Cyber Security

Lab-1

Gohel Mruganshi Jatinbhai
B20CS014

1. Create a client-server simple handshaking program using socket programming.

In the below code, the server will create a socket object, bind it to an address and port, and start listening for incoming connections. and after the establishment of the connection, it will send a handshaking message to the client and then close the connection. On the other hand, the client creates a socket object, connects to the server and sends a handshaking message to the server, then receives the message from the server, prints it, and closes the connection. here, we use the local host as the server address, meaning that we are running the server and client on the same device.

Client

```
import socket
SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((SERVER_ADDRESS, SERVER_PORT))
client_socket.sendall(b"Requesting a handshake")
data = client_socket.recv(1024)
print("Received: ", data.decode())
client_socket.close()
```

Server

```
import socket
SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind((SERVER_ADDRESS, SERVER_PORT))
server_socket.listen()
print("Listening for incoming connections...")
```

```
client_socket, client_address = server_socket.accept()
print("Connection from", client_address, "has been established.")
client_socket.sendall(b"Handshake successful!")
client_socket.close()
server_socket.close()
```

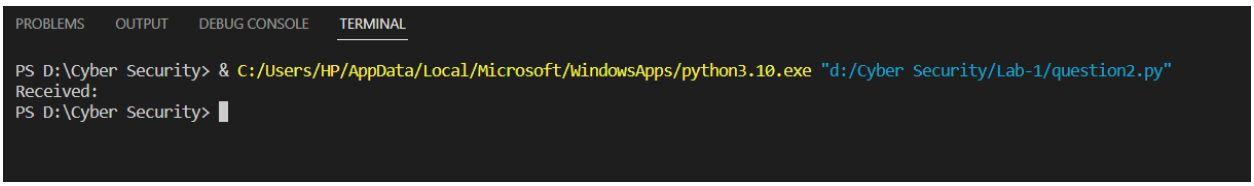
output

2. Create a socket program to connect to a predefined server (try https://iitj.ac.in/ use others too).

You will need to replace localhost with the actual address of the server you want to connect to, and the port number if it's different in the client code . This uses port 80, which is the default port for HTTP.

```
import socket
SERVER_ADDRESS = "iitj.ac.in"
SERVER_PORT = 80
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((SERVER_ADDRESS, SERVER_PORT))
client_socket.sendall(b"Hello, server!")
data = client_socket.recv(1024)
print("Received: ", data.decode())
client_socket.close()
```

output



3. Create a client server program using socket programming, Where The server will simply echo whatever it receives back to the client.

the server will create a socket object , bind it to the specific address and port, and start listening for all incoming connections. after the establishment of connection it will receive a data from the client and echo it back to the client. and likewise the client will create a socket object, connect to the server and send a message, then send a message to the server and print it.

client

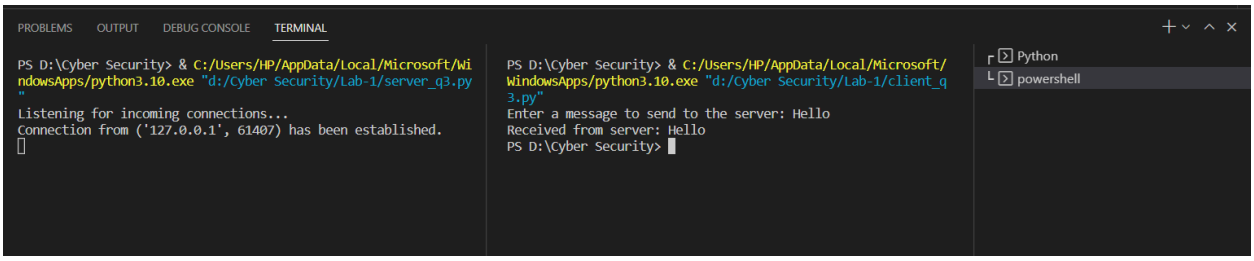
```
import socket
SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((SERVER_ADDRESS, SERVER_PORT))
data = input("Enter a message to send to the server: ")
client_socket.sendall(data.encode())
data = client_socket.recv(1024)
print("Received from server:", data.decode())
client_socket.close()
```

server

```
import socket
SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind((SERVER_ADDRESS, SERVER_PORT))
server_socket.listen()
print("Listening for incoming connections...")

while True:
    client_socket, client_address = server_socket.accept()
    print("Connection from", client_address, "has been established.")
    data = client_socket.recv(1024)
    client_socket.sendall(data)
    client_socket.close()
server_socket.close()
```

Output



4. Create a client server program using socket programming, Where the client sends a text message in Lowercase to the server and the server returns it in uppercase to the client.

the server will create a socket object , bind it to the specific address and port, and start listening for all incoming connections. after the establishment of connection it will receive a data from the client and echo it back to the client. and likewise the client will create a socket object, connect to the server and send a message, then receive a message from the server and print it in the uppercase.

client

```
import socket

SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((SERVER_ADDRESS, SERVER_PORT))
data = input("Enter a message to send to the server: ")
client_socket.sendall(data.encode())
data = client_socket.recv(1024).decode()
print("Received from server:", data)
client_socket.close()
```

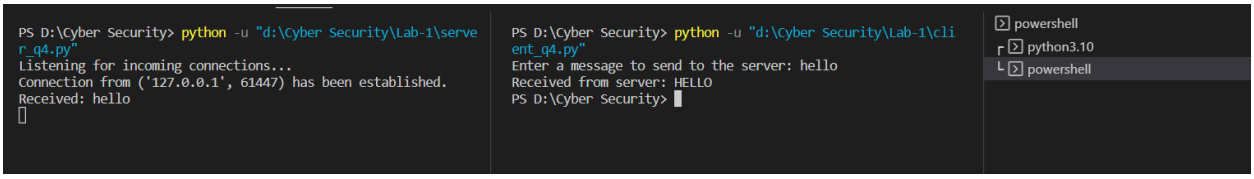
server

```
import socket
SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind((SERVER_ADDRESS, SERVER_PORT))
server_socket.listen()
print("Listening for incoming connections...")

while True:
    client_socket, client_address = server_socket.accept()
```

```
    print("Connection from", client_address, "has been established.")
    data = client_socket.recv(1024).decode()
    print("Received:", data)
    client_socket.sendall(data.upper().encode())
    client_socket.close()
server_socket.close()
```

Output



5. Create a client server program using socket programming, Where the client sends a text message to the server and the server returns it in reverse order to the client.

the server will create a socket object , bind it to the specific address and port, and start listening for all incoming connections. after the establishment of connection it will receive a data from the client and echo it back to the client. and likewise the client will create a socket object, connect to the server and send a message, then receive a message from the server and print it in the reverse order.

client

```
import socket

SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((SERVER_ADDRESS, SERVER_PORT))
data = input("Enter a message to send to the server: ")
client_socket.sendall(data.encode())
data = client_socket.recv(1024).decode()
print("Received from server:", data)
client_socket.close()
```

server

```
import socket
SERVER_ADDRESS = "localhost"
SERVER_PORT = 1234
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind((SERVER_ADDRESS, SERVER_PORT))
server_socket.listen()
print("Listening for incoming connections...")

while True:
    client_socket, client_address = server_socket.accept()
    print("Connection from", client_address, "has been established.")
    data = client_socket.recv(1024).decode()
    print("Received:", data)
    client_socket.sendall(data[::-1].encode())
    client_socket.close()
server_socket.close()
```

Output