

Question

Indicate to Go TCP Server that Java TCP Client Has Finished Writing to the Stream

Description

My problem is that the server keeps waiting for the client to write its message even after all bytes of the message have been written to the stream. This seems to be because the `ioutil.ReadAll` function is waiting for an `io.EOF` error as the signal to stop reading from the stream. How can I tell Go that I am done writing to the TCP stream from Java? I can change both the Java TCP Client and Golang TCP Server codes, if that helps.

```
func writeToConnection(connection *net.TCPConn, tcpCommand structs.TCPCommand) error {
    messageBytes, err := json.Marshal(tcpCommand)
    if err != nil {
        err = merry.Wrap(err)
        return err
    }
    _, err = connection.Write(messageBytes)
    if err != nil {
        err = merry.Wrap(err)
        return err
    }
    err = connection.CloseWrite()
    if err != nil {
        err = merry.Wrap(err)
        return err
    }
    return nil
}
```

Skills Required :

JAVA
GO
TCP
CLIENT-SERVER
EOF