**Experiment 2**

**Roll No.**

**Aim:** To implement movie-booking system using MultiThreaded Server

**Program:**

**Server:**

import java.io.\*;

import java.net.\*;

import java.util.\*;

import java.io.IOException;

import java.io.PrintStream;

import java.net.ServerSocket;

import java.net.Socket;

public class MultiThreadServer implements Runnable {

volatile static int[] seats= new int[]{1,2,3,4,5};

volatile static int[] avail= new int[]{0,0,0,0,0};

Socket csocket;

DataInputStream dis;

DataOutputStream dos;

MultiThreadServer(Socket csocket) {

this.csocket = csocket;

}

public static void main(String args[]) throws Exception {

ServerSocket ssock = new ServerSocket(1068);

System.out.println("Listening");

while (true) {

Socket sock = ssock.accept();

System.out.println("Connected");

new Thread(new MultiThreadServer(sock)).start();

}

}

public void run() {

try {

System.out.println(csocket);

System.out.println("CLIENT CONNECTED");

dis= new DataInputStream(csocket.getInputStream());

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

String str, s1;

do

{

str=dis.readUTF();

System.out.println("Client Message:"+str);

String [] div;

div=str.split(" ");

str="Movie: "+div[0]+"\nSeat:"+div[1];

if(avail[Integer.parseInt(div[1])-1]==0)

{

avail[Integer.parseInt(div[1])-1]=1;

s1="Seat Booked";

}

else

{

s1="Seat Not available";

}

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

for(int i=0;i<5;i++)

{ System.out.println(avail[i]);

}

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

dos.writeUTF(s1);

dos.flush();

}

while(!s1.equals("bye"));

}

catch (IOException e) {

System.out.println(e);

}

}

}

**Client:**

import java.io.\*;

import java.net.\*;

public class MyClient1

{

Socket s;

DataInputStream din;

DataOutputStream dout;

public MyClient1()

{

try

{

//s=new Socket("10.10.0.3,10");

s=new Socket("10.0.7.106",1068);

System.out.println(s);

din= new DataInputStream(s.getInputStream());

dout= new DataOutputStream(s.getOutputStream());

ClientChat();

}

catch(Exception e)

{

System.out.println(e);

}

}

public void ClientChat() throws IOException

{

BufferedReader br= new BufferedReader(new InputStreamReader(System.in));

String s1;

do

{

s1=br.readLine();

dout.writeUTF(s1);

dout.flush();

System.out.println("Server Message:"+din.readUTF());

}

while(!s1.equals("stop"));

}

public static void main(String as[])

{

new MyClient1();

}

}

**Output:**

**Server**

C:\Users\Folder\Downloads>javac MultiThreadServer.java

C:\Users\Folder\Downloads>java MultiThreadServer

Listening

Connected

Socket[addr=/10.0.24.59,port=8062,localport=1068]

CLIENT CONNECTED

Client Message:chapak 1

Message:

1

0

0

0

0

Client Message:chapak 2

Message:

1

1

0

0

0

**Client**

C:\Users\Folder>cd Downloads

C:\Users\Folder\Downloads>javac MyClient1.java

C:\Users\Folder\Downloads>java MyClient1

Socket[addr=/10.0.24.59,port=1068,localport=8062]

chapak 1

Server Message:Seat Booked

chapak 2

Server Message:Seat Booked

**Conclusion:** Thus, we have successfully implemented movie-booking system using MultiThreaded Server