V.SWARNAMALA **241901116**

ECHOSERVER **Date: 18/8/25**

Aim

To create a client-server application where the server listens for client connections, receives messages, and echoes the same message back to the client using socket programming in Python.

Introduction

Socket programming enables communication between two nodes over a network. This experiment demonstrates TCP socket communication where the server creates a socket and listens for incoming connections, while the client connects to the server, sends a message, and receives the same message in response. This basic Echo Server is foundational for understanding client-server architecture and network communication protocols.

Procedure

1. **Server Side:**
    - Import the socket module.
    - Create a socket object using AF_INET for IPv4 and SOCK_STREAM for TCP.
    - Bind the socket to a host and port (e.g., localhost and port 55555).
    - Put the socket into listening mode with a backlog queue (e.g., 3).
    - Use an infinite loop to accept incoming client connections.
    - Receive data from the connected client.
    - Echo the received data back to the client.
    - Allow the server to continue or break the loop based on user input.

2. **Client Side:**
    - Import the socket module.
    - Create a socket object similarly as the server.
    - Connect to the server using the server's host and port.
    - Send a message to the server after taking input from the user.
    - Receive the echoed message from the server.
    - Display the message to the user.

This experiment helps to understand socket creation, binding, listening, accepting connections, sending and receiving data over networks in a client-server model.

PROGRAM:

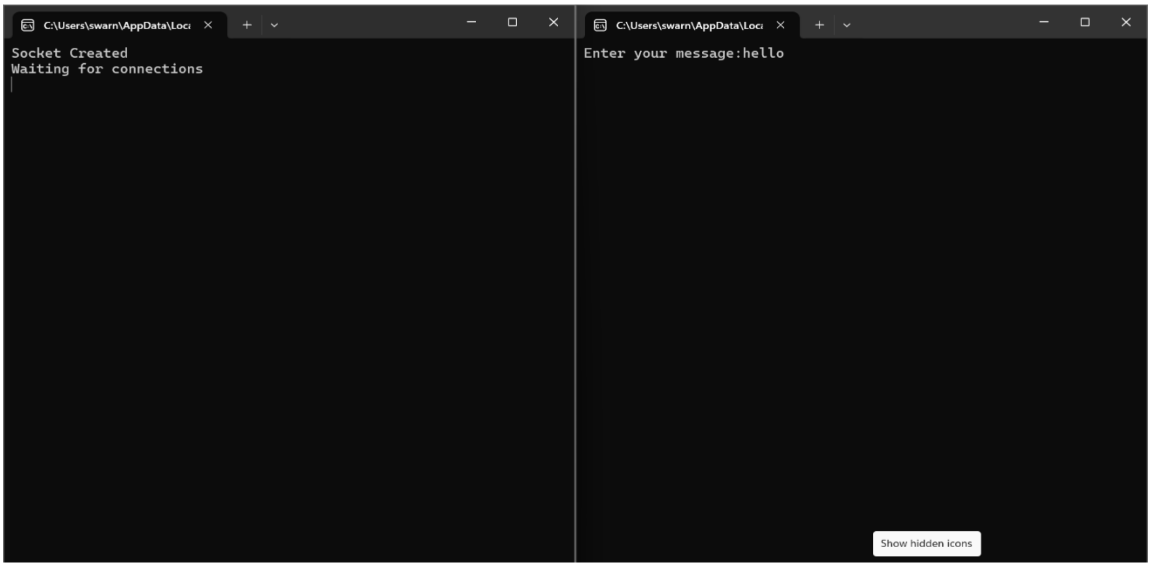
SERVER:

```
import socket
sockfd=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
print('Socket Created')
sockfd.bind(('localhost',55555))
sockfd.listen(3)
print('Waiting for connections')
while True:
   clientfd,addr=sockfd.accept()
   receivedMsg=clientfd.recv(1024).decode()
   print("Connected with ",addr)
   print("Message Received from Client: ",receivedMsg)
   clientfd.send(bytes(receivedMsg,'utf-8'))
   print("Message reply sent to Client!")
   print("Do you want to continue(type y or n):")
   choice=input()
   if choice=='n':
      break
```

CLIENT:

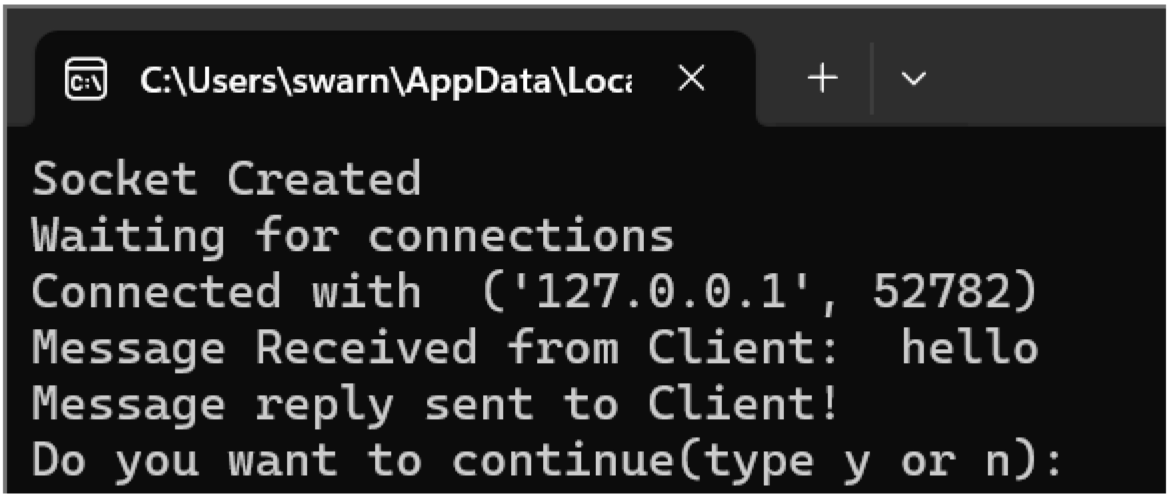
```
import socket
clientfd=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
clientfd.connect(('localhost',55555))
name=input("Enter your message:")
clientfd.send(bytes(name,'utf-8'))
print("Message Received from Server: ",clientfd.recv(1024).decode())
```

OUTPUT:



```
Socket Created
Waiting for connections
```

```
Enter your message:hello
```



```
Socket Created
Waiting for connections
Connected with  ('127.0.0.1', 52782)
Message Received from Client:  hello
Message reply sent to Client!
Do you want to continue(type y or n):
```

RESULT:

Thus the program for echoserver has been executed.