**Name:** Mitrajeet Golsangi

**Roll No:** 01

**PRN:** 12010484

**Division:** Ty-CS-B

**Batch:** B1

**TY. B. Tech.**

**CS3052: Computer Networks**

**Lab No: 5**

**Develop a client server using TCP Berkeley socket primitives to transfer a file in peer-to-peer and client-server mode. Demonstrate the packets captured traces using Wireshark Packet Analyzer Tool for peer-to-peer mode.**

**Code: (Client)**

import java.io.\*;

import java.net.Socket;

public class client\_ftp {

    private static DataOutputStream dataOutputStream = null;

    private static DataInputStream dataInputStream = null;

    public static void main(String[] args)

    {

        try (Socket socket = new Socket("localhost", 4000)) {

        dataInputStream = new DataInputStream(

                socket.getInputStream());

            dataOutputStream = new DataOutputStream(

                socket.getOutputStream());

            System.out.println(

                "Sending the File to the Server");

        sendFile(

                "C:/Users/arjun/Downloads/OS-LAB.pdf");

            dataInputStream.close();

            dataInputStream.close();

        }

        catch (Exception e) {

            e.printStackTrace();

        }

    }

    private static void sendFile(String path)

        throws Exception

    {

        int bytes = 0;

        File file = new File(path);

        FileInputStream fileInputStream

            = new FileInputStream(file);

        dataOutputStream.writeLong(file.length());

        byte[] buffer = new byte[4 \* 1024];

        while ((bytes = fileInputStream.read(buffer))

            != -1) {

        dataOutputStream.write(buffer, 0, bytes);

            dataOutputStream.flush();

        }

        fileInputStream.close();

    }

}

**Code: (Server)**

import java.io.DataInputStream;

import java.io.DataOutputStream;

import java.io.FileOutputStream;

import java.net.ServerSocket;

import java.net.Socket;

public class server\_ftp {

    private static DataOutputStream dataOutputStream = null;

    private static DataInputStream dataInputStream = null;

    public static void main(String[] args)

    {

        try (ServerSocket serverSocket

            = new ServerSocket(4000)) {

            System.out.println(

                "Server is Starting in Port 4000");

            Socket clientSocket = serverSocket.accept();

            System.out.println("Connected");

            dataInputStream = new DataInputStream(

                clientSocket.getInputStream());

            dataOutputStream = new DataOutputStream(

                clientSocket.getOutputStream());

            receiveFile("NewFile.pdf");

            dataInputStream.close();

            dataOutputStream.close();

            clientSocket.close();

        }

        catch (Exception e) {

            e.printStackTrace();

        }

    }

    private static void receiveFile(String fileName)

        throws Exception

    {

        int bytes = 0;

        FileOutputStream fileOutputStream

            = new FileOutputStream(fileName);

        long size

            = dataInputStream.readLong();

        byte[] buffer = new byte[4 \* 1024];

        while (size > 0

            && (bytes = dataInputStream.read(

                    buffer, 0,

                    (int)Math.min(buffer.length, size)))

                    != -1) {

            fileOutputStream.write(buffer, 0, bytes);

            size -= bytes;

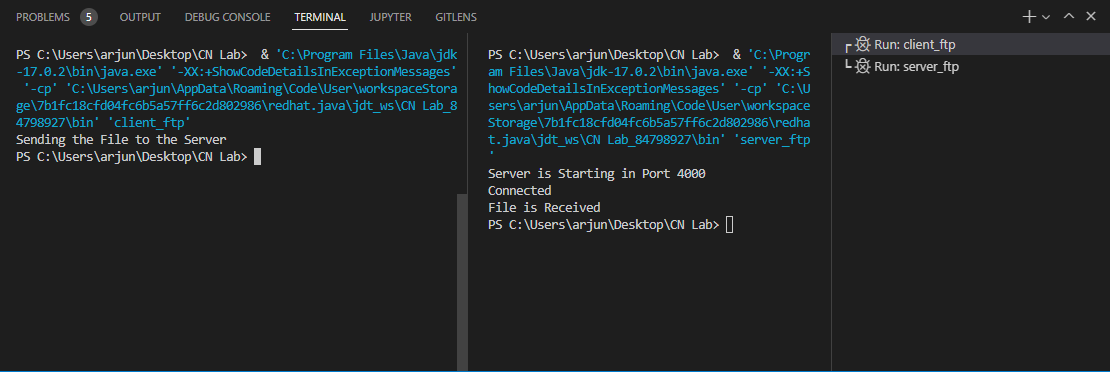
        }

        System.out.println("File is Received");

        fileOutputStream.close();

    }

}

**Output:**