

Lab3

Q1) TCP/IP Chat Programme

Java Code:

Client:

[illegible]

```
        str = sc.nextLine();
        synchronized (clientsock) {
            dos.writeUTF(str);
        }
        Thread.sleep(400);
    } catch (Exception e) {
        System.out.println(e.getMessage());
    } //try-catch
} //while
} //run
} ; //threads
```

```
ts.setPriority(Thread.MAX_PRIORITY);
ts.start();
```

```
} //send()
```

```
public void Receive() {
```

```
    Thread tr = new Thread() {
```

```
        public void run() {
```

```
            while (clientsock.isConnected()) {
```

```
                try {
```

```
                    str = dis.readUTF();
```

```
                    System.out.println("Message from Server:" + str);
```

```
                } catch (Exception e) {
```

```
                    System.out.println(e.getMessage());
```

```
                } //try-catch
```

```

        }//while
    }//run
};//threads

tr.setPriority(Thread.MAX_PRIORITY);
tr.start();
} //Receive()

public void createConnection() {
    try {
        clientsock = new Socket("localhost", 1121);
        dos = new DataOutputStream(clientsock.getOutputStream());
        dis = new DataInputStream(clientsock.getInputStream());
        Receive();
        Send();
    } catch (Exception e) {
        System.out.println(e.getMessage());
    } //try-catch

} //createConnection()

public static void main(String[] args) {

    ChatClient cc = new ChatClient();
    cc.createConnection();

} //main()

} //class

```

Server:

```
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.net.*;
import java.util.Scanner;

/**
 *
 * @author Abhishek Karan
 */
public class ChatServer {

    ServerSocket serversock = null;//new ServerSocket(1121);
    Socket sock = null;// new Socket("localhost", 1121);
    DataOutputStream dos = null;
    Scanner sc = new Scanner(System.in);
    DataInputStream dis = null;
    String str = null;//sc.nextLine();

    public void Send() {

        Thread ts = new Thread() {

            public void run() {

                while (sock.isConnected()) {
                    try {
                        System.out.println("Enter Message:");
                        str = sc.nextLine();
```

```

        synchronized (sock) {
            dos.writeUTF(str);
        }
        Thread.sleep(400);
    } catch (Exception e) {
        System.out.println(e.getMessage());
    } //try-catch
} //while
} //run
}; //threads

ts.setPriority(Thread.MAX_PRIORITY);
ts.start();

} //send()

public void Receive() {

    Thread tr = new Thread() {

        public void run() {

            while (sock.isConnected()) {
                try {
                    str = dis.readUTF();
                    System.out.println("Message from Server:" + str);
                } catch (Exception e) {
                    System.out.println(e.getMessage());
                } //try-catch
            } //while
        }
    };
}

```

```

        }//run
    };//threads

    tr.setPriority(Thread.MAX_PRIORITY);
    tr.start();
} //Receive()

public void createConnection() {
    try {
        serversock = new ServerSocket(1121);
        while (true) {
            sock = serversock.accept();
            dis = new DataInputStream(sock.getInputStream());
            dos = new DataOutputStream(sock.getOutputStream());
            Receive();
            Send();
        } //while
    } catch (Exception e) {
        System.out.println(e.getMessage());
    } //try-catch
} //createConnection()

public static void main(String[] args) {

    ChatServer cs = new ChatServer();
    cs.createConnection();

} //main()

```

```
}//class
```

Q2) TCP/IP Demonstration:

Java Code:

Client:

```
package Ques1;
```

```
import java.io.*;
```

```
import java.net.*;
```

```
import java.util.Scanner;
```

```
/**
```

```
 *
```

```
 * @author Abhishek Karan
```

```
 */
```

```
public class TCPClient {
```

```
    public static void main(String[] args) {
```

```
        try {
```

```
            Socket sock = new Socket("localhost", 1121);
```

```
            DataOutputStream dos = new DataOutputStream(sock.getOutputStream());
```

```
            System.out.println("Enter Message:");
```

```
            Scanner sc = new Scanner(System.in);
```

```
            String str = sc.nextLine();
```

```
            dos.writeUTF(str);
```

```
            DataInputStream dis = new DataInputStream(sock.getInputStream());
```

```
            System.out.println("Client:" + (String) dis.readUTF());
```

```
            dos.flush();
```

```
        dos.close();
        sock.close();
    } catch (Exception e) {
        System.out.println(e.getMessage());
    }
} //main()
```

```
} //class
```

Server:

```
package Ques1;
```

```
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.net.*;
import java.util.Scanner;
```

```
/**
```

```
 *
```

```
 * @author Abhishek Karan
```

```
 */
```

```
public class TCPServer {
```

```
    public static void main(String[] args) {
```

```
        try {
```

```
            Scanner sc = new Scanner(System.in);
```

```
            ServerSocket ss = new ServerSocket(1121);
```

```
            Socket sock = ss.accept();
```

```
            DataInputStream dis = new DataInputStream(sock.getInputStream());
```



```

        String str = (String) dis.readUTF();

        System.out.println("Server:" + str);

        System.out.println("Enter Message:");


        str = sc.nextLine();

        DataOutputStream dos = new DataOutputStream(sock.getOutputStream());

        dos.writeUTF(str);


        sock.close();
    } catch (Exception e) {

        System.out.println(e.getMessage());

    }

}

} //main

} //class

```

Q3) UDP Demonstration:

Java Code:

Client:

```
package Ques1;
```

```
import java.net.*;
```

```
import java.util.Scanner;
```

```
/**
```

```
*
```

```
* @author Abhishek Karan
```

```
*/
```

```
public class UDPClient {
```

```

public static void main(String[] args) {

    try {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Message:");
        String str = sc.nextLine();

        DatagramSocket dgs = new DatagramSocket();

        InetAddress ina = InetAddress.getByName("127.0.0.1");

        DatagramPacket dgp = new DatagramPacket(str.getBytes(), str.length(), ina, 1121);

        dgs.send(dgp);
        dgs.close();

    } catch (Exception e) {
        System.out.println(e.getMessage());
    }

} //main()

} //class

Server:
package Ques1;

import java.net.*;
import java.util.Scanner;

```

```
/**
 *
 * @author Abhishek Karan
 */
public class UDPServer {

    public static void main(String[] args) {

        try {
            Scanner sc = new Scanner(System.in);

            DatagramSocket dgs = new DatagramSocket(1121);
            byte[] buf = new byte[1024];

            DatagramPacket dgp = new DatagramPacket(buf, buf.length);
            dgs.receive(dgp);
            String str = new String(dgp.getData(), 0, dgp.getLength());
            System.out.println("Server:" + str);

            //System.out.println("Enter Message:");
            //str = sc.nextLine();
            dgs.send(dgp);
            dgs.close();
        } catch (Exception e) {
            System.out.println(e.getMessage());
        }
    } //main()
} //class
```

Abhishek Karan

130911122