**武汉大学计算机学院**

**本科生实验报告**

**计算机网络课程设计**

专 业 名 称 ：计算机科学与技术

课 程 名 称 ：计算机网络课程设计

团 队 名 称 ：xxx

指 导 教 师 ：xx

团 队 成 员 ：xxx

二○二一年八月

**郑 重 声 明**

本团队呈交的实验报告，是在指导老师的指导下，独立进行实验工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本实验报告不包含他人享有著作权的内容。对本实验报告做出贡献的其他个人和集体，均已在文中以明确的方式标明。本实验报告的知识产权归属于培养单位。

团队成员签名： 日期： 2021.8.4

摘 要

本实验的实验目的是通过socket网络通信编程进一步理解计算机网络协议的原理与应用，以及如何编写网络应用。

实验设计主要在windows11下开发，开发工具为Visual Studio2019，开发语言为C#，使用winform框架以及.netframework 4.7.2。

实验内容主要是实现ftp客户端，主要实现的功能包括上传和下载功能，以及图形化界面。

**关键词：**ftp客户端；上传；下载；图形化界面；C#；winfrom

目录

[1 实验目的和意义 6](#_Toc79063746)

[1.1 实验目的 6](#_Toc79063747)

[1.2 实验意义 6](#_Toc79063748)

[2 实验设计 7](#_Toc79063749)

[2.1 概述 7](#_Toc79063750)

[2.1.1 开发环境 7](#_Toc79063751)

[2.1.2 实验内容 7](#_Toc79063752)

[2.2 实验原理 7](#_Toc79063753)

[2.2.1 FTP协议主要原理与应用 7](#_Toc79063754)

[2.2.2 C#语言与winform 9](#_Toc79063755)

[2.3 实验方案 10](#_Toc79063756)

[2.3.1 界面设计 10](#_Toc79063757)

[2.3.2 全局变量与全局函数 11](#_Toc79063758)

[2.3.3 具体功能实现大致流程与事件绑定 12](#_Toc79063759)

[结论 16](#_Toc79063760)

[1. 程序主要界面及结果 16](#_Toc79063761)

[2. 程序源程序 23](#_Toc79063762)

[参考文献 47](#_Toc79063763)

# 1 实验目的和意义

## 1.1 实验目的

本实验是使学生熟悉socket网络通信编程以及协议如何工作与应用、掌握网络应用的编程方法，将具体的网络应用通过某一编程语言实现出来并逐步深入了解协议的主要内容与功能，以及如何对于网络应用进行调试与复写。

## 1.2 实验意义

该实验是理论知识和动手能力的综合体现。通过本实验，可以掌握socket网络通信编程方法、ftp协议的命令格式、开发平台的使用、如何进行调试与测试、c#语言调用socket包。

# 2 实验设计

## 2.1 概述

### 2.1.1 开发环境

操作系统：windows11

开发工具：Visual Studio2019

开发语言：C#

开发框架：.netframework4.7.2

界面框架：winform

### 2.1.2 实验内容

实现ftp客户端，主要实现的功能包括上传和下载功能，以及图形化界面。

## 2.2 实验原理

### 2.2.1 FTP协议主要原理与应用

FTP（File Transfer Protocol，文件传输协议） 是 TCP/IP 协议组中的协议之一。FTP协议包括两个组成部分，其一为FTP服务器，其二为FTP客户端。其中FTP服务器用来存储文件，用户可以使用FTP客户端通过FTP协议访问位于FTP服务器上的资源。在开发网站的时候，通常利用FTP协议把网页或程序传到Web服务器上。此外，由于FTP传输效率非常高，在网络上传输大的文件时，一般也采用该协议。

默认情况下FTP协议使用TCP端口中的 20和21这两个端口，其中20用于传输数据，21用于传输控制信息。但是，是否使用20作为传输数据的端口与FTP使用的传输模式有关，如果采用主动模式，那么数据传输端口就是20；如果采用被动模式，则具体最终使用哪个端口要服务器端和客户端协商决定。本次实验采用随机端口进行数据传输。

本次实验中使用的FTP协议命令列表如下：

|  |  |
| --- | --- |
| 命令 | 描述 |
| USER <用户名> | 登录FTP的用户名 |
| PASS <密码> | 登录FTP的密码 |
| QUIT | 断开连接 |
| PASV | 进入被动模式，返回server的数据端口，等待client连接 |
| ABOR | 断开数据端口的连接 |
| LIST | 查看服务器文件（从数据端口返回结果） |
| STOR <文件名> | 请求上传 |
| RETR <文件名> | 请求下载 |
| CWD | 改变工作目录 |

FTP服务端与客户端交互示意图如下：

图示

描述已自动生成

### 2.2.2 C#语言与winform

C#是由C和C++衍生出来的一种安全的、稳定的、简单的、优雅的面向对象编程语言。它在继承C和C++强大功能的同时去掉了一些它们的复杂特性（例如没有宏以及不允许多重继承）。C#综合了VB简单的可视化操作和C++的高运行效率，以其强大的操作能力、优雅的语法风格、创新的语言特性和便捷的面向组件编程的支持成为.net开发的首选语言。[1]

C#是面向对象的编程语言。它使得程序员可以快速地编写各种基于MICROSOFT .NET平台的应用程序，MICROSOFT .NET提供了一系列的工具和服务来最大程度地开发利用计算与通讯领域。

WinForm是.Net开发平台中对Windows Form的一种称谓。

它具有以下优点：

1.功能强大

Windows窗体可用于设计窗体和可视控件，以创建丰富的基于Windows的应用程序。

2.操作方便

新的数据提供程序管理：数据提供程序管理提供易于连接OLEDB和ODBC数据源的数据控件，包括Microsoft SQL Server、Microsoft Access、Jet、DB2以及Oracle等。

3.使用安全

Windows窗体充分利用公共语言运行库的安全特性。这就意味着，一切都可以通过Windows窗体来实现，包括在浏览器中运行的不可信控件和用户硬盘上安装的完全可信的应用程序。

## 2.3 实验方案

### 2.3.1 界面设计

图形用户界面, 应用程序

描述已自动生成

主要的控件有：

Button（3个）：button1（上级目录）、btn\_conn（连接）、btnGetPath（获取）

Textbox（5个）：tb\_IP（ftp地址）、tb\_port（端口）、tb\_username（用户名）、tb\_password（密码）、txtpath（文件目录）

ListBox（2个）：lsb\_server、lsb\_status（状态信息）

Treeview（1个）：tvDir

Listview（1个）：lvList（4个列表头：文件名、扩展名、文件大小、文件全路径）

contextMenuStrip（2个）：contextMenuStrip1（下载）、contextMenuStrip2（上传）

### 2.3.2 全局变量与全局函数

全局变量：

private TcpClient cmdServer;

private TcpClient dataServer;

private NetworkStream cmdStrmWtr;

private StreamReader cmdStrmRdr;

private NetworkStream dataStrmWtr;

private StreamReader dataStrmRdr;

private String cmdData;

private byte[] szData;

private const String CRLF = "\r\n";

对于前两个TcpClient类型的变量来说，它们是与服务器进行通信的变量；对于NetworkStream类型的cmdStrmWtr和dataStrmWtr，它们分别是发送命令和数据的变量；对于StreamReader类型的cmdStrmRdr和dataStrmRdr来说，它们分别是读取命令和数据的变量；对于string类型的cmdData，它是ftp命令的变量；对于szData来说，它是经过转码后的命令的变量，用于发送命令；对于CRLF来说，因为ftp需要结束，所以作为一种结束符加在发送的命令结尾。

全局函数：

1.private String getSatus()；//获取命令端口返回结果，并记录在lsb\_status上

2.private void openDataPort()；//进入被动模式，并初始化数据端口的输入输出流

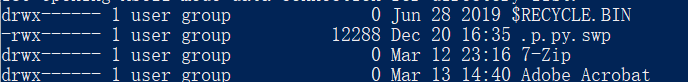
3.private void closeDataPort()；//断开数据端口的连接

4.private void freshFileBox\_Right()；//获得/刷新服务器文件列表

全局函数大致实现：

1. 读取cmdStrmRdr数据流，并将其加载到lsb\_status中。
2. 首先通过发送ftp命令——“PASV”（注意应该加上CRLF），将其转化为ASCII码。之后获得状态信息。通过状态信息获得端口号，将端口信息加载到状态列表中。然后使用ftp地址和端口号初始化TcpClient作为数据socket。最后通过dataStrmRdr来读取服务器发来的数据。
3. 发送“ABOR”FTP命令，流程如上所示。
4. 首先开启数据端口。之后发送“LIST”FTP命令（如上）。再将读取的数据流进行解码，并加载到lsb\_Server中。

发送回的数据流格式如下：



将其使用“ ”（空格）进行分割，最后一个元组为文件名称；第一个字符为文件类型，其中“d”为文件夹（目录），“-”为文件。

### 2.3.3 具体功能实现大致流程与事件绑定

#### 2.3.3.1 连接功能

首先通过tb\_IP和tb\_port的Text值来初始化一个socket（cmdServer）进行连接，将其读取的状态加入到状态列表中。之后主要是登录流程，首先发送“USER ”加上用户名——这一FTP命令；之后发送“PASS ”加上密码——这一FTP命令；如果服务器发来的状态信息为530，则代表用户名密码错误。

#### 2.3.3.2 断开连接功能

发送“QUIT”这一FTP命令，将返回的状态信息加入到状态列表中。

#### 2.3.3.3 上述两个功能的事件绑定

将它们绑定到btn\_conn按钮上，并与Click事件进行绑定；通过按钮的文本信息（连接和断开）来进行不同功能的调用。

#### 2.3.3.4 进入服务器目录功能

首先判断文件类型是否为目录，若是，则可发送“CWD ”加上要进入的目录——这一FTP命令来改变工作目录；若不是，则不操作。

这一功能将与lsb\_sever的MouseClick事件进行绑定。

#### 2.3.3.5 返回上级目录功能

与上述流程类似，将要进入的目录改为“..”即可。

并将此功能与button1按钮的Click事件进行绑定。

#### 2.3.3.6 右键服务器文件列表文件类型出现“下载”字样功能

读取选中项的文件类型，若为文件，则显示contextMenuStrip控件。

将其与lsb\_server的MouseUp事件进行绑定。

#### 2.3.3.7 下载功能

首先初始化一个文件全路径；开启数据连接；之后发送“RETR ”加上需要下载的文件名——这一FTP命令；最后创建文件流将服务器返回的dataStrmWtr写入到文件流中；关闭数据连接。

将之与 下载ToolStripMenuItem 控件的Click事件进行绑定。

#### 2.3.3.8 选择本地路径功能

使用FolderBrowserDialog类进行选择本地的路径，选择的结果显示在txtPath上。将其与txtPath控件的Click事件进行绑定。

#### 2.3.3.9 获取指定目录子目录和子文件功能

调用两个函数LoadData()和LoadFiles()。并将其与btnGetPath控件的Click事件进行绑定。

#### 2.3.3.10 加载树节点和listview控件的项功能（LoadData()）

函数声明：

void LoadData(string path, TreeNode parentNode)；

大致实现流程：

根据指定的path初始化一个DirectoryInfo类型的dir变量；调用dir变量的GetDirectories方法，得到目录列表dirs；遍历dirs，创建子节点；之后使用递归调用，不断加载子目录。

#### 2.3.3.11 加载文件功能

函数声明：

void LoadFiles(string path)；

大致实现流程：

根据给定的path初始化dir；之后获取子文件列表，调用dir的GetFiles方法；遍历子文件列表，加入listview中。

#### 2.3.3.12 上传功能

首先获取本地的文件，文件路径就是listview中选中项的某个子项；打开数据连接；发送“STOR ”加上文件路径——这一FTP命令；再初始化一个文件流，将其转码为ASCII码，使用dataStrmWtr进行发送这一串ASCII码。

将其与 上传ToolStripMenuItem 控件的Cilck事件进行绑定。

#### 2.3.3.13 选中listview功能

使用两个逻辑判断，即e.Button == MouseButtons.Right和lvList.SelectedIndices.Count!=0

后一个逻辑判断代表选中listview项数不为零。

# 结论

## 1. 程序主要界面及结果

FTP服务端设置如下（密码为123）：

图片包含 表格

描述已自动生成

服务端文件设置如下：

主目录有两个文件夹（test1和test2），一个文件test1.txt；其中test1子目录内有一个文件test2.txt。

客户端进行连接，假如密码错误，会出现

表格

描述已自动生成

密码正确时，如下图：

图形用户界面, 表格

描述已自动生成

点击目录类型文件，进入test1子目录，如下图：

图形用户界面, 表格

描述已自动生成

右键点击文件类型的文件选择下载，如下图：

电脑萤幕的截图

描述已自动生成

下载文件处于程序的文件夹下，如下图：

图形用户界面, 应用程序

描述已自动生成

图形用户界面, 文本, 应用程序

描述已自动生成

点击上级目录按钮即可进入上级目录，如下图：

图形用户界面, 表格

描述已自动生成

选择本地文件目录，如下图：

图形用户界面, 应用程序

描述已自动生成

进入一个测试文件夹，如下图：

图形用户界面, 表格

描述已自动生成

本地的文件如下图：

图形用户界面

描述已自动生成

右键上传，如下图：

图形用户界面, 应用程序, Word

描述已自动生成

上传结果如图：

图形用户界面, 表格

描述已自动生成

上传到服务端文件如下：

图形用户界面

中度可信度描述已自动生成

图片没有失真。

## 2. 程序源程序

Form1.cs代码如下：

1. **using** System;
2. **using** System.IO;
3. **using** System.Net.Sockets;
4. **using** System.Text.RegularExpressions;
5. **using** System.Windows.Forms;
6. **using** System.Configuration;
8. **namespace** WindowsFormsApp5
9. {
10. **public** partial **class** Form1 : Form
11. {
12. #region  Private variable
13. **private** TcpClient cmdServer;
14. **private** TcpClient dataServer;
15. **private** NetworkStream cmdStrmWtr;
16. **private** StreamReader cmdStrmRdr;
17. **private** NetworkStream dataStrmWtr;
18. **private** StreamReader dataStrmRdr;
19. **private** String cmdData;
20. **private** **byte**[] szData;
21. **private** **const** String CRLF = "\r\n";
22. #endregion

25. #region  Private Functions
27. /// <summary>
28. /// 获取命令端口返回结果，并记录在lsb\_status上
29. /// </summary>
30. **private** String getSatus()
31. {
33. String ret = cmdStrmRdr.ReadLine();
34. lsb\_status.Items.Add(ret);
35. lsb\_status.SelectedIndex = lsb\_status.Items.Count - 1;
36. **return** ret;
37. }
39. /// <summary>
40. /// 进入被动模式，并初始化数据端口的输入输出流
41. /// </summary>
42. **private** **void** openDataPort()
43. {
44. **string** retstr;
45. **string**[] retArray;
46. **int** dataPort;
48. // Start Passive Mode
49. cmdData = "PASV" + CRLF;
50. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
51. cmdStrmWtr.Write(szData, 0, szData.Length);
52. retstr = **this**.getSatus();
54. // Calculate data's port
55. retArray = Regex.Split(retstr, ",");
56. **if** (retArray[5][2] != ')') retstr = retArray[5].Substring(0, 3);
57. **else** retstr = retArray[5].Substring(0, 2);
58. dataPort = Convert.ToInt32(retArray[4]) \* 256 + Convert.ToInt32(retstr);
59. lsb\_status.Items.Add("Get dataPort=" + dataPort);
61. **string** IP = tb\_IP.Text;
63. //Connect to the dataPort
64. //dataServer = new TcpClient(tb\_IP.Text, dataPort);
65. dataServer = **new** TcpClient(IP, dataPort);
66. dataStrmRdr = **new** StreamReader(dataServer.GetStream());
67. dataStrmWtr = dataServer.GetStream();
68. }
70. /// <summary>
71. /// 断开数据端口的连接
72. /// </summary>
73. **private** **void** closeDataPort()
74. {
75. dataStrmRdr.Close();
76. dataStrmWtr.Close();
77. **this**.getSatus();
79. cmdData = "ABOR" + CRLF;
80. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
81. cmdStrmWtr.Write(szData, 0, szData.Length);
82. **this**.getSatus();
84. }
86. /// <summary>
87. /// 获得/刷新 右侧的服务器文件列表
88. /// </summary>
89. **private** **void** freshFileBox\_Right()
90. {
92. openDataPort();
94. **string** absFilePath;
96. //List
97. cmdData = "LIST" + CRLF;
98. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
99. cmdStrmWtr.Write(szData, 0, szData.Length);
100. **this**.getSatus();
102. lsb\_server.Items.Clear();
103. **while** ((absFilePath = dataStrmRdr.ReadLine()) != **null**)
104. {
105. **string** type;
106. **string** prefix;
107. **string**[] temp = Regex.Split(absFilePath, " ");
108. //string[] a = line.Split(new string[] { " " }, StringSplitOptions.RemoveEmptyEntries);
109. type = temp[0].Substring(0, 1);
110. **switch** (type)
111. {
112. **case** "d":
113. prefix = "[目录] ";
114. **break**;
115. **case** "-":
116. prefix = "[文件] ";
117. **break**;
118. **case** "l":
119. prefix="[连接] " ; // 符号链接
120. **break**;
122. **case** "b":
123. prefix = "[设备] ";
124. **break**;
125. **case** "c":
126. prefix = "[设备] ";
127. **break**;
128. **default**:
129. prefix = "[未知] ";
130. **break**;
131. }


135. lsb\_server.Items.Add(prefix+temp[temp.Length - 1]);
136. }
138. closeDataPort();
139. }
141. #endregion


145. **public** Form1()
146. {
147. InitializeComponent();
148. **int** i = **int**.Parse(ConfigurationManager.AppSettings["able"]);
149. **if** (i == 0)
150. {
151. //默认
152. ConfigurationManager.AppSettings["local"] = Environment.CurrentDirectory;
153. ConfigurationManager.AppSettings["able"] = "1";
155. }

158. //CpuMessage();
159. }



164. **private** **void** textBox3\_TextChanged(**object** sender, EventArgs e)
165. {
167. }

170. #region  Button:  Connect & Disconnect
172. **private** **void** btn\_conn\_Click(**object** sender, EventArgs e)
173. {
174. **if** (btn\_conn.Text == "连接")
175. {
176. Cursor cr = Cursor.Current;
177. Cursor.Current = Cursors.WaitCursor;
178. **try**
179. {
180. cmdServer = **new** TcpClient(tb\_IP.Text, Convert.ToInt32(tb\_port.Text));
181. lsb\_status.Items.Clear();
182. cmdStrmRdr = **new** StreamReader(cmdServer.GetStream());
183. cmdStrmWtr = cmdServer.GetStream();
184. **this**.getSatus();
186. **string** retstr;
188. //Login
189. cmdData = "USER " + tb\_username.Text + CRLF;
190. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
191. cmdStrmWtr.Write(szData, 0, szData.Length);
192. **this**.getSatus();
194. cmdData = "PASS " + tb\_password.Text + CRLF;
195. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
196. cmdStrmWtr.Write(szData, 0, szData.Length);
197. retstr = **this**.getSatus().Substring(0, 3);
198. **if** (Convert.ToInt32(retstr) == 530) **throw** **new** InvalidOperationException("帐号密码错误");
200. **this**.freshFileBox\_Right();
202. lb\_IP.Text = tb\_IP.Text + ":";
203. btn\_conn.Text = "断开";
204. //btn\_upload.Enabled = true;
205. //btn\_download.Enabled = true;
206. }
207. **catch** (InvalidOperationException err)
208. {
209. lsb\_status.Items.Add("ERROR: " + err.Message.ToString());
210. }
211. **finally**
212. {
213. Cursor.Current = cr;
214. }
215. }
216. **else**
217. {
218. Cursor cr = Cursor.Current;
219. Cursor.Current = Cursors.WaitCursor;
221. //Logout
223. cmdData = "QUIT" + CRLF;
224. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
225. cmdStrmWtr.Write(szData, 0, szData.Length);
226. **this**.getSatus();

229. cmdStrmWtr.Close();
230. cmdStrmRdr.Close();
232. lb\_IP.Text = "";
233. btn\_conn.Text = "连接";
234. lsb\_server.Items.Clear();
236. Cursor.Current = cr;
237. }
238. }
240. #endregion

243. /// <summary>
244. /// 进入服务器文件夹
245. /// </summary>
246. /// <param name="sender"></param>
247. /// <param name="e"></param>
248. **private** **void** lsb\_server\_MouseClick(**object** sender, MouseEventArgs e)
249. {
251. **string** retstr;
252. **string** type;
253. **int** index = lsb\_server.IndexFromPoint(e.X, e.Y);
254. lsb\_server.SelectedIndex = index;
255. **if** (lsb\_server.SelectedIndex != -1)
256. {
257. MessageBox.Show(lsb\_server.SelectedItem.ToString());
259. type = lsb\_server.SelectedItem.ToString().Substring(1, 2);
260. **if** (type == "目录")
261. {
262. cmdData = "CWD " + lsb\_server.SelectedItem.ToString().Substring(5) + CRLF;
263. lb\_IP.Text += "/" + lsb\_server.SelectedItem.ToString().Substring(5);
264. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
265. cmdStrmWtr.Write(szData, 0, szData.Length);
266. retstr = **this**.getSatus();
267. lsb\_status.Items.Add(retstr);
268. freshFileBox\_Right();
270. }
272. **else**
273. {
274. MessageBox.Show("目标不是目录，无法进入！");
276. }
277. }


281. }

284. /// <summary>
285. /// 返回上级目录
286. /// </summary>
287. /// <param name="sender"></param>
288. /// <param name="e"></param>
289. **private** **void** button1\_Click(**object** sender, EventArgs e)
290. {
291. **string** pa = lb\_IP.Text;
292. **string** retstr;
293. cmdData = "CWD .." + CRLF;
294. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
295. cmdStrmWtr.Write(szData, 0, szData.Length);
296. retstr = **this**.getSatus();
297. lsb\_status.Items.Add(retstr);
298. freshFileBox\_Right();
299. lb\_IP.Text = pa.Substring(0, pa.LastIndexOf("/"));

302. }
304. **private** **void** lsb\_server\_MouseUp(**object** sender, MouseEventArgs e)
305. {

308. **if** (e.Button == MouseButtons.Right)
309. {
310. lsb\_server.SelectedIndex = lsb\_server.IndexFromPoint(e.X, e.Y);

313. **string** abc = lsb\_server.SelectedItem.ToString();
315. **string** type = abc.Substring(1, 2);
316. **if** (type == "文件")
317. {
318. contextMenuStrip1.Show(Control.MousePosition.X, Control.MousePosition.Y);
319. }
321. }
322. }
324. **private** **void** 下载ToolStripMenuItem\_Click(**object** sender, EventArgs e)
325. {
326. //默认下载地址
327. **string** path=ConfigurationManager.AppSettings["local"];
329. **if** (path == "" || lsb\_server.SelectedIndex < 0)
330. {
331. MessageBox.Show("请选择目标文件和下载路径", "ERROR");
332. **return**;
333. }
335. Cursor cr = Cursor.Current;
336. Cursor.Current = Cursors.WaitCursor;
338. **string** fileName = Regex.Split(lsb\_server.Items[lsb\_server.SelectedIndex].ToString(), " ")[1];
339. **string** filePath = path + "\\" + fileName;
341. **this**.openDataPort();
343. cmdData = "RETR " + fileName + CRLF;
344. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
345. cmdStrmWtr.Write(szData, 0, szData.Length);
346. **this**.getSatus();
348. FileStream fstrm = **new** FileStream(filePath,FileMode.OpenOrCreate);
349. **char**[] fchars = **new** **char**[1030];
350. **byte**[] fbytes = **new** **byte**[1030];
351. **int** cnt = 0;
352. **while** ((cnt = dataStrmWtr.Read(fbytes, 0, 1024)) > 0)
353. {
354. fstrm.Write(fbytes, 0, cnt);
355. }
356. fstrm.Close();
358. **this**.closeDataPort();
359. Cursor.Current = cr;
361. }
363. **private** **void** Form1\_Load(**object** sender, EventArgs e)
364. {
366. }
368. #region 选择路径  -void txtPath\_Click(object sender, EventArgs e)
370. ///   <summary>
371. ///  选择路径
372. ///   </summary>
373. ///   <param name="sender"></param>
374. ///   <param name="e"></param>
375. **private** **void** txtPath\_Click(**object** sender, EventArgs e)
376. {
378. FolderBrowserDialog fbd = **new** FolderBrowserDialog();
379. **if** (fbd.ShowDialog() == DialogResult.OK)
380. {
381. txtPath.Text = fbd.SelectedPath;
382. }
383. }
384. #endregion
386. #region 获取指定目录子目录和子文件 -void btnGetPath\_Click(object sender, EventArgs e)
388. ///   <summary>
389. ///  获取指定目录子目录和子文件
390. ///   </summary>
391. ///   <param name="sender"></param>
392. ///   <param name="e"></param>
393. **private** **void** btnGetPath\_Click(**object** sender, EventArgs e)
394. {
395. tvDir.Nodes.Clear();
396. **if** (txtPath.Text.Trim().Length == 0)
397. {
398. **return**;
399. }
400. **else**
401. {
402. LoadData(txtPath.Text, **null**);
403. LoadFiles(txtPath.Text);
404. }
405. }
407. #endregion
409. #region 树控件节点点击之后触发  -void tvDir\_AfterSelect(object sender, TreeViewEventArgs e)
411. ///   <summary>
412. ///  树控件节点点击之后触发
413. ///   </summary>
414. ///   <param name="sender"></param>
415. ///   <param name="e"></param>
416. **private** **void** tvDir\_AfterSelect(**object** sender, TreeViewEventArgs e)
417. {
418. // 获取当前所选择的节点
419. TreeNode node = **this**.tvDir.SelectedNode;
420. **if** (node == **null**)  // 判断有没有选择某一个节点
421. {
422. **return**;
423. }
424. // 再将之前存储到Tag值的全路径取出来使用
425. **string** path = node.Tag.ToString();
426. LoadData(path, node);
427. LoadFiles(path);
428. }
429. #endregion
431. #region 加载树节点和listview控件的项  - void LoadData(string path, TreeNode parentNode)
433. ///   <summary>
434. ///  加载树节点和listview控件的项
435. ///   </summary>
436. ///   <param name="path"></param>
437. ///   <param name="parentNode"></param>
438. **void** LoadData(**string** path, TreeNode parentNode)
439. {
440. // tvDir.Nodes.Clear();
441. **if** (path.Trim().Length == 0)
442. {
443. MessageBox.Show(" 路径为空 ");
444. **return**;
445. }
446. **else**
447. {
448. DirectoryInfo dir = **new** DirectoryInfo(path);
449. DirectoryInfo[] dirs = dir.GetDirectories();
450. // 将得到的目录生成为树节点添加给刚刚点击的节点对象
451. **foreach** (DirectoryInfo d **in** dirs)
452. {
453. TreeNode subNode = **new** TreeNode(d.Name);  // 目录名称
454. subNode.Tag = d.FullName;  // 全路径
455. // 任何一个节点都会有一个了节点集合
456. **if** (parentNode == **null**)  // 为null说明当前是为树的根节点添加
457. {
458. tvDir.Nodes.Add(subNode);
459. }
460. **else**
461. {
462. parentNode.Nodes.Add(subNode);
463. }
464. // 加载一个目录，立刻来加载它的子目录
465. LoadData(subNode.Tag.ToString(), subNode);
466. }
467. }
468. }
469. #endregion
471. #region 加载文件  ListView- void LoadFiles(string path)
473. ///   <summary>
474. ///  加载文件
475. ///   </summary>
476. ///   <param name="path"></param>
477. **void** LoadFiles(**string** path)
478. {
479. DirectoryInfo dir = **new** DirectoryInfo(path);
480. **this**.tvDir.ExpandAll();
481. **this**.lvList.Items.Clear();
482. // 获取子文件
483. FileInfo[] myFiles = dir.GetFiles();
484. **foreach** (FileInfo f **in** myFiles)
485. {
486. ListViewItem lv = **new** ListViewItem(f.Name);
487. lv.SubItems.AddRange(**new** **string**[] { f.Extension, (f.Length / 1024).ToString() + " KB ", f.FullName });
488. lv.Tag = f.FullName;
489. **this**.lvList.Items.Add(lv);
490. }
491. }

494. #endregion
496. /// <summary>
497. /// 上传
498. /// </summary>
499. **private** **void** 上传ToolStripMenuItem\_Click(**object** sender, EventArgs e)
500. {
501. Cursor cr = Cursor.Current;
502. Cursor.Current = Cursors.WaitCursor;
504. **if** (btn\_conn.Text == "断开")
505. {
506. **string** fileName = lvList.SelectedItems[0].Text;
507. **string** filePath = lvList.SelectedItems[0].SubItems[3].Text;
509. **this**.openDataPort();
511. cmdData = "STOR " + fileName + CRLF;
512. szData = System.Text.Encoding.ASCII.GetBytes(cmdData.ToCharArray());
513. cmdStrmWtr.Write(szData, 0, szData.Length);
514. **this**.getSatus();
516. FileStream fstrm = **new** FileStream(filePath, FileMode.Open);
517. **byte**[] fbytes = **new** **byte**[1030];
518. **int** cnt = 0;
519. **while** ((cnt = fstrm.Read(fbytes, 0, 1024)) > 0)
520. {
521. dataStrmWtr.Write(fbytes, 0, cnt);
522. }
523. fstrm.Close();
525. **this**.closeDataPort();
527. **this**.freshFileBox\_Right();
528. }
529. **else**
530. {
531. MessageBox.Show("请先连接！");
532. }
533. Cursor.Current = cr;
535. }
537. **private** **void** lvList\_MouseUp(**object** sender, MouseEventArgs e)
538. {
540. //Object obj = lvList.SelectedIndices;

543. **if** (e.Button == MouseButtons.Right&&lvList.SelectedIndices.Count!=0)
544. {
546. contextMenuStrip2.Show(Control.MousePosition.X, Control.MousePosition.Y);

549. }
550. }
551. }
552. }

Form1.Designer.cs代码如下：

1. **namespace** WindowsFormsApp5
2. {
3. partial **class** Form1
4. {
5. /// <summary>
6. /// 必需的设计器变量。
7. /// </summary>
8. **private** System.ComponentModel.IContainer components = **null**;
10. /// <summary>
11. /// 清理所有正在使用的资源。
12. /// </summary>
13. /// <param name="disposing">如果应释放托管资源，为 true；否则为 false。</param>
14. **protected** **override** **void** Dispose(**bool** disposing)
15. {
16. **if** (disposing && (components != **null**))
17. {
18. components.Dispose();
19. }
20. **base**.Dispose(disposing);
21. }
23. #region Windows 窗体设计器生成的代码
25. /// <summary>
26. /// 设计器支持所需的方法 - 不要修改
27. /// 使用代码编辑器修改此方法的内容。
28. /// </summary>
29. **private** **void** InitializeComponent()
30. {
31. **this**.components = **new** System.ComponentModel.Container();
32. **this**.flowLayoutPanel1 = **new** System.Windows.Forms.FlowLayoutPanel();
33. **this**.lsb\_server = **new** System.Windows.Forms.ListBox();
34. **this**.contextMenuStrip1 = **new** System.Windows.Forms.ContextMenuStrip(**this**.components);
35. **this**.下载ToolStripMenuItem = **new** System.Windows.Forms.ToolStripMenuItem();
36. **this**.flowLayoutPanel2 = **new** System.Windows.Forms.FlowLayoutPanel();
37. **this**.tvDir = **new** System.Windows.Forms.TreeView();
38. **this**.lvList = **new** System.Windows.Forms.ListView();
39. **this**.columnHeader1 = ((System.Windows.Forms.ColumnHeader)(**new** System.Windows.Forms.ColumnHeader()));
40. **this**.columnHeader2 = ((System.Windows.Forms.ColumnHeader)(**new** System.Windows.Forms.ColumnHeader()));
41. **this**.columnHeader3 = ((System.Windows.Forms.ColumnHeader)(**new** System.Windows.Forms.ColumnHeader()));
42. **this**.columnHeader4 = ((System.Windows.Forms.ColumnHeader)(**new** System.Windows.Forms.ColumnHeader()));
43. **this**.panel1 = **new** System.Windows.Forms.Panel();
44. **this**.btn\_conn = **new** System.Windows.Forms.Button();
45. **this**.tb\_password = **new** System.Windows.Forms.TextBox();
46. **this**.tb\_username = **new** System.Windows.Forms.TextBox();
47. **this**.tb\_port = **new** System.Windows.Forms.TextBox();
48. **this**.tb\_IP = **new** System.Windows.Forms.TextBox();
49. **this**.label4 = **new** System.Windows.Forms.Label();
50. **this**.label3 = **new** System.Windows.Forms.Label();
51. **this**.label2 = **new** System.Windows.Forms.Label();
52. **this**.label1 = **new** System.Windows.Forms.Label();
53. **this**.lsb\_status = **new** System.Windows.Forms.ListBox();
54. **this**.lb\_IP = **new** System.Windows.Forms.Label();
55. **this**.button1 = **new** System.Windows.Forms.Button();
56. **this**.contextMenuStrip2 = **new** System.Windows.Forms.ContextMenuStrip(**this**.components);
57. **this**.上传ToolStripMenuItem = **new** System.Windows.Forms.ToolStripMenuItem();
58. **this**.txtPath = **new** System.Windows.Forms.TextBox();
59. **this**.btnGetPath = **new** System.Windows.Forms.Button();
60. **this**.label5 = **new** System.Windows.Forms.Label();
61. **this**.label6 = **new** System.Windows.Forms.Label();
62. **this**.flowLayoutPanel1.SuspendLayout();
63. **this**.contextMenuStrip1.SuspendLayout();
64. **this**.flowLayoutPanel2.SuspendLayout();
65. **this**.panel1.SuspendLayout();
66. **this**.contextMenuStrip2.SuspendLayout();
67. **this**.SuspendLayout();
68. //
69. // flowLayoutPanel1
70. //
71. **this**.flowLayoutPanel1.BackColor = System.Drawing.SystemColors.Window;
72. **this**.flowLayoutPanel1.Controls.Add(**this**.lsb\_server);
73. **this**.flowLayoutPanel1.Location = **new** System.Drawing.Point(2, 141);
74. **this**.flowLayoutPanel1.Name = "flowLayoutPanel1";
75. **this**.flowLayoutPanel1.Size = **new** System.Drawing.Size(437, 462);
76. **this**.flowLayoutPanel1.TabIndex = 0;
77. //
78. // lsb\_server
79. //
80. **this**.lsb\_server.FormattingEnabled = **true**;
81. **this**.lsb\_server.ItemHeight = 15;
82. **this**.lsb\_server.Location = **new** System.Drawing.Point(3, 3);
83. **this**.lsb\_server.Name = "lsb\_server";
84. **this**.lsb\_server.Size = **new** System.Drawing.Size(434, 454);
85. **this**.lsb\_server.TabIndex = 0;
86. **this**.lsb\_server.MouseClick += **new** System.Windows.Forms.MouseEventHandler(**this**.lsb\_server\_MouseClick);
87. **this**.lsb\_server.MouseUp += **new** System.Windows.Forms.MouseEventHandler(**this**.lsb\_server\_MouseUp);
88. //
89. // contextMenuStrip1
90. //
91. **this**.contextMenuStrip1.ImageScalingSize = **new** System.Drawing.Size(20, 20);
92. **this**.contextMenuStrip1.Items.AddRange(**new** System.Windows.Forms.ToolStripItem[] {
93. **this**.下载ToolStripMenuItem});
94. **this**.contextMenuStrip1.Name = "contextMenuStrip1";
95. **this**.contextMenuStrip1.Size = **new** System.Drawing.Size(109, 28);
96. //
97. // 下载ToolStripMenuItem
98. //
99. **this**.下载ToolStripMenuItem.Name = "下载ToolStripMenuItem";
100. **this**.下载ToolStripMenuItem.Size = **new** System.Drawing.Size(108, 24);
101. **this**.下载ToolStripMenuItem.Text = "下载";
102. **this**.下载ToolStripMenuItem.Click += **new** System.EventHandler(**this**.下载ToolStripMenuItem\_Click);
103. //
104. // flowLayoutPanel2
105. //
106. **this**.flowLayoutPanel2.BackColor = System.Drawing.SystemColors.Window;
107. **this**.flowLayoutPanel2.Controls.Add(**this**.tvDir);
108. **this**.flowLayoutPanel2.Controls.Add(**this**.lvList);
109. **this**.flowLayoutPanel2.Location = **new** System.Drawing.Point(446, 140);
110. **this**.flowLayoutPanel2.Name = "flowLayoutPanel2";
111. **this**.flowLayoutPanel2.Size = **new** System.Drawing.Size(599, 462);
112. **this**.flowLayoutPanel2.TabIndex = 1;
113. //
114. // tvDir
115. //
116. **this**.tvDir.Location = **new** System.Drawing.Point(3, 3);
117. **this**.tvDir.Name = "tvDir";
118. **this**.tvDir.Size = **new** System.Drawing.Size(301, 459);
119. **this**.tvDir.TabIndex = 0;
120. **this**.tvDir.AfterSelect += **new** System.Windows.Forms.TreeViewEventHandler(**this**.tvDir\_AfterSelect);
121. //
122. // lvList
123. //
124. **this**.lvList.Columns.AddRange(**new** System.Windows.Forms.ColumnHeader[] {
125. **this**.columnHeader1,
126. **this**.columnHeader2,
127. **this**.columnHeader3,
128. **this**.columnHeader4});
129. **this**.lvList.GridLines = **true**;
130. **this**.lvList.HideSelection = **false**;
131. **this**.lvList.Location = **new** System.Drawing.Point(310, 3);
132. **this**.lvList.Name = "lvList";
133. **this**.lvList.Size = **new** System.Drawing.Size(275, 455);
134. **this**.lvList.TabIndex = 1;
135. **this**.lvList.UseCompatibleStateImageBehavior = **false**;
136. **this**.lvList.View = System.Windows.Forms.View.Details;
137. **this**.lvList.MouseUp += **new** System.Windows.Forms.MouseEventHandler(**this**.lvList\_MouseUp);
138. //
139. // columnHeader1
140. //
141. **this**.columnHeader1.Text = "文件名";
142. //
143. // columnHeader2
144. //
145. **this**.columnHeader2.Text = "扩展名";
146. //
147. // columnHeader3
148. //
149. **this**.columnHeader3.Text = "文件大小";
150. //
151. // columnHeader4
152. //
153. **this**.columnHeader4.Text = "文件全路径";
154. //
155. // panel1
156. //
157. **this**.panel1.Controls.Add(**this**.btn\_conn);
158. **this**.panel1.Controls.Add(**this**.tb\_password);
159. **this**.panel1.Controls.Add(**this**.tb\_username);
160. **this**.panel1.Controls.Add(**this**.tb\_port);
161. **this**.panel1.Controls.Add(**this**.tb\_IP);
162. **this**.panel1.Controls.Add(**this**.label4);
163. **this**.panel1.Controls.Add(**this**.label3);
164. **this**.panel1.Controls.Add(**this**.label2);
165. **this**.panel1.Controls.Add(**this**.label1);
166. **this**.panel1.Location = **new** System.Drawing.Point(2, 55);
167. **this**.panel1.Name = "panel1";
168. **this**.panel1.Size = **new** System.Drawing.Size(1042, 45);
169. **this**.panel1.TabIndex = 2;
170. //
171. // btn\_conn
172. //
173. **this**.btn\_conn.Location = **new** System.Drawing.Point(903, 6);
174. **this**.btn\_conn.Name = "btn\_conn";
175. **this**.btn\_conn.Size = **new** System.Drawing.Size(104, 34);
176. **this**.btn\_conn.TabIndex = 8;
177. **this**.btn\_conn.Text = "连接";
178. **this**.btn\_conn.UseVisualStyleBackColor = **true**;
179. **this**.btn\_conn.Click += **new** System.EventHandler(**this**.btn\_conn\_Click);
180. //
181. // tb\_password
182. //
183. **this**.tb\_password.Location = **new** System.Drawing.Point(700, 13);
184. **this**.tb\_password.Name = "tb\_password";
185. **this**.tb\_password.Size = **new** System.Drawing.Size(151, 25);
186. **this**.tb\_password.TabIndex = 7;
187. **this**.tb\_password.Text = "123";
188. //
189. // tb\_username
190. //
191. **this**.tb\_username.Location = **new** System.Drawing.Point(500, 13);
192. **this**.tb\_username.Name = "tb\_username";
193. **this**.tb\_username.Size = **new** System.Drawing.Size(151, 25);
194. **this**.tb\_username.TabIndex = 6;
195. **this**.tb\_username.Text = "xwh";
196. **this**.tb\_username.TextChanged += **new** System.EventHandler(**this**.textBox3\_TextChanged);
197. //
198. // tb\_port
199. //
200. **this**.tb\_port.Location = **new** System.Drawing.Point(285, 13);
201. **this**.tb\_port.Name = "tb\_port";
202. **this**.tb\_port.Size = **new** System.Drawing.Size(151, 25);
203. **this**.tb\_port.TabIndex = 5;
204. **this**.tb\_port.Text = "21";
205. //
206. // tb\_IP
207. //
208. **this**.tb\_IP.Location = **new** System.Drawing.Point(85, 13);
209. **this**.tb\_IP.Name = "tb\_IP";
210. **this**.tb\_IP.Size = **new** System.Drawing.Size(151, 25);
211. **this**.tb\_IP.TabIndex = 4;
212. **this**.tb\_IP.Text = "172.19.176.1";
213. //
214. // label4
215. //
216. **this**.label4.AutoSize = **true**;
217. **this**.label4.Location = **new** System.Drawing.Point(657, 16);
218. **this**.label4.Name = "label4";
219. **this**.label4.Size = **new** System.Drawing.Size(37, 15);
220. **this**.label4.TabIndex = 3;
221. **this**.label4.Text = "密码";
222. //
223. // label3
224. //
225. **this**.label3.AutoSize = **true**;
226. **this**.label3.Location = **new** System.Drawing.Point(442, 16);
227. **this**.label3.Name = "label3";
228. **this**.label3.Size = **new** System.Drawing.Size(52, 15);
229. **this**.label3.TabIndex = 2;
230. **this**.label3.Text = "用户名";
231. //
232. // label2
233. //
234. **this**.label2.AutoSize = **true**;
235. **this**.label2.Location = **new** System.Drawing.Point(242, 16);
236. **this**.label2.Name = "label2";
237. **this**.label2.Size = **new** System.Drawing.Size(37, 15);
238. **this**.label2.TabIndex = 1;
239. **this**.label2.Text = "端口";
240. //
241. // label1
242. //
243. **this**.label1.AutoSize = **true**;
244. **this**.label1.Location = **new** System.Drawing.Point(18, 16);
245. **this**.label1.Name = "label1";
246. **this**.label1.Size = **new** System.Drawing.Size(61, 15);
247. **this**.label1.TabIndex = 0;
248. **this**.label1.Text = "FTP地址";
249. //
250. // lsb\_status
251. //
252. **this**.lsb\_status.FormattingEnabled = **true**;
253. **this**.lsb\_status.ItemHeight = 15;
254. **this**.lsb\_status.Location = **new** System.Drawing.Point(-1, 648);
255. **this**.lsb\_status.Name = "lsb\_status";
256. **this**.lsb\_status.Size = **new** System.Drawing.Size(1042, 79);
257. **this**.lsb\_status.TabIndex = 3;
258. //
259. // lb\_IP
260. //
261. **this**.lb\_IP.AutoSize = **true**;
262. **this**.lb\_IP.Location = **new** System.Drawing.Point(3, 119);
263. **this**.lb\_IP.Name = "lb\_IP";
264. **this**.lb\_IP.Size = **new** System.Drawing.Size(0, 15);
265. **this**.lb\_IP.TabIndex = 4;
266. //
267. // button1
268. //
269. **this**.button1.Location = **new** System.Drawing.Point(38, 21);
270. **this**.button1.Name = "button1";
271. **this**.button1.Size = **new** System.Drawing.Size(118, 28);
272. **this**.button1.TabIndex = 5;
273. **this**.button1.Text = "上级目录";
274. **this**.button1.UseVisualStyleBackColor = **true**;
275. **this**.button1.Click += **new** System.EventHandler(**this**.button1\_Click);
276. //
277. // contextMenuStrip2
278. //
279. **this**.contextMenuStrip2.ImageScalingSize = **new** System.Drawing.Size(20, 20);
280. **this**.contextMenuStrip2.Items.AddRange(**new** System.Windows.Forms.ToolStripItem[] {
281. **this**.上传ToolStripMenuItem});
282. **this**.contextMenuStrip2.Name = "contextMenuStrip2";
283. **this**.contextMenuStrip2.Size = **new** System.Drawing.Size(109, 28);
284. //
285. // 上传ToolStripMenuItem
286. //
287. **this**.上传ToolStripMenuItem.Name = "上传ToolStripMenuItem";
288. **this**.上传ToolStripMenuItem.Size = **new** System.Drawing.Size(108, 24);
289. **this**.上传ToolStripMenuItem.Text = "上传";
290. **this**.上传ToolStripMenuItem.Click += **new** System.EventHandler(**this**.上传ToolStripMenuItem\_Click);
291. //
292. // txtPath
293. //
294. **this**.txtPath.Location = **new** System.Drawing.Point(492, 109);
295. **this**.txtPath.Name = "txtPath";
296. **this**.txtPath.Size = **new** System.Drawing.Size(386, 25);
297. **this**.txtPath.TabIndex = 6;
298. **this**.txtPath.Click += **new** System.EventHandler(**this**.txtPath\_Click);
299. //
300. // btnGetPath
301. //
302. **this**.btnGetPath.Location = **new** System.Drawing.Point(901, 101);
303. **this**.btnGetPath.Name = "btnGetPath";
304. **this**.btnGetPath.Size = **new** System.Drawing.Size(94, 32);
305. **this**.btnGetPath.TabIndex = 7;
306. **this**.btnGetPath.Text = "获取";
307. **this**.btnGetPath.UseVisualStyleBackColor = **true**;
308. **this**.btnGetPath.Click += **new** System.EventHandler(**this**.btnGetPath\_Click);
309. //
310. // label5
311. //
312. **this**.label5.AutoSize = **true**;
313. **this**.label5.Location = **new** System.Drawing.Point(389, 112);
314. **this**.label5.Name = "label5";
315. **this**.label5.Size = **new** System.Drawing.Size(97, 15);
316. **this**.label5.TabIndex = 8;
317. **this**.label5.Text = "本地文件目录";
318. //
319. // label6
320. //
321. **this**.label6.AutoSize = **true**;
322. **this**.label6.Location = **new** System.Drawing.Point(12, 620);
323. **this**.label6.Name = "label6";
324. **this**.label6.Size = **new** System.Drawing.Size(82, 15);
325. **this**.label6.TabIndex = 9;
326. **this**.label6.Text = "状态信息：";
327. //
328. // Form1
329. //
330. **this**.AutoScaleDimensions = **new** System.Drawing.SizeF(8F, 15F);
331. **this**.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
332. **this**.ClientSize = **new** System.Drawing.Size(1043, 757);
333. **this**.Controls.Add(**this**.label6);
334. **this**.Controls.Add(**this**.label5);
335. **this**.Controls.Add(**this**.btnGetPath);
336. **this**.Controls.Add(**this**.txtPath);
337. **this**.Controls.Add(**this**.button1);
338. **this**.Controls.Add(**this**.lb\_IP);
339. **this**.Controls.Add(**this**.lsb\_status);
340. **this**.Controls.Add(**this**.panel1);
341. **this**.Controls.Add(**this**.flowLayoutPanel2);
342. **this**.Controls.Add(**this**.flowLayoutPanel1);
343. **this**.Name = "Form1";
344. **this**.Text = "FTP\_Client";
345. **this**.Load += **new** System.EventHandler(**this**.Form1\_Load);
346. **this**.flowLayoutPanel1.ResumeLayout(**false**);
347. **this**.contextMenuStrip1.ResumeLayout(**false**);
348. **this**.flowLayoutPanel2.ResumeLayout(**false**);
349. **this**.panel1.ResumeLayout(**false**);
350. **this**.panel1.PerformLayout();
351. **this**.contextMenuStrip2.ResumeLayout(**false**);
352. **this**.ResumeLayout(**false**);
353. **this**.PerformLayout();
355. }
357. #endregion
359. **private** System.Windows.Forms.FlowLayoutPanel flowLayoutPanel1;
360. **private** System.Windows.Forms.FlowLayoutPanel flowLayoutPanel2;
361. **private** System.Windows.Forms.Panel panel1;
362. **private** System.Windows.Forms.TextBox tb\_username;
363. **private** System.Windows.Forms.TextBox tb\_port;
364. **private** System.Windows.Forms.TextBox tb\_IP;
365. **private** System.Windows.Forms.Label label4;
366. **private** System.Windows.Forms.Label label3;
367. **private** System.Windows.Forms.Label label2;
368. **private** System.Windows.Forms.Label label1;
369. **private** System.Windows.Forms.Button btn\_conn;
370. **private** System.Windows.Forms.TextBox tb\_password;
371. **private** System.Windows.Forms.TreeView tvDir;
372. **private** System.Windows.Forms.ListBox lsb\_status;
373. **private** System.Windows.Forms.ListBox lsb\_server;
374. **private** System.Windows.Forms.Label lb\_IP;
375. **private** System.Windows.Forms.Button button1;
376. **private** System.Windows.Forms.ContextMenuStrip contextMenuStrip1;
377. **private** System.Windows.Forms.ToolStripMenuItem 下载ToolStripMenuItem;
378. **private** System.Windows.Forms.ContextMenuStrip contextMenuStrip2;
379. **private** System.Windows.Forms.ToolStripMenuItem 上传ToolStripMenuItem;
380. **private** System.Windows.Forms.ListView lvList;
381. **private** System.Windows.Forms.TextBox txtPath;
382. **private** System.Windows.Forms.Button btnGetPath;
383. **private** System.Windows.Forms.ColumnHeader columnHeader1;
384. **private** System.Windows.Forms.ColumnHeader columnHeader2;
385. **private** System.Windows.Forms.ColumnHeader columnHeader3;
386. **private** System.Windows.Forms.ColumnHeader columnHeader4;
387. **private** System.Windows.Forms.Label label5;
388. **private** System.Windows.Forms.Label label6;
389. }
390. }

# 参考文献

[1] 唐大仕. C#程序设计教程[M]. 第2版. 清华大学出版社, 2018.

[2] 薄荷. FTP命令列表[EB/OL]. 2013[2021.8.4]. https://blog.csdn.net/chenzhiqin20/article/details/17450597.

[3] IronLavender. 使用Socket通信实现FTP客户端程序【很详细】[EB/OL]. 2016[2021.8.4]. https://blog.csdn.net/wumenglu1018/article/details/53749452.

[4] zzkdev. ftp客户端实现（一）[EB/OL]. 2018[2021.8.4]. https://www.jianshu.com/p/a17ab9ccb569.

|  |
| --- |
| 【结论】：本次课程设计主要实现了FTP客户端，主要功能就是上传下载以及其他调整服务器工作目录等功能，进一步体会了使用socket进行网络通信，以及winform一些控件相关的知识，完成了一个图形化的客户端。 |
| 【小结】：通过本次课程设计，我进一步体会到如何使用编程语言来进行网络通信，实现一个微型的客户端程序，以及socket在网络编程中的作用与应用，同时也锻炼了我的动手能力以及自学能力，实现了简单的UI界面。在这次实际开发过程中也遇到不少问题，比如因为在VS2019中没添加引用，导致程序一度代码全红，最后通过导入引用包来解决，提高了我的调试与排查能力。 |
| 指导老师评语及成绩 |
| 【评语】：  成 绩： 指导老师签名：吕慧  批阅日期：2021.8.15 |