

# 计算机网络

## 实验四 Socket 网络编程

---

谢瑞桃

[xie@szu.edu.cn](mailto:xie@szu.edu.cn)

[rtxie.github.io](https://github.com/rtxie)

计算机与软件学院

深圳大学



# 实验四 Socket 网络编程

---

## ■ 实验目的

- 理解UDP与TCP套接字的区别
- 掌握UDP和TCP套接字编程方法
- 了解简单网络应用的编程思路
- 了解网络编程相关的一些库

## ■ 实验环境

- 具有Internet连接的主机
- 某一种编程语言



# 实验内容

---

1. URL 请求程序
2. 系统时间查询
3. 网络文件传输
4. 网络聊天室



# 实验任务要求

- 请参考本讲义与附件资料学习套接字编程的基础知识
- 了解网络编程的相关库
- 掌握编写简单网络应用的技能
- 依照步骤完成实验内容1—4
- 对实验结果截图
- 撰写实验报告



# 实验报告撰写要求

- 使用教务处制作的实验报告模板
- 注意按进度填写实验时间和实验报告提交时间
- 填写模板中的每一部分
- 填写实验步骤时，做到条理清晰
- 注意截图清晰、美观
- 对于实验操作返回结果的解释为加分项，解释地越详细越好
- 禁止抄袭实验报告，抄袭的实验报告都记为0分



# 前言

---

- 请在Python官网下载并安装Python 3。  
<https://www.python.org/>
- 开发环境可以用Wing Python ID，也可以用任何文本编辑器。  
<https://wingware.com/>



# 学习Python的基础知识

---

- Python官方的Tutorial
  - <https://docs.python.org/3/tutorial/>
- 掌握以下内容
  - 3. An informal Introduction to Python
    - strings
  - 4. More Control Flow Tools
    - while 语句
    - if/else 语句
    - for 语句
  - 7. Input and Output
    - 文件读取/写入
    - 打印字符串



# 遇到问题怎么办？

---

- ***Please Relax !***
- ***请放轻松，地球不会爆炸！***
- ***请擅用搜索引擎！***
- ***请在动手之前先学习！***
- ***学会读英文资料，你将能学到更多！***
- ***请尽情发挥你的聪明才智！***





# 实验内容

---

1. URL 请求程序
2. 系统时间查询
3. 网络文件传输
4. 网络聊天室



# 1. URL请求程序

---

- 利用Python的HTTP库**Requests**实现一个简单的程序。
- Requests库的官网：[requests.readthedocs.io](https://requests.readthedocs.io)
- 请在官网上阅读用户指南（有中文版），你只需要浏览快速上手部分的四个段落：发送请求、响应内容、二进制内容、原始响应内容。
- 请求一个网页，并存储为html文件。
- 计算所请求网页的大小。



# 1. URL请求程序

- 要求打印所请求网页的URL、存储文件名、文件大小等信息。
- 下图是一个范例。

```
Windows PowerShell
PS D:\doris\computer network\laboratory\lab-4-resources\url download> python .\url_download.py
Please input a URL:
http://rtxie.github.io
The requested url: https://rtxie.github.io/
The saved file: rtxie.github.io.html
The requested file size: 63645 bytes
PS D:\doris\computer network\laboratory\lab-4-resources\url download>
```



## 2. 系统时间查询

---

- 实现一个基于客户/服务器的系统时间查询程序。
- 传输层使用TCP。
- 交互过程
  - 1) 客户端向服务器端发送字符串"Time"。
  - 2) 服务器端收到该字符串后，返回当前系统时间。
  - 3) 客户端向服务器端发送字符串"Exit"。
  - 4) 服务器端返回"Bye"，然后结束TCP连接。

## 2. 系统时间查询

- 要求服务器端和客户端打印交互过程。
- 下图是一个范例。

### 服务器

```
Windows PowerShell
python .\system_time_inquiry_server.py

The server is ready to connect.
The server address: ('0.0.0.0', 12000)

Accepted a new connection.
The connection address: ('127.0.0.1', 12000)
The client address: ('127.0.0.1', 35637)
Received a request: Time.
Send a response: 2020-04-01 15:39:30.
Received a request: Exit.
Send a response: Bye.

Accepted a new connection.
The connection address: ('192.168.2.178', 12000)
The client address: ('192.168.2.119', 61632)
Received a request: Time.
Send a response: 2020-04-01 15:39:46.
Received a request: Exit.
Send a response: Bye.

Accepted a new connection.
The connection address: ('192.168.2.178', 12000)
The client address: ('192.168.2.119', 61633)
Received a request: Time.
Send a response: 2020-04-01 15:40:00.
Received a request: Exit.
Send a response: Bye.
```

### 本地客户端1

```
Windows PowerShell
python .\system_time_inquiry_client.py
A client is running.
The client address: ('127.0.0.1', 35637)
Connected to 127.0.0.1:12000.
Send a request: Time.
Received the current system time on the server: 2020-04-01 15:39:30.
Send a request: Exit.
Received a response: Bye.
```

### 远程客户端2

```
die@Doriss-Air ~/Downloads — zsh — 80x17
python system_time_inquiry_client.py
A client is running.
The client address: ('192.168.2.119', 61632)
Connected to 192.168.2.178:12000.
Send a request: Time
Received the current system time on the server: 2020-04-01 15:39:46.
Send a request: Exit
Received a response: Bye.
python system_time_inquiry_client.py
A client is running.
The client address: ('192.168.2.119', 61633)
Connected to 192.168.2.178:12000.
Send a request: Time.
Received the current system time on the server: 2020-04-01 15:40:00.
Send a request: Exit.
Received a response: Bye.
```



### 3. 网络文件传输

---

- 实现一个基于客户/服务器的网络文件传输程序。
- 传输层使用TCP。
- 交互过程
  - 1) 客户端从用户输入获得待请求的文件名。
  - 2) 客户端向服务器端发送文件名。
  - 3) 服务器端收到文件名后，传输文件。
  - 4) 客户端接收文件，重命名并存储在硬盘。





## 4. 网络聊天室

---

- 实现一个基于客户/服务器的网络聊天程序。
- 要求实现多个用户的群聊。
- 传输层使用UDP。
- 不要求实现GUI界面。
  - 计算机专业或感兴趣的同学可以挑战一下。





## 4. 网络聊天室Q&A

---

Q 什么是群聊呢?

A 一个用户发消息, 所有用户都能收到。相应地, 一个客户端把聊天消息发给服务器, 服务器再将收到的消息转发给所有客户端。

Q 服务器怎么知道聊天室里有哪些用户(客户端)呢?

A 这就需要服务器程序维护一个聊天室用户列表。新用户到达时加进来, 旧用户离开时删除。

Q 我们要求用UDP实现, 那每次服务器收到消息, 它怎么知道这次是新用户?

A 通过查询用户列表, 不就知道了, 新用户肯定不在表里。



## 4. 网络聊天室Q&A

---

**Q** 那怎么才能知道旧用户什么时候离开呢？

**A** UDP数据报是反映不出来的。那这就需要应用层设计协议实现，比如要求用户离开的时候输入特殊字符串。服务器收到消息时，通过查看消息内容判断用户是不是终止聊天。

**Q** 客户端好像要做两件事情：要从键盘获取用户输入并把消息发给服务器，还要接收并显示服务器发送的其他用户的聊天消息。那这两件事怎么能同时进行了？

**A** 可以通过两个线程实现的。一个线程负责接收并显示消息，另一个线程负责获取输入和发送消息。Python里threading库可以实现多线程编程。

# 4. 网络聊天室

- 要求客户端打印聊天消息，服务器打印系统信息。
- 下图是一个范例。(用户Leonard, Sheldon, Penny)

```
Windows PowerShell
python .\chat_client_udp.py
Your client is created at port 6996
Please input your nickname:
Leonard
Welcome Leonard, input <quit> to quit the chat!
Leonard has joined the chat!
Sheldon has joined the chat!
Penny has joined the chat!
New neighbour?
[Leonard] : New neighbour?
[Sheldon] : Evidently.
Significant improvement over the old neighbour.
[Leonard] : Significant improvement over the old neighbour.
[Sheldon] : Two hundred pound transvestite with a skin condition,
yes she is.
[Penny] : Oh, hi!
Hi.

[Leonard] : Hi.
[Leonard] :
[Sheldon] : Hi.
Hi.
[Leonard] : Hi.
[Sheldon] : Hi.
[Penny] : Hi?
We don't mean to interrupt, we live across the hall.
[Leonard] : We don't mean to interrupt, we live across the hall.

[Penny] : Oh, that's nice.
Oh... uh... no... we don't live together... um... we live together bu
t in separate, heterosexual bedrooms.
[Leonard] : Oh... uh... no... we don't live together... um... we live
together but in separate, heterosexual bedrooms.
[Penny] : Oh, okay, well, guess I'm your new neighbour, Penny.
Leonard, Sheldon.
[Leonard] : Leonard, Sheldon.
[Penny] : Hi.
Hi.
[Leonard] : Hi.
[Sheldon] : Hi.
[Penny] : Hi.
Hi. Well, uh, oh, welcome to the building.
[Leonard] : Hi. Well, uh, oh, welcome to the building.
[Penny] : Thankyou, maybe we can have coffee sometime.
Oh, great.
[Leonard] : Oh, great.
[Penny] : Great.
[Sheldon] : Great.
Great. Well, bye.
[Leonard] : Great. Well, bye.
[Penny] : Bye.
Penny has left the chat!
[Sheldon] : Bye.
Bye.
[Leonard] : Bye.
```

```
Windows PowerShell
python .\chat_client_udp.py
Your client is created at port 6678
Please input your nickname:
Sheldon
Welcome Sheldon, input <quit> to quit the chat!
Sheldon has joined the chat!
Penny has joined the chat!
[Leonard] : New neighbour?
Evidently.
[Sheldon] : Evidently.
[Leonard] : Significant improvement over the old neighbour.
Two hundred pound transvestite with a skin condition, yes she is.

[Sheldon] : Two hundred pound transvestite with a skin condition,
yes she is.
[Penny] : Oh, hi!
[Leonard] : Hi.
[Leonard] :
Hi.
[Sheldon] : Hi.
[Penny] : Hi?
[Leonard] : We don't mean to interrupt, we live across the hall.

[Penny] : Oh, that's nice.
[Leonard] : Oh... uh... no... we don't live together... um... we live
together but in separate, heterosexual bedrooms.
[Penny] : Oh, okay, well, guess I'm your new neighbour, Penny.
[Leonard] : Leonard, Sheldon.
[Penny] : Hi.
[Leonard] : Hi.
Hi.
[Sheldon] : Hi.
[Penny] : Hi.
[Leonard] : Hi. Well, uh, oh, welcome to the building.
[Penny] : Thankyou, maybe we can have coffee sometime.
[Leonard] : Oh, great.
[Penny] : Great.
Great.
[Sheldon] : Great.
[Leonard] : Great. Well, bye.
[Penny] : Bye.
Penny has left the chat!
Bye.
[Sheldon] : Bye.
[Leonard] : Bye.
```

```
Windows PowerShell
python .\chat_client_udp.py
Your client is created at port 8810
Please input your nickname:
Penny
Welcome Penny, input <quit> to quit the chat!
Penny has joined the chat!
[Leonard] : New neighbour?
[Sheldon] : Evidently.
[Leonard] : Significant improvement over the old neighbour.
[Sheldon] : Two hundred pound transvestite with a skin condition,
yes she is.
Oh, hi!
[Penny] : Oh, hi!
[Leonard] : Hi.
[Leonard] :
[Sheldon] : Hi.
[Leonard] : Hi.
[Sheldon] : Hi.
Hi?
[Penny] : Hi?
[Leonard] : We don't mean to interrupt, we live across the hall.

Oh, that's nice.
[Penny] : Oh, that's nice.
[Leonard] : Oh... uh... no... we don't live together... um... we live
together but in separate, heterosexual bedrooms.
Oh, okay, well, guess I'm your new neighbour, Penny.
[Penny] : Oh, okay, well, guess I'm your new neighbour, Penny.
[Leonard] : Leonard, Sheldon.
Hi.
[Penny] : Hi.
[Leonard] : Hi.
[Sheldon] : Hi.
Hi.
[Penny] : Hi.
[Leonard] : Hi. Well, uh, oh, welcome to the building.
Thankyou, maybe we can have coffee sometime.
[Penny] : Thankyou, maybe we can have coffee sometime.
[Leonard] : Oh, great.
Great.
[Penny] : Great.
[Sheldon] : Great.
[Leonard] : Great. Well, bye.
Bye.
[Penny] : Bye.
<quit>
PS D:\doris\computer network\laboratory\lab-4-resources\chatroom>
```

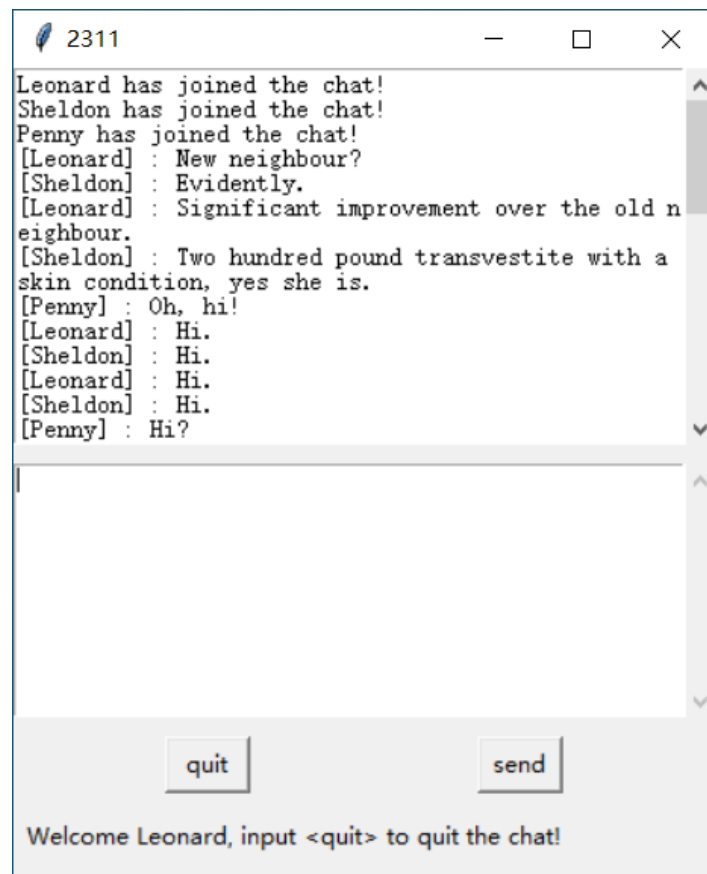
# 4. 网络聊天室

- 要求客户端打印聊天消息，服务器打印系统信息。
- 下图是一个范例。(服务器端)

```
Windows PowerShell
PS D:\doris\computer network\laboratory\lab-4-resources\chatroom> python .\chat_server_udp.py
The server is ready to receive.
Receive from 127.0.0.1:6996-->Leonard
Send to 127.0.0.1:6996-->b'Leonard has joined the chat!'
Receive from 127.0.0.1:6678-->Sheldon
Send to 127.0.0.1:6996-->b'Sheldon has joined the chat!'
Send to 127.0.0.1:6678-->b'Sheldon has joined the chat!'
Receive from 127.0.0.1:8810-->Penny
Send to 127.0.0.1:6996-->b'Penny has joined the chat!'
Send to 127.0.0.1:6678-->b'Penny has joined the chat!'
Send to 127.0.0.1:8810-->b'Penny has joined the chat!'
Receive from 127.0.0.1:6996-->New neighbour?
Send to 127.0.0.1:6996-->b'[Leonard] : New neighbour?'
Send to 127.0.0.1:6678-->b'[Leonard] : New neighbour?'
Send to 127.0.0.1:8810-->b'[Leonard] : New neighbour?'
Receive from 127.0.0.1:6678-->Evidently.
Send to 127.0.0.1:6996-->b'[Sheldon] : Evidently.'
Send to 127.0.0.1:6678-->b'[Sheldon] : Evidently.'
Send to 127.0.0.1:8810-->b'[Sheldon] : Evidently.'
Receive from 127.0.0.1:6996-->Significant improvement over the old neighbour.
Send to 127.0.0.1:6996-->b'[Leonard] : Significant improvement over the old neighbour.'
Send to 127.0.0.1:6678-->b'[Leonard] : Significant improvement over the old neighbour.'
Send to 127.0.0.1:8810-->b'[Leonard] : Significant improvement over the old neighbour.'
Receive from 127.0.0.1:6678-->Two hundred pound transvestite with a skin condition, yes she is.
Send to 127.0.0.1:6996-->b'[Sheldon] : Two hundred pound transvestite with a skin condition, yes she is.'
Send to 127.0.0.1:6678-->b'[Sheldon] : Two hundred pound transvestite with a skin condition, yes she is.'
Send to 127.0.0.1:8810-->b'[Sheldon] : Two hundred pound transvestite with a skin condition, yes she is.'
Receive from 127.0.0.1:8810-->Oh, hi!
Send to 127.0.0.1:6996-->b'[Penny] : Oh, hi!'
Send to 127.0.0.1:6678-->b'[Penny] : Oh, hi!'
Send to 127.0.0.1:8810-->b'[Penny] : Oh, hi!'
Receive from 127.0.0.1:6996-->Hi.
Receive from 127.0.0.1:6996-->
Send to 127.0.0.1:6996-->b'[Leonard] : Hi.'
Send to 127.0.0.1:6678-->b'[Leonard] : Hi.'
Send to 127.0.0.1:8810-->b'[Leonard] : Hi.'
Send to 127.0.0.1:6996-->b'[Leonard] : '
Send to 127.0.0.1:6678-->b'[Leonard] : '
Send to 127.0.0.1:8810-->b'[Leonard] : '
Receive from 127.0.0.1:6678-->Hi.
Send to 127.0.0.1:6996-->b'[Sheldon] : Hi.'
Send to 127.0.0.1:6678-->b'[Sheldon] : Hi.'
Send to 127.0.0.1:8810-->b'[Sheldon] : Hi.'
Receive from 127.0.0.1:6996-->Hi.
Send to 127.0.0.1:6996-->b'[Leonard] : Hi.'
Send to 127.0.0.1:6678-->b'[Leonard] : Hi.'
Send to 127.0.0.1:8810-->b'[Leonard] : Hi.'
Receive from 127.0.0.1:6678-->Hi.
Send to 127.0.0.1:6996-->b'[Sheldon] : Hi.'
Send to 127.0.0.1:6678-->b'[Sheldon] : Hi.'
Send to 127.0.0.1:8810-->b'[Sheldon] : Hi.'
```

## 4. 网络聊天室

- GUI客户端（不要求，加分项）
- Python的Tkinter库实现GUI界面，右图是一个范例。  
<https://wiki.python.org/moin/Tkinter>
- 高级的Qt <https://www.qt.io/qt-for-python>





恭喜你已完成实验