4. SOCKET PROGRAMMING – FILE TRANSFER

Write a program to create a server that listens to port 55 using stream sockets. Write a simple client program to connect to the server. The client should request for a text file and the server should return the file before terminating the connection

server.java

```
import java.io.*;
import java.net.*;
public class Server55{
   public static void main(String[] args){
      int port = 55;
      ServerSocket serverSocket = null;
      Socket clientSocket = null;
      BufferedReader clientInput = null;
      PrintWriter clientOutput = null;
      FileInputStream fileInputStream = null;
      BufferedOutputStream clientOutputStream = null;

      try{
         /* create a server socket that listens to port 55 */
         serverSocket = new ServerSocket(port);
         System.out.println("Server is listening to port " +port);
         clientSocket = serverSocket.accept();
         System.out.println("Client connected");
         clientInput = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));
         clientOutputStream = new
BufferedOutputStream(clientSocket.getOutputStream());
         String requestedFileName = clientInput.readLine();
         System.out.println("Client requested file:
"+requestedFileName);
```

```
            File file = new File(requestedFileName);

            if(file.exists()){
               System.out.println("File is there");
               fileInputStream = new FileInputStream(file);
               byte[] buffer = new byte[1024];
               int bytesRead;

               while((bytesRead = fileInputStream.read(buffer)) != -1
){              //opened while
                  clientOutputStream.write(buffer,0,bytesRead);
               }

               clientOutputStream.flush();
               System.out.println("File sent to client");
            } else {
               System.out.println("Requested File doesn't exist");
            }
         }
         catch(IOException e){
            e.printStackTrace();
         }
         finally{
            try{
               if(fileInputStream != null)
                  fileInputStream.close();
               if(clientInput != null)
                  clientInput.close();
               if(clientOutputStream != null)
                  clientOutputStream.close();
               if(clientSocket != null)
                  clientSocket.close();
               if(serverSocket != null)
```

```
                serverSocket.close();
            } catch(IOException e){
            e.printStackTrace();
            }
        }
    }
}
```

client.java

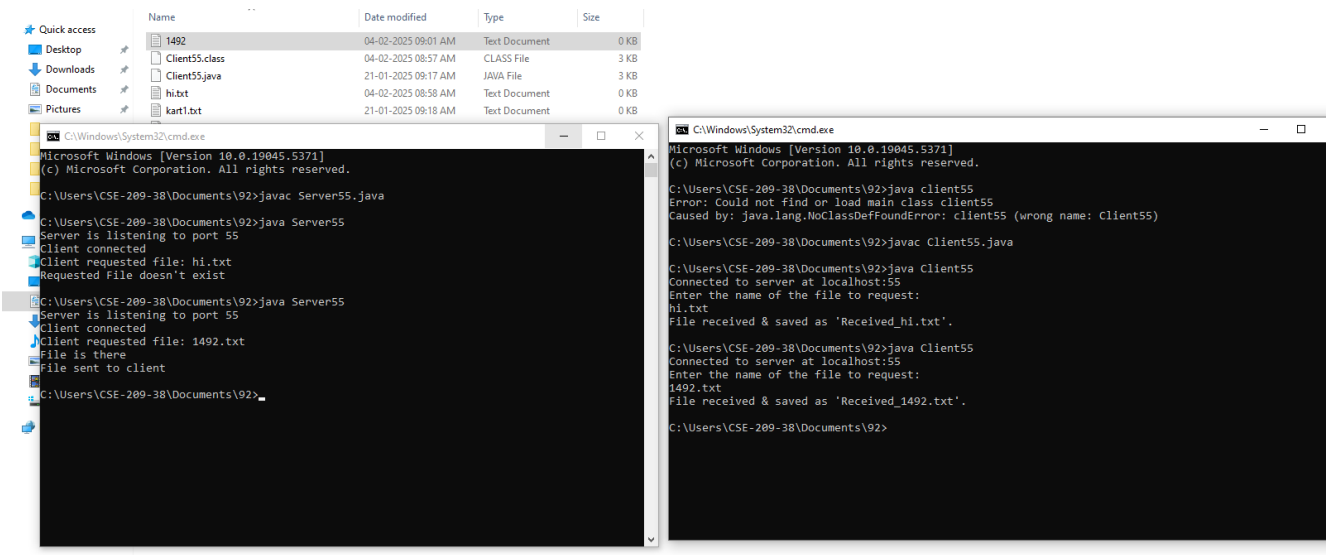
```
import java.io.*;
import java.net.*;
public class Client55 {
    public static void main(String[] args) {
        String serverAddress = "localhost";
        int port = 55;
        Socket socket = null;
        BufferedReader userInputReader = null;
        PrintWriter serverOutput = null;
        BufferedReader serverInput = null;
        FileOutputStream fileOutputStream = null;
         try {
            socket = new Socket(serverAddress, port);
            System.out.println("Connected to server at " +
serverAddress + ":" + port);
            serverOutput = new PrintWriter(socket.getOutputStream(),
true);
            serverInput = new BufferedReader(new
InputStreamReader(socket.getInputStream()));


            userInputReader = new BufferedReader(new
InputStreamReader(System.in));
            System.out.println("Enter the name of the file to
request: ");
            String fileName = userInputReader.readLine();
```

```
            serverOutput.println(fileName);
            fileOutputStream = new FileOutputStream("Received_" +
fileName);
            BufferedInputStream serverInputStream = new
BufferedInputStream(socket.getInputStream());
            byte[] buffer = new byte[1024];
            int bytesRead;
            while ((bytesRead = serverInputStream.read(buffer)) != -
1) {
                fileOutputStream.write(buffer, 0, bytesRead);
            }
            System.out.println("File received & saved as 'Received_"
+ fileName + "'.");

        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            try {
                // Close all resources in finally block
                if (userInputReader != null)
                    userInputReader.close();
                if (serverOutput != null)
                    serverOutput.close();
                if (serverInput != null)
                    serverInput.close();
                if (fileOutputStream != null)
                    fileOutputStream.close();
                if (socket != null)
                    socket.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
```

}


Quick access
Desktop
Downloads
Documents
Pictures
Name
Date modified
Type
Size
1492
04-02-2025 09:01 AM
Text Document
0 KB
Client55.class
04-02-2025 08:57 AM
CLASS File
3 KB
Client55.java
21-01-2025 09:17 AM
JAVA File
3 KB
hi.txt
04-02-2025 08:58 AM
Text Document
0 KB
kart1.txt
21-01-2025 09:18 AM
Text Document
0 KB
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.
C:\Users\CSE-209-38\Documents\92>javac Server55.java
C:\Users\CSE-209-38\Documents\92>java Server55
Server is listening to port 55
Client connected
Client requested file: hi.txt
Requested File doesn't exist
C:\Users\CSE-209-38\Documents\92>java Server55
Server is listening to port 55
Client connected
Client requested file: 1492.txt
File is there
File sent to client
C:\Users\CSE-209-38\Documents\92>
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.
C:\Users\CSE-209-38\Documents\92>java client55
Error: Could not find or load main class client55
Caused by: java.lang.NoClassDefFoundError: client55 (wrong name: Client55)
C:\Users\CSE-209-38\Documents\92>javac Client55.java
C:\Users\CSE-209-38\Documents\92>java Client55
Connected to server at localhost:55
Enter the name of the file to request:
hi.txt
File received & saved as 'Received_hi.txt'.
C:\Users\CSE-209-38\Documents\92>java Client55
Connected to server at localhost:55
Enter the name of the file to request:
1492.txt
File received & saved as 'Received_1492.txt'.
C:\Users\CSE-209-38\Documents\92>

server.py

```
import socket
import os

PORT = 55

server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind(("localhost", PORT))
server_socket.listen(1)
print(f"Server listening on port {PORT}...")

client_socket, addr = server_socket.accept()
print(f"Client connected from {addr}")

# Receive file request
file_name = client_socket.recv(1024).decode()
print(f"Client requested: {file_name}")

# Check and send file
if os.path.exists(file_name):
    with open(file_name, "rb") as file:
        client_socket.sendall(file.read())
    print("File sent successfully.")
else:
    print("File not found.")

client_socket.close()
server_socket.close()
```

client.py

```
import socket

def main():
    server_address = "localhost"
    port = 55
    try:
        # Connect to the server
        client_socket = socket.socket(socket.AF_INET,
socket.SOCK_STREAM)
        client_socket.connect((server_address, port))
        print(f"Connected to server at {server_address}:{port}")
        file_name = input("Enter the name of the file to request: ")
        client_socket.sendall(file_name.encode())
        with open(f"Received_{file_name}", "wb") as file:
            while True:
                data = client_socket.recv(1024)
                if not data:
                    break  # Stop when no more data is received
                file.write(data)

        print(f"File received & saved as 'Received_{file_name}'.")

    except Exception as e:
        print("Error:", e)

    finally:
        client_socket.close()

if __name__ == "__main__":
    main()
```