NETWORK PROGRAMMING LAB

LAB 9 – UDP CLIENT SERVER CHAT

NAME : SREENIDHI GANACHARI

REG NO : 19BCE7230

Q) UDP based Client Server Chat, with GUI. Display the Client / Server's name before the message as and when received.

UDPServer.java

```
import java.net.*;
import java.io.*;
public class UDPServer
{
 public static void main(String args[])
 {
 int port = 8001;

 DatagramSocket serverDatagramSocket = null;
 try
 {
 serverDatagramSocket = new DatagramSocket(port);
 System.out.println(" UDP Server : "+port);
 }
 catch(IOException e)
 {
 System.out.println(e);
 System.exit(1);
 }
```

```
try
{
byte buffer[] = new byte[1024];
DatagramPacket datagramPacket = new
DatagramPacket(buffer, buffer.length);
String input;
while(true)
{

serverDatagramSocket.receive(datagramPacket);
input = new String(datagramPacket.getData(), 0,
datagramPacket.getLength());
System.out.println("Received : "+input);
serverDatagramSocket.send(datagramPacket);
}
}
catch(IOException e)
{
System.out.println(e);
}
}
}
```

UDPClient.java

```
import java.net.*;
import java.io.*;
public class UDPClient
{
 public static class UDPReader extends Thread
 {
 public UDPReader(DatagramSocket socket)
```

```
{
datagramSocket = socket;
active = true;
}
public void run()
{
byte[] buffer = new byte[1024];
DatagramPacket incoming = new DatagramPacket(buffer,
buffer.length);
String receivedString;
while(active)
{
try
{

datagramSocket.receive(incoming);

receivedString = new String(incoming.getData(),
0, incoming.getLength());
System.out.println("Received : "+receivedString);
}
catch(IOException e)
{
System.out.println(e);
active = false;
}
}
}
public boolean active;
public DatagramSocket datagramSocket;
}
```

```
public static void main(String[] args)
{
InetAddress address = null;
int port = 8001;
DatagramSocket datagramSocket = null;
BufferedReader keyboardReader = null;

try
{
address = InetAddress.getByName("127.0.0.1");
datagramSocket = new DatagramSocket();
keyboardReader = new BufferedReader(new InputStreamReader(System.in));
}
catch (IOException e)
{
System.out.println(e);
System.exit(1);
}

UDPReader reader = new UDPReader(datagramSocket);
reader.setDaemon(true);
reader.start();
System.out.println("Ready to send your messages...");
try
{
String input;
while (true)
{

input = keyboardReader.readLine();
```

```
DatagramPacket datagramPacket = new DatagramPacket
(input.getBytes(), input.length(), address, port);
datagramSocket.send(datagramPacket);
}
}
catch(IOException e)
{
System.out.println(e);
}
}
}
```

OUTPUT –