open. The code tries to connect to each of these ports. If possible, it prints open. Else closed.

Code

```
# Here we import two modules, socket and time

import socket
import time

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

# here we asking for the target website
# or host
target = input('What you want to scan?: ')

# next line gives us the ip address
# of the target
target_ip = socket.gethostbyname(target)
print('Starting scan on host:', target_ip)

# function for scanning ports


def port_scan(port):
    try:
        s.connect((target_ip, port))
        return True
    except:
        return False
```

```python
start = time.time()

# here we are scanning port 440 to 445
for port in range(440,445):
    if port_scan(port):
        print(f'port {port} is open')
    else:
        print(f'port {port} is closed')

end = time.time()
print(f'Time taken {end-start:.2f} seconds')
```