Practical:14 Write a program in C/C++/ JAVA/ Python for socket programming and share your file from one system to another system.

- **Server.java**

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package socket_programming;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.FileOutputStream;
import java.net.ServerSocket;
import java.net.Socket;
/**
 *
 * @author pranav
 */

public class Server {

        private static DataOutputStream dataOutputStream = null;
        private static DataInputStream dataInputStream = null;

        public static void main(String[] args)
        {
                // Here we define Server Socket running on port 900
                try (ServerSocket serverSocket
                        = new ServerSocket(900)) {
                        System.out.println(
                                "Server is Starting in Port 900");
                        // Accept the Client request using accept method
                        Socket clientSocket = serverSocket.accept();
                        System.out.println("Connected");
                        dataInputStream = new DataInputStream(
                                clientSocket.getInputStream());
                        dataOutputStream = new DataOutputStream(
                                clientSocket.getOutputStream());
                        // Here we call receiveFile define new for that
                        // file
                        receiveFile("NewFile1.pdf");

                        dataInputStream.close();
                        dataOutputStream.close();
```

```
                clientSocket.close();
            }
            catch (Exception e) {
                e.printStackTrace();
            }
        }

        // receive file function is start here

        private static void receiveFile(String fileName)
            throws Exception
        {
            int bytes = 0;
            FileOutputStream fileOutputStream
                = new FileOutputStream(fileName);

            long size
                = dataInputStream.readLong(); // read file size
            byte[] buffer = new byte[4 * 1024];
            while (size > 0
                && (bytes = dataInputStream.read(
                            buffer, 0,
                            (int)Math.min(buffer.length, size)))
                            != -1) {
                // Here we write the file using write method
                fileOutputStream.write(buffer, 0, bytes);
                size -= bytes; // read upto file size
            }
            // Here we received file
            System.out.println("File is Received");
            fileOutputStream.close();
        }
    }
```

- **Output**

```
run:
Sending the File to the Server
File sent successfully
BUILD SUCCESSFUL (total time: 0 seconds)
```

Fig. 14.1 Sending pdf file to server using socket program

➢ Socket_programming.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package socket_programming;
import java.io.*;
import java.net.Socket;
/**
 *
 * @author pranav
 */
public class Socket_programming {

    /**
     * @param args the command line arguments
     */


    private static DataOutputStream dataOutputStream = null;
    private static DataInputStream dataInputStream = null;

    public static void main(String[] args)
    {
        // Create Client Socket connect to port 900
        try (Socket socket = new Socket("localhost", 900)) {

        dataInputStream = new DataInputStream(
                socket.getInputStream());
            dataOutputStream = new DataOutputStream(
                socket.getOutputStream());
            System.out.println(
                "Sending the File to the Server");
        // Call SendFile Method
        sendFile(
                "C:\\Users\\User\\Downloads\\Practical list CN.pdf");

            dataInputStream.close();
            dataInputStream.close();
        }
        catch (Exception e) {
            e.printStackTrace();
        }
    }

    // sendFile function define here
```

```
        private static void sendFile(String path)
                throws Exception
        {
                int bytes = 0;
                // Open the File where he located in your pc
                File file = new File(path);
                FileInputStream fileInputStream
                        = new FileInputStream(file);

                // Here we send the File to Server
                dataOutputStream.writeLong(file.length());
                // Here we break file into chunks
                byte[] buffer = new byte[4 * 1024];
                while ((bytes = fileInputStream.read(buffer))
                        != -1) {
                // Send the file to Server Socket
                dataOutputStream.write(buffer, 0, bytes);
                        dataOutputStream.flush();
                }
           System.out.println("File sent successfully");
                // close the file here
                fileInputStream.close();
        }
}
```

- **Output**

```
run:
Server is Starting in Port 900
Connected
File is Received
BUILD SUCCESSFUL (total time: 7 seconds)
```

Fig. 14.2 File Received on server using port 900

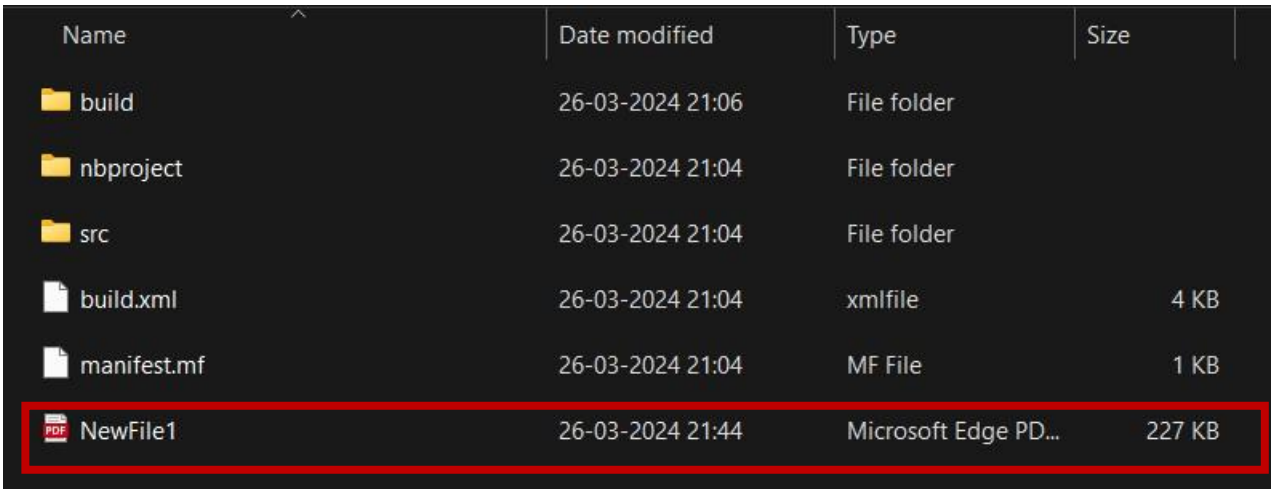

Fig. 14.3 NewFile1 received in server's folder