

حل الوظيفة



كلية الهندسة المعلوماتية

السنة: الرابعة

المادة: نظم موزعة

شرح الكود :

1- كلاس الزبون (client class)

```
public class Client {
```

```
public static void main(String arg[]) throws IOException {
```

```
String choice, filename;
```

المتحول choice للدلالة على نوع العملية (PUT, GET) و
filename هو اسم الملف (مع اللاحقة .txt)

```
String res;
```

Res المتحول الذي يمثل استجابة السيرفر

```
FileReader fileReader;
```

```
FileWriter fileWriter;
```

```
try (Socket clientSocket = new Socket("localhost", 6689)) {
```

```
System.out.println("Attempt to connect to server on port " + 6689);
```

أنشأنا socket ال host فيها هو localhost وال port=6689 مثلاً

```
DataOutputStream outToServer = new  
DataOutputStream(clientSocket.getOutputStream());
```

من خلال outToServer سنرسل إلى السيرفر

```
BufferedReader inFromServer = new BufferedReader(new  
InputStreamReader(clientSocket.getInputStream()));
```

من خلال inFromServer ستصلنا استجابة من السيرفر

```
BufferedReader br = new BufferedReader(new  
InputStreamReader(System.in));
```

br متحول من أجل إدخال سلسلة من لوحة المفاتيح

```
System.out.println("Enter PUT for upload file to server, GET for get a  
file from server OR EXIT to exit");
```

جملة توضيحية

```
choice = br.readLine();
```

إدخال قيمة choice

```
while (true) {
```

```
if (choice.equals ("PUT")) { // في حال أراد المستخدم رفع ملف
```

```
System.out.println("Enter file name you want to upload");
```

```
filename = br.readLine();// إدخال اسم الملف (مع اللاحقة)
```

```
// قراءة محتويات الملف من أجل إرسالها للسيرفر
```

```
fileReader = new FileReader("E:\\client\\" + filename);
```

```
// من أجل قراءة محتويات الملف btr المتحول
```

```
BufferedReader brc = new BufferedReader(fileReader);
```

```
String content = new String();// سنخزن فيها محتويات الملف
```

```
String line = null;
//هنا سنمر على عناصر الملف سطر سطر ومن ثم إضافته للمتحول
while ((line = br.readLine()) != null) {
    content += line;
}
//إغلاق الملف من أجل حفظ التعديلات
fileReader.close();
//إرسال المتحول choice إلى السيرفر لكي يعرف ماذا اختار المستخدم
outToServer.writeBytes(choice + '\n');

//إرسال اسم الملف إلى السيرفر
outToServer.writeBytes(filename + '\n');

//إرسال محتوى الملف إلى السيرفر
outToServer.writeBytes(content + '\n');

//المتحول يحوي استجابة السيرفر للطلب
res = inFromServer.readLine();
System.out.println(res);
}
//GET في حال اختار المستخدم تحميل ملف أي الخيار هو
else if (choice.equals ("GET")) {
    System.out.println("Enter file name you want to get it ");
    filename = br.readLine();//إدخال اسم الملف الذي سنحمله من السيرفر
```

أيضا نرسل choice واسم الملف للسيرفر لكي أحمله//

```
outToServer.writeBytes(choice + '\n');
```

```
outToServer.writeBytes(filename + '\n');
```

متحول يحوي استجابة السيرفر والتي هي محتويات الملف الذي طلبناه//

```
res = inFromServer.readLine();
```

الآن سوف نضع محتوى الملف الذي طلبناه في ملف ننشئه نحن وله نفس الاسم //

ولكن في ضمن مجلد الزبون

```
fileWriter = new FileWriter("E:\\client\\" + filename);
```

```
String temp = new String();
```

```
for (int i = 0; i < res.length(); i++) {
```

```
    temp += res.charAt(i);
```

```
}
```

```
fileWriter.write(temp);
```

```
fileWriter.close();
```

```
}
```

يتم الخروج EXIT في حال أدخل المستخدم كلمة //

```
if (choice.equals ("EXIT")) {
```

```
    break;
```

```
}
```

وإلا نطبع رسالة خطأ//

```
System.out.println("Invalid choice, valid choices are
```

```
(GET,PUT,EXIT)");
```

```
}
```

System.out.println("Enter PUT for upload file to server, GET
for get a file from server OR EXIT to exit");

```
//choice ونكرر العمل السابق ونطلب من المستخدم إدخال قيمة choice
choice = br.readLine();
}
}
}
```

2- كود السيرفر:

```
public class Server {

    public static void main(String arg[]) throws IOException {

        String clientchoice;//متحول يمثل نوع العملية التي يريد المستخدم
        String clientFileName;//اسم الملف الذي سيرسله المستخدم

        FileWriter fileWriter;//للكتابة في ملف
        FileReader fileReader;//للقراءة من ملف

        //رسالة تشير إلى جاهزية السيرفر
        System.out.println("server is up and running");

        //يجب ملاحظة أن ال port يجب أن يكون نفسه عند السيرفر والزيوم
        ServerSocket welcomeSocket = new ServerSocket(6689);

        Socket connectionSocket = welcomeSocket.accept();
```

```
while (true) {
```

// هذا المتحول يقرأ البيانات المرسله من الزبون

```
BufferedReader inFromClient = new BufferedReader(new  
InputStreamReader(connectionSocket.getInputStream()));
```

// هذه المتحول يستخدم لإرسال البيانات للزبون

```
DataOutputStream outToClient = new  
DataOutputStream(connectionSocket.getOutputStream());
```

// استقبال نوع العملية من قبل الزبون وتخزينها في هذا المتحول

```
clientchoice = inFromClient.readLine();
```

// استقبال اسم الملف من الزبون ووضعها في المتحول

```
clientFileName = inFromClient.readLine();
```

//PUT اذا كانت العملية هي

```
if (clientchoice.equals ("PUT")) {
```

//content تلقي محتوى الملف الذي سيرفعه الزبون ووضعها بمتحول

```
String content = inFromClient.readLine();
```

```
fileWriter = new FileWriter("E:\\server\\" + clientFileName);
```

// كتابة المحتوى الذي أرسله الزبون إلى ملف موجود ضمن مجلد السيرفر وطبعاً

يحمل نفس الاسم

```
String temp = new String();
for (int i = 0; i < content.length(); i++) {
    temp += content.charAt(i);
}
fileWriter.write(temp);
fileWriter.close();//نقل الملف لحفظ التغييرات

//إرسال استجابته للزبون (ليست أساسية فقط للتأكد)
outToClient.writeBytes("uploaded successfully\n");
}
else //إذا كان الزبون قد اختار تحميل ملف
if (clientchoice.equals ("GET")) {

    //متحول يحوي محتوى الملف الذي سنرسله للزبون
    String content = new String();
    //سنقوم بقراءة محتوى الملف ووضعه في المتحول السابق
    FileReader = new FileReader("E:\\server\\" + clientFileName);
    BufferedReader brf = new BufferedReader(fileReader);

    String line = null;

    //قراءة المحتوى سطر سطر ووضعه بالمتحول content
    while ((line = brf.readLine()) != null) {
        content += line;
    }

    fileReader.close();
```

إرسال المحتوى للزبون وكما شرحنا سابقا يقوم بحفظ المحتوى بملف ضمن مجلد //client

```
outToClient.writeBytes(content + "\n");
```

```
}
```

```
}
```

```
}
```

```
}
```

- في المثال فرضنا أن ملفات الزبون موجودة ضمن مجلد اسمه client موجود على

القرص E

- وملفات السيرفر موجودة على نفس القرص ولكن ضمن مجلد اسمه server