Untitled-1

```java
import java.io.*;
import java.net.*;
import java.util.*;

public class FileServer {
    private static final String FILE_DIRECTORY = "server_files"; // Directory to store files

    public static void main(String[] args) {
        try {
            ServerSocket serverSocket = new ServerSocket(12345);
            System.out.println("File server started. Waiting for clients...");

            // Create the directory if it doesn't exist
            File directory = new File(FILE_DIRECTORY);
            if (!directory.exists()) {
                directory.mkdir();
            }

            while (true) {
                Socket clientSocket = serverSocket.accept();
                System.out.println("Client connected.");

                new ClientHandler(clientSocket).start();
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    private static class ClientHandler extends Thread {
        private Socket clientSocket;

        public ClientHandler(Socket socket) {
            this.clientSocket = socket;
        }

        public void run() {
            try (
                DataInputStream dis = new DataInputStream(clientSocket.getInputStream());
                DataOutputStream dos = new DataOutputStream(clientSocket.getOutputStream());
            ) {
                String command = dis.readUTF();

                switch (command) {
                    case "UPLOAD":
                        uploadFile(dis, dos);
                        break;
                    case "DOWNLOAD":
                        downloadFile(dis, dos);
                        break;
                    case "LIST":
                        listFiles(dos);
                        break;
                    default:
                        dos.writeUTF("Invalid command.");
                }
            } catch (IOException e) {
                e.printStackTrace();
            }
        }

        private void uploadFile(DataInputStream dis, DataOutputStream dos) throws IOException {
            String fileName = dis.readUTF();
            long fileSize = dis.readLong();
            File file = new File(FILE_DIRECTORY + "/" + fileName);

            try (FileOutputStream fos = new FileOutputStream(file)) {
                byte[] buffer = new byte[4096];
                int read;
                long totalRead = 0;
                while ((read = dis.read(buffer, 0, Math.min(buffer.length, (int) (fileSize -
totalRead)))) > 0) {
                    fos.write(buffer, 0, read);
                    totalRead += read;
                }
                System.out.println("File " + fileName + " uploaded successfully.");
                dos.writeUTF("File uploaded successfully.");
            }
        }

        private void downloadFile(DataInputStream dis, DataOutputStream dos) throws IOException
{
            String fileName = dis.readUTF();
            File file = new File(FILE_DIRECTORY + "/" + fileName);

            if (file.exists()) {
                dos.writeUTF("OK");
                dos.writeLong(file.length());

                try (FileInputStream fis = new FileInputStream(file)) {
                    byte[] buffer = new byte[4096];
                    int read;
                    while ((read = fis.read(buffer)) > 0) {
                        dos.write(buffer, 0, read);
                    }
                }
                System.out.println("File " + fileName + " downloaded successfully.");
            } else {
                dos.writeUTF("File not found.");
            }
        }

        private void listFiles(DataOutputStream dos) throws IOException {
            File directory = new File(FILE_DIRECTORY);
            String[] files = directory.list();

            if (files != null && files.length > 0) {
                dos.writeUTF("OK");
                dos.writeInt(files.length);
                for (String file : files) {
                    dos.writeUTF(file);
                }
            } else {
                dos.writeUTF("No files available.");
            }
        }
    }
}

```

Untitled-2

```java
import java.io.*;
import java.net.*;

public class FileClient {
    public static void main(String[] args) {
        try (Socket socket = new Socket("localhost", 12345);
             DataInputStream dis = new DataInputStream(socket.getInputStream());
             DataOutputStream dos = new DataOutputStream(socket.getOutputStream());
             BufferedReader br = new BufferedReader(new InputStreamReader(System.in))) {

            System.out.println("Connected to file server.");

            while (true) {
                System.out.println("\nEnter a command (UPLOAD, DOWNLOAD, LIST, EXIT):");
                String command = br.readLine();

                if ("EXIT".equalsIgnoreCase(command)) {
                    break;
                }

                dos.writeUTF(command);

                switch (command.toUpperCase()) {
                    case "UPLOAD":
                        uploadFile(dis, dos, br);
                        break;
                    case "DOWNLOAD":
                        downloadFile(dis, dos, br);
                        break;
                    case "LIST":
                        listFiles(dis);
                        break;
                    default:
                        System.out.println("Invalid command.");
                }
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    private static void uploadFile(DataInputStream dis, DataOutputStream dos, BufferedReader br) throws IOException {
        System.out.print("Enter the file path to upload: ");
        String filePath = br.readLine();
        File file = new File(filePath);

        if (file.exists() && !file.isDirectory()) {
            dos.writeUTF(file.getName());
            dos.writeLong(file.length());

            try (FileInputStream fis = new FileInputStream(file)) {
                byte[] buffer = new byte[4096];
                int read;
                while ((read = fis.read(buffer)) > 0) {
                    dos.write(buffer, 0, read);
                }
            }
            System.out.println("Server response: " + dis.readUTF());
        } else {
            System.out.println("File not found.");
        }
    }

    private static void downloadFile(DataInputStream dis, DataOutputStream dos, BufferedReader br) throws IOException {
        System.out.print("Enter the file name to download: ");
        String fileName = br.readLine();
        dos.writeUTF(fileName);

        String response = dis.readUTF();
        if ("OK".equals(response)) {
            long fileSize = dis.readLong();
            File file = new File("client_files/" + fileName);

            try (FileOutputStream fos = new FileOutputStream(file)) {
                byte[] buffer = new byte[4096];
                int read;
                long totalRead = 0;
                while ((read = dis.read(buffer, 0, Math.min(buffer.length, (int) (fileSize - totalRead)))) > 0) {
                    fos.write(buffer, 0, read);
                    totalRead += read;
                }
                System.out.println("File downloaded successfully.");
            }
        } else {
            System.out.println("Server response: " + response);
        }
    }

    private static void listFiles(DataInputStream dis) throws IOException {
        String response = dis.readUTF();
        if ("OK".equals(response)) {
            int fileCount = dis.readInt();
            System.out.println("Files available on server:");
            for (int i = 0; i < fileCount; i++) {
                System.out.println("- " + dis.readUTF());
            }
        } else {
            System.out.println("Server response: " + response);
        }
    }
}

```