EXP-8 CN

1) TCP

A)TCP_SERVER

tcp_server.py

```python
import socket

TCP_IP = '127.0.0.1'
TCP_PORT = 5005
BUFFER_SIZE = 1024

# Create TCP socket
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind((TCP_IP, TCP_PORT))
s.listen(1)

print(f"Server started.")

conn, addr = s.accept()
print("Connection from:", addr)

while True:
    data = conn.recv(BUFFER_SIZE)
    if not data:
        break
    print("Received:", data.decode())
    conn.send(data)
conn.close()
```

```
comp@comp:~/cn8$ python tcp_server.py
Server started.
Connection from: ('127.0.0.1', 59346)
Received: Namaste Duniya
comp@comp:~/cn8$
 * History restored
```

B) TCP_CLIENT

tcp_client.py > ...

```python
import socket

TCP_IP = '127.0.0.1'
TCP_PORT = 5005
BUFFER_SIZE = 1024
MESSAGE = "Namaste Duniya"


s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((TCP_IP, TCP_PORT))

print(" Sending:", MESSAGE)
s.send(MESSAGE.encode())

data = s.recv(BUFFER_SIZE)
print(" Received from server:", data.decode())

s.close()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
comp@comp:~/cn8$ python tcp_server.py
Server started.
Connection from: ('127.0.0.1', 59346)
Received: Namaste Duniya
comp@comp:~/cn8$
* History restored
```

2)UDP

A)UDP_SERVER

udp_server.py

```python
import socket

UDP_IP = "127.0.0.1"
UDP_PORT = 5005
BUFFER_SIZE = 1024

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

s.bind((UDP_IP, UDP_PORT))

print(f"UDP Server started.")

while True:
    data, addr = s.recvfrom(BUFFER_SIZE)
    print("Received from", addr, ":", data.decode())
    response = f"Server got: {data.decode()}"
    s.sendto(response.encode(), addr)
```

```
comp@comp:~/udp$ python udp_server.py
UDP Server started.
Received from ('127.0.0.1', 42793) : Hello UDP Server
```

B)UDP_CLIENT

udp_client.py > ...

```python
import socket

UDP_IP = "127.0.0.1"
UDP_PORT = 5005
BUFFER_SIZE = 1024
MESSAGE = "Hello UDP Server"

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

print("Sending:", MESSAGE)
s.sendto(MESSAGE.encode(), (UDP_IP, UDP_PORT))

data, server = s.recvfrom(BUFFER_SIZE)
print("Received from server:", data.decode())

s.close()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
comp@comp:~/udp$ python udp_client.py
Sending: Hello UDP Server
Received from server: Server got: Hello UDP Server
comp@comp:~/udp$
```