

Write a client application, which will accept a file name from the user. This file name must be passed to a server application (which you must also write). The server application must read the file name from the client application and then endeavor to locate the file in question. If the server can locate the file, it sends a status code of “200” to the client, followed by the contents of the file. The file is then to be displayed on the clients machine. If the server application cannot locate the file, the server must send back a status code of “404” to the client. The client must then display a suitable error message to the user.

The responsibilities of the **client** are as follows:

1. Accept a value from the user. This value represents a file name.
2. This value must be passed to the server as a **String**.
3. Receive an **int** from the server. This value represents a status code. A status code of 200 means that the server has successfully located the file while a status code of 404 means that the server has been unable to locate the file.
4. If a status code of 200 has been returned, the client must display the contents of the file using a suitable GUI component.
5. If a status code of 404 has been returned, the client must display an error message using a suitable GUI component.

The responsibilities of the **server** are as follows:

1. Accept a value from the client as a **String**. This value represents a file name.
2. Attempt to locate the requested file.
3. If the file can be located, return a status code of 200 to the client along with the contents of the file.
4. If the file cannot be located, return a status code of 404 to the client.

Note 1: *The server must be multi-threaded.*

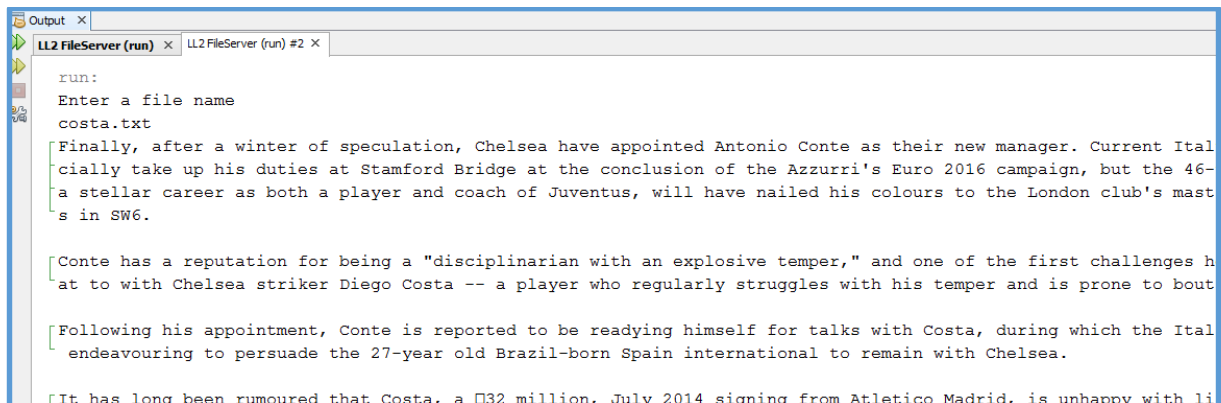
Note 2: *All necessary exceptions must be caught and declared.*

Note 3: *There is no input/output from/to the user in the server.*

Note 4: *Assume that the server will only search its own working directory for the requested file.*

Note 5: *Consider using a `BufferedWriter` and a `BufferedReader` to send/receive the file contents. It should help speed up your application.*

A sample run (for the client) would be as follows:



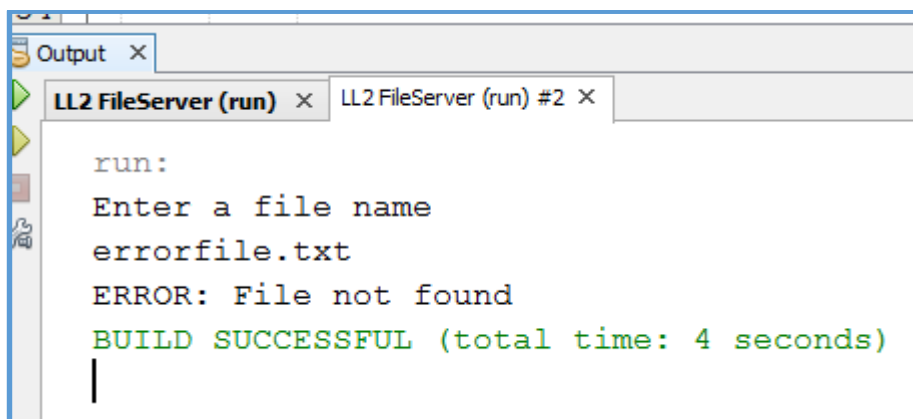
The screenshot shows a window titled "LL2 FileServer (run)" with a tab labeled "LL2 FileServer (run) #2". The output text is as follows:

```
run:
Enter a file name
costa.txt
[Finally, after a winter of speculation, Chelsea have appointed Antonio Conte as their new manager. Current Ital
cially take up his duties at Stamford Bridge at the conclusion of the Azzurri's Euro 2016 campaign, but the 46-
a stellar career as both a player and coach of Juventus, will have nailed his colours to the London club's mast
s in SW6.

[Conte has a reputation for being a "disciplinarian with an explosive temper," and one of the first challenges h
at to with Chelsea striker Diego Costa -- a player who regularly struggles with his temper and is prone to bout

[Following his appointment, Conte is reported to be readying himself for talks with Costa, during which the Ital
endeavouring to persuade the 27-year old Brazil-born Spain international to remain with Chelsea.

[It has long been rumoured that Costa, a £32 million, July 2014 signing from Atletico Madrid, is unhappy with li
```



The screenshot shows a window titled "LL2 FileServer (run)" with a tab labeled "LL2 FileServer (run) #2". The output text is as follows:

```
run:
Enter a file name
errorfile.txt
ERROR: File not found
BUILD SUCCESSFUL (total time: 4 seconds)
|
```