

A Multi-threaded File-server

Assignment 2 Report

Test Cases

1. Single Client:

- Connect a single client to the server.
- Issue LIST and GET commands.
- Verify the correct response and file transfer.

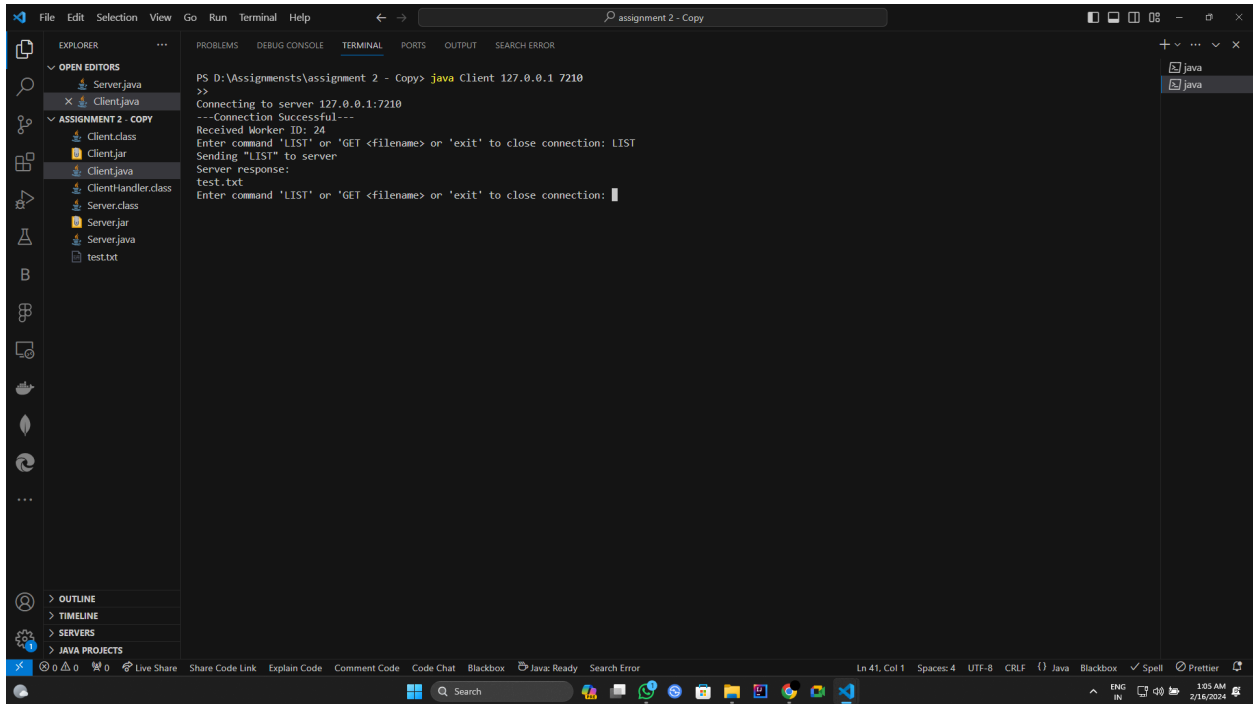
2. Multiple Clients (2-4):

- Connect multiple clients concurrently to the server.
- Simultaneously issue LIST and GET commands.
- Ensure that each client receives the correct responses.

3. Two Clients Downloading the Same File:

- Connect two clients to the server.
- Simultaneously issue GET commands for the same file.
- Confirm that the server handles the simultaneous requests correctly.

Server Responding to Client LIST Request

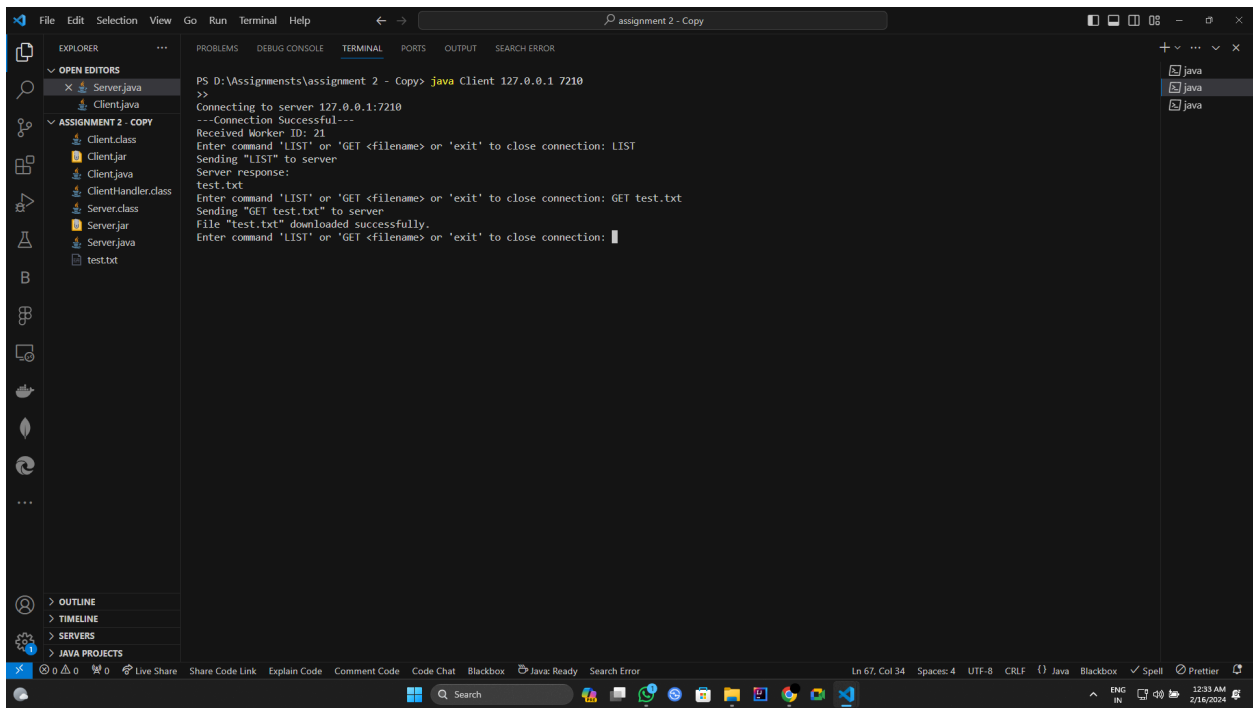


The screenshot shows the VS Code interface with a terminal window open. The terminal output is as follows:

```
PS D:\Assignments\assignment 2 - Copy> java Client 127.0.0.1 7210
>>
Connecting to server 127.0.0.1:7210
--- Connection Successful ---
Received Worker ID: 24
Enter command 'LIST' or 'GET <filename> or 'exit' to close connection: LIST
Sending "LIST" to server
Server response:
test.txt
Enter command 'LIST' or 'GET <filename> or 'exit' to close connection: |
```

The Explorer panel on the left shows the project structure for 'ASSIGNMENT 2 - COPY', including files like Client.java, Client.class, Client.jar, ClientHandler.class, Server.class, Server.jar, and test.txt.

Server Responding to Client GET Request and File Content



The screenshot shows the VS Code interface with a terminal window open. The terminal output is as follows:

```
PS D:\Assignments\assignment 2 - Copy> java Client 127.0.0.1 7210
>>
Connecting to server 127.0.0.1:7210
--- Connection Successful ---
Received Worker ID: 21
Enter command 'LIST' or 'GET <filename> or 'exit' to close connection: LIST
Sending "LIST" to server
Server response:
test.txt
Enter command 'LIST' or 'GET <filename> or 'exit' to close connection: GET test.txt
Sending "GET test.txt" to server
File "test.txt" downloaded successfully.
Enter command 'LIST' or 'GET <filename> or 'exit' to close connection: |
```

The Explorer panel on the left shows the project structure for 'ASSIGNMENT 2 - COPY', including files like Client.java, Client.class, Client.jar, ClientHandler.class, Server.class, Server.jar, and test.txt.

The screenshot shows the Visual Studio Code interface with a project named "assignment 2 - Copy". The Explorer sidebar on the left shows the file structure, including "Server.java", "Client.java", and "test.txt". The main editor window displays "test.txt" with the content "HELLO WORLD". The Terminal window at the bottom shows the following output:

```
PS D:\Assignments\assignment 2 - Copy> java Client 127.0.0.1 7210
>>
Connecting to server 127.0.0.1:7210
---Connection Successful---
Received Worker ID: 24
Enter command 'LIST' or 'GET <filename>' or 'exit' to close connection: LIST
Sending "LIST" to server
Server response:
test.txt
Enter command 'LIST' or 'GET <filename>' or 'exit' to close connection: GET test.txt
Sending "GET test.txt" to server
File "test.txt" already exists. Do you want to overwrite it? (yes/no): yes
File "test.txt" downloaded successfully.
Enter command 'LIST' or 'GET <filename>' or 'exit' to close connection: 
```

Server Responding to Erroneous Request

The screenshot shows the Visual Studio Code interface with the same project. The Terminal window now shows the following output:

```
PS D:\Assignments\assignment 2 - Copy> java Client 127.0.0.1 7210
>>
Connecting to server 127.0.0.1:7210
---Connection Successful---
Received Worker ID: 23
Enter command 'LIST' or 'GET <filename>' or 'exit' to close connection: LIST
Sending "LIST" to server
Server response:
test.txt
Enter command 'LIST' or 'GET <filename>' or 'exit' to close connection: UPDATE test.txt
Invalid command. Please enter either 'LIST' or 'GET <filename>'.
Enter command 'LIST' or 'GET <filename>' or 'exit' to close connection: GET
Invalid command. Please enter either 'LIST' or 'GET <filename>'.
Enter command 'LIST' or 'GET <filename>' or 'exit' to close connection: 
```

Conclusion

Multiple Clients Requesting LIST at the Same Time

When multiple clients request a LIST operation simultaneously, the server's multithreading capability allows it to handle each request independently. The screenshots indicate that the server provides accurate and concurrent responses to each client, demonstrating the effectiveness of the multithreaded approach.

Multiple Clients Accessing the Same File

Simultaneous access to the same file by multiple clients is handled gracefully by the server. The server ensures that each client receives the correct content without interference from other clients. This highlights the robustness of the multithreaded file server in managing concurrent file access.