**实验一 Socket端到端通信程序**

**实验目的:**

1. 理解socket通信的原理和过程
2. 掌握python中通信编程的基本方法

3. 掌握tkinter模块的基本用法和GUI设计方法

4. 掌握socket模块的基本用法和TCP网络编程方法

5. 掌握客户端和服务器之间的通信协议和数据交换过程

6. 掌握调试网络通信程序的方法

**实验步骤：**

以两位同学为一组进行实验

**练习1：使用python编写一个socket通信程序，该程序指定使用python中的tkinter模块设计界面，功能包括对指定用户发送文本信息，同时监听其他用户发送给本机的信息。**

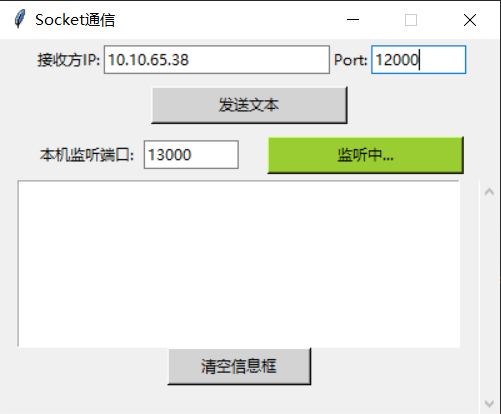
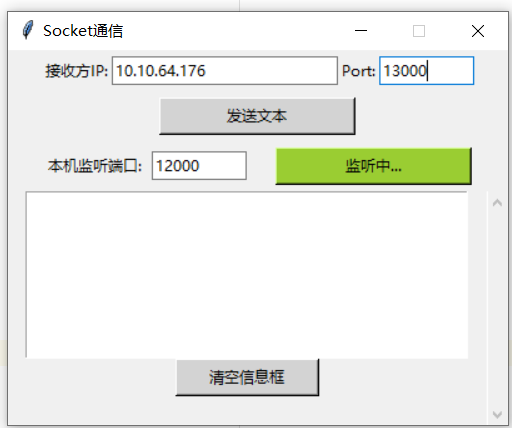
程序主界面的示范如下：

1. 点击“启动监听”按钮后，程序才能不断接收其他用户发送的信息。
2. 要成功对其他用户发送信息，必须准确填写其他用户IP地址以及监听端口。

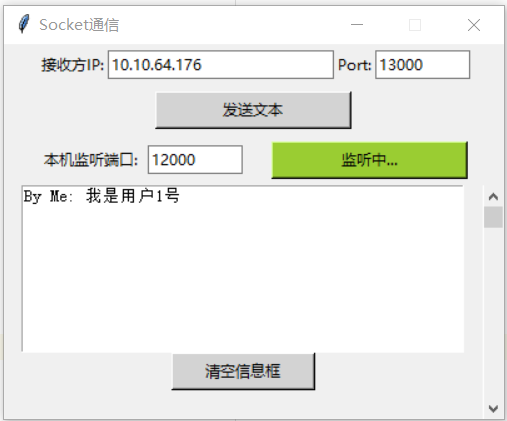


程序通信过程的示范如下：

1. 使用的主机需要连接在同一局域网之下，各自设定本机的监听端口号后启动监听，准确填写接收方IP和端口号。用户1号：10.10.65.38；用户2号：10.10.64.176。



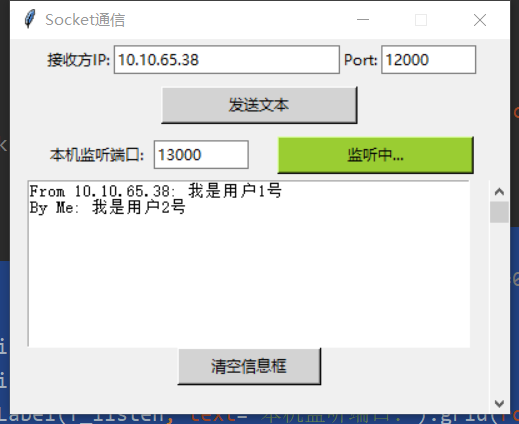
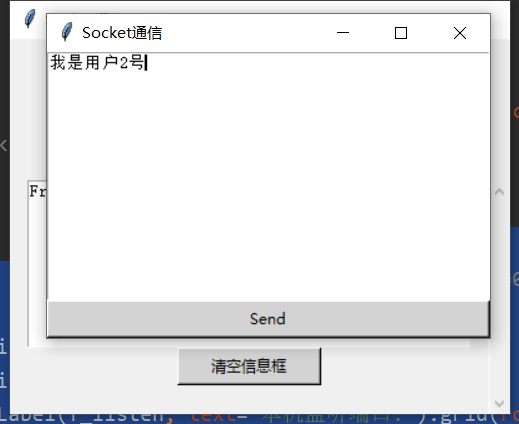
1. 用户1号点击发送文本，弹出文本编辑框，填写信息后点击发送按钮，文本编辑框清空，已发送的信息显示在本机聊天框中。



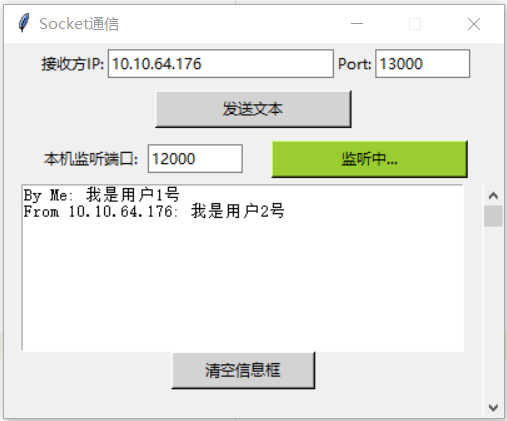
1. 用户2号收到信息。



1. 用户2号进行信息回复。



1. 用户1号接收信息。



● **请按照以上要求，编写socket通信程序，截图记录网络通信过程，并附上实验代码。**

**练习2：在练习1的基础上增加发送文件以及图片功能。**

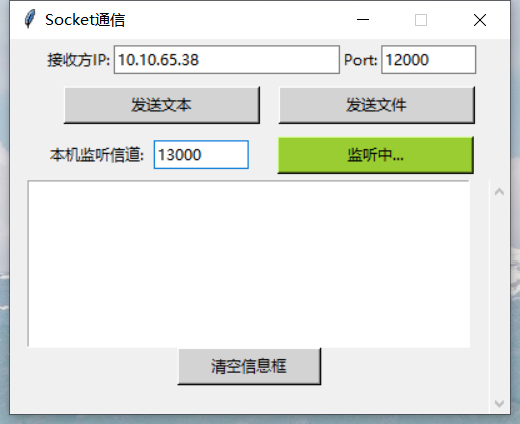
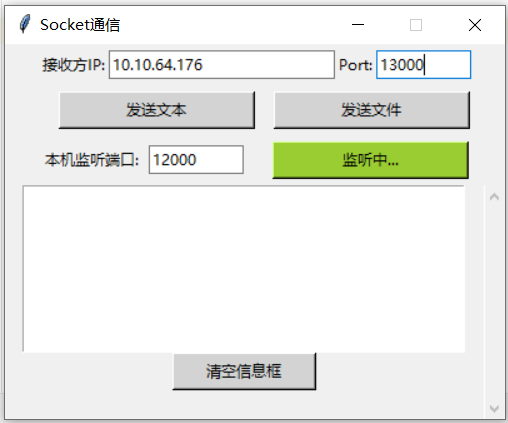
程序主界面的示范如下：

1. 增加“发送文件”按钮，实现发送文件以及图片功能。

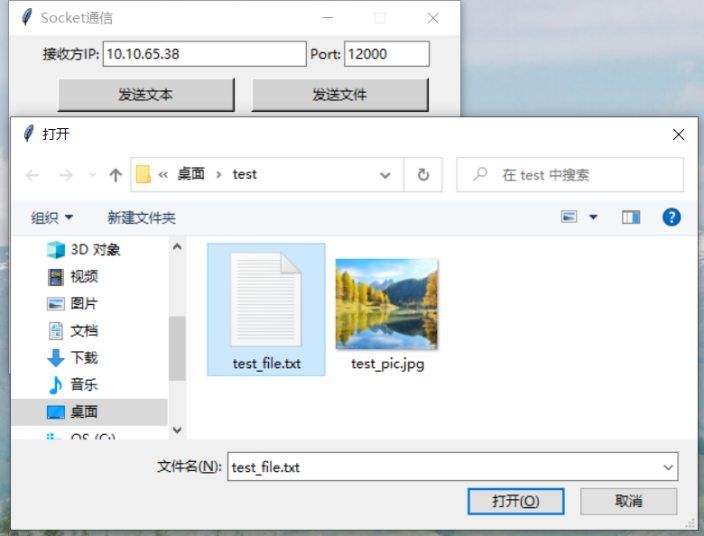


程序发送文件功能的示范如下：

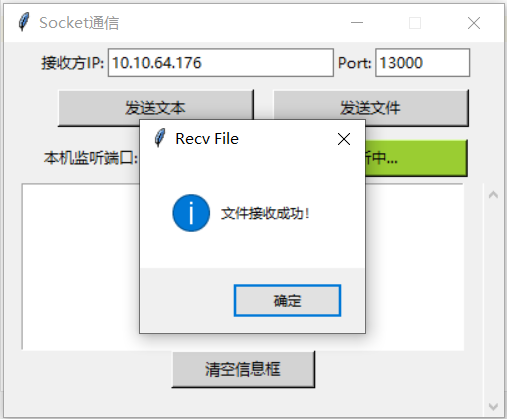
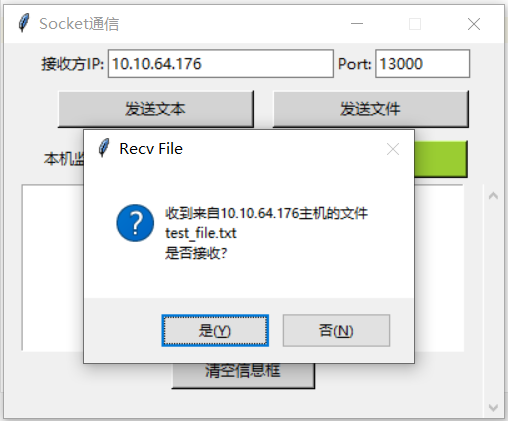
1. 使用的主机需要连接在同一局域网之下，各自设定本机的监听端口号后启动监听，准确填写接收方IP和端口号。用户1号：10.10.65.38；用户2号：10.10.64.176。

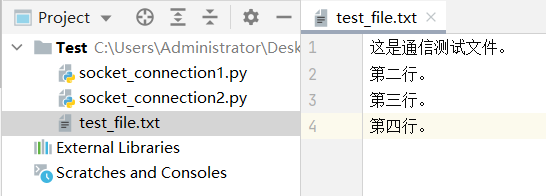


1. 用户2号发送文件。

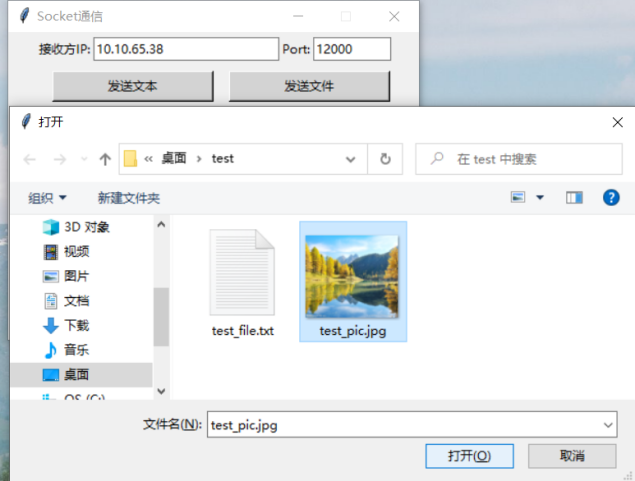


1. 用户1号接收文件，默认存储在项目根目录下。

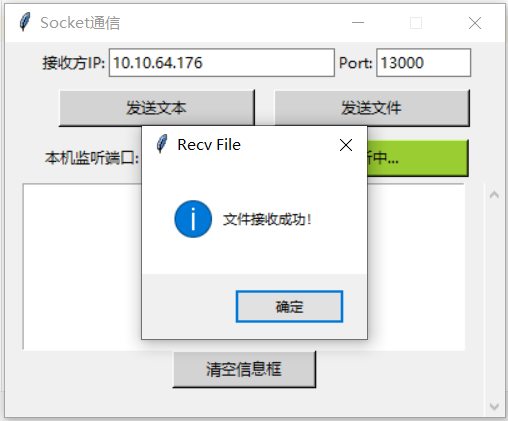
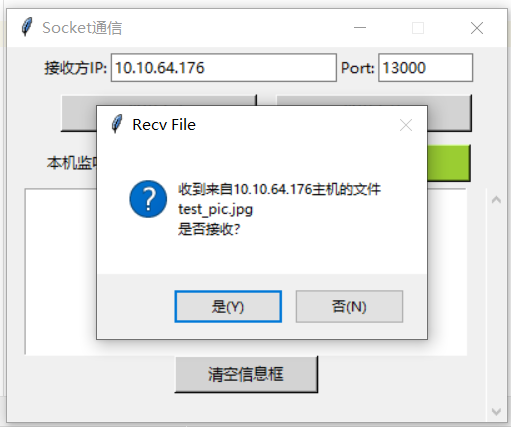


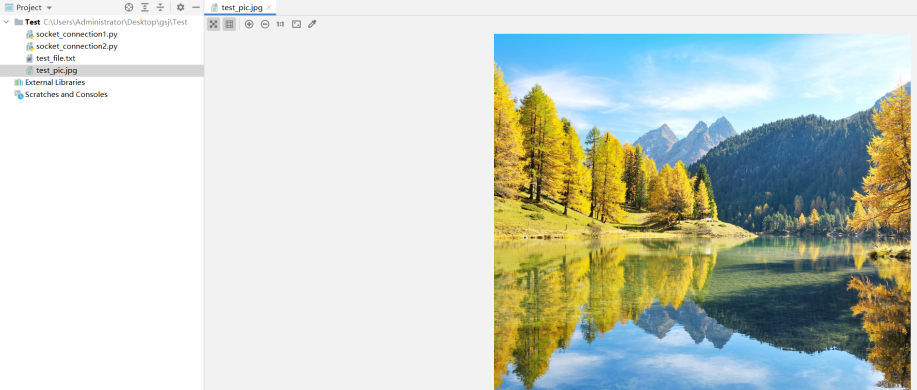


1. 用户2号发送图片。



1. 用户1号接收图片，默认存储在项目根目录下。

****

****

● **请按照以上要求，优化socket通信程序，截图记录网络通信过程，并附上实验代码。**

**练习3：在练习2的基础上美化界面，增加程序的鲁棒性和用户友好性。**

例如：

1. 修改界面布局、颜色使其更美观。
2. 增加错误输入提示功能，使程序鲁棒性更高。
3. 显示本机的IP在标题栏中，使截图中不同用户更具有区分度。
4. 增加修改接收文件默认保存地址功能，或者在接收文件时弹出文件保存选择框。
5. 文件的发送和接收之后在聊天框中增加“文件已发送（接收）”的友好提示。

……

● **请自行优化socket通信程序，说明优化点并截图记录，并附上实验代码。**

**主要代码：**

1. **socket通信的主要步骤**

|  |
| --- |
| # 导入socket包 import socket  # 本机设定为服务器，通过监听端口接收信息  self.server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  self.server.bind(("0.0.0.0", int(port)))  self.server.listen(3) while True:  try:  conn, addr = self.server.accept()  except:  break  recv\_data = conn.recv(1024)  # 客户端连接并发送信息  try:  client = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  client.connect((host, int(port)))  client.send(bytes('#msg#', encoding='utf-8'))  client.send(bytes(msg, encoding='utf-8'))  client.close() |

1. **tkinter创建按钮**

|  |
| --- |
| # 导入tkinter包 import tkinter as tk  from tkinter import (  ttk,  filedialog,  messagebox,  )  btn\_choice = tk.Frame(self.root, pady=5)  btn\_choice.pack()  tk.Button(btn\_choice, text='发送文本',  bg=self.btn\_color,  padx=15,  width=17,  command=self.choicetxt).grid(row=0, column=1, padx=15)  tk.Button(btn\_choice, text='发送文件',  bg=self.btn\_color,  padx=15,  width=17,  command=self.choicefile).grid(row=0, column=2) |

1. **tkinter创建文本框**

|  |
| --- |
| self.f\_panel = tk.Toplevel(self.root)  self.f\_panel.geometry("+580+210")  self.f\_panel.grab\_set()  self.textbox = tk.Text(self.f\_panel, width=50, height=15)  self.textbox.pack()  tk.Button(self.f\_panel, text='Send', bg=self.btn\_color,  command=lambda: self.thread\_it(self.send)).pack(fill=tk.X) |