NETWORK PROGRAMMING LAB

LAB 8 - SOCKET CLIENT SERVER – SECURE CONNECTION

NAME : SREENIDHI GANACHARI

REG NO : 19BCE7230

Q) Justify through java code: Secure Client / Server communication. Compare the time taken for file exchange between Client and Server, with and Without Secure communication

CODE –

SocketServer.java

```
import java.io.*;

import java.net.*;

class SocketServer {

public static void main(String args[])

throws Exception

{

ServerSocket ss = new ServerSocket(8100);

Socket s = ss.accept();

System.out.println("Connection established");

PrintStream ps

= new PrintStream(s.getOutputStream());

BufferedReader br

= new BufferedReader(

new InputStreamReader(

s.getInputStream()));

BufferedReader kb

= new BufferedReader(

new InputStreamReader(System.in));
```

```java
while (true) {

String str, str1;

while ((str = br.readLine()) != null) {

System.out.println("Message Recevied = "+str);

str1 = kb.readLine();

ps.println(str1);

if(!str1.equals("exit"))

{

System.out.println("Message Sent ");

}

}

ps.close();

br.close();

kb.close();

ss.close();

s.close();

System.exit(0);

}

}

}
```

SocketClient.java

```java
import java.util.*;

import java.io.*;

import java.net.*;

class SocketClient {

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

static String user[]={"sreenidhi", "padma","ramamohan"};

static String pass[]={"pass1","pass2","pass3"};

public static void username()

{

System.out.println("Enter Username");
```

```java
String str=sc.next();

boolean count=false;

for(int i=0;i<user.length;i++)

{

if(str.equals(user[i]))

{

count=false;

password(i);

break;

}

else

{

count=true;

}

}

if(count)

{

System.out.println("Please Try Again");

username();

}

}

public static void password(int j)

{

System.out.println("Enter Password ");

String str1=sc.next();

int count=0;

if(str1.equals(pass[j]))

{

System.out.println("Logged In");

}

else
```

```java
{
System.out.println("Please Try Again");
password(j);
}
}
public static void main(String args[]) throws Exception
{
username();
Socket s = new Socket("localhost", 8100);
DataOutputStream dos
= new DataOutputStream(
s.getOutputStream());
BufferedReader br
= new BufferedReader(
new InputStreamReader(
s.getInputStream()));
BufferedReader kb= new BufferedReader(
new InputStreamReader(System.in));
String str, str1;
while (!(str = kb.readLine()).equals("exit")) {
dos.writeBytes(str + "\n");
System.out.println("Message Sent ");
str1 = br.readLine();
System.out.println("Message Recevied = "+str1);
}
dos.close();
br.close();
kb.close();
s.close();
}
}
```

OUTPUT -