

```

# Lab 5 - network (socket), multi-threading, system
# Names: Trien Bang Huynh and Marcel Gunadi
# client.py

import socket
import pickle
import sys

HOST = '127.0.0.1'
PORT = int(sys.argv[1])

def menu():
    """
    A function which processes user input and return objects for sending to the
    server
    """
    print("Please select one of following options")
    print("g. Get current directory")
    print("c. Change to a new directory")
    print("l. List current directory")
    print("f. Create a new file ")
    print("q. Quit")

    userInput = input("Enter your choice: ").strip().lower()
    while userInput != 'g' and userInput != 'c' and userInput != 'l' \
        and userInput != 'f' and userInput != 'q':
        userInput = input("Invalid input. Enter your choice again: ")
        userInput = userInput.strip().lower()

    if userInput == 'g':
        return (userInput, currentDir)
    if userInput == 'c':
        newDir = input("Enter path, starting from current directory: ")
        return (userInput, newDir, currentDir)
    elif userInput == 'l':
        return (userInput, currentDir)
    elif userInput == 'f':
        newFile = input("Enter filename: ")
        return (userInput, newFile, currentDir)
    else:
        return (userInput, None)

with socket.socket() as s:
    s.settimeout(5) # set 5s time out connection to the server for a client
    try:
        s.connect((HOST, PORT))
        currentDir = pickle.loads(s.recv(1024))
        print("Client connect to:", HOST, "port:", PORT, '\n')
    except socket.timeout:
        print("Connection to server failed")
        raise SystemExit
    except ConnectionRefusedError as e:
        print("Connection refused. Check your port number or local host")
        raise SystemExit

    mesg = menu()
    s.send(pickle.dumps(mesg))

```

```
while mesg[0] != 'q':
    fromServer = pickle.loads(s.recv(1024))

    if mesg[0] == 'c':
        print("Received from server:", fromServer[0])
        currentDir = fromServer[1]
    else:
        print("Received from server:", fromServer)
    mesg = menu()
    s.send(pickle.dumps(mesg))
```