

实验名称：SOCKET 网络程序设计	
实验台号：	实验时间：
实验小组：张楷	
实验目的与要求： <ul style="list-style-type: none"> •理解进程通信的原理及通信过程； •掌握基本的网络编程方法； •需要有流程图； •需要抓取传输内容，比如“Hello World!”； •实现并发服务器。 	
实验环境说明： <p>PC, python3.8</p>	
实验过程、步骤（可另附页、使用网络拓扑图等辅助说明）及结果： <p>一、Python 的 socket 基本编程。</p> <p>1) 客户端</p>  <pre> import socket as sc client = sc.socket(sc.AF_INET, sc.SOCK_STREAM) #套接字类型AF_INET, socket.SOCK_STREAM tcp协议 client.connect(('127.0.0.1', 8080)) #与ip地址为192.168.1.1, 端口号为8080的端口建立连接。 msg = 'Hello World' client.send(msg.encode('utf-8'))#向服务器发送消息 data=client.recv(1024) print('recv', data.decode()) </pre> <p>2) 服务器端</p> <pre> import socket as sc server =sc.socket(sc.AF_INET, sc.SOCK_STREAM)#同客户端 server.bind(('127.0.0.1', 8080)) #绑定服务器ip地址和端口 server.listen(5) while 1: conn, add = server.accept() # 监听端口号 print(conn, add)#打印 while 1: msg = conn.recv(1024) #收到消息 print('服务器收到消息', msg.decode('utf-8'))#打印所受到的消息 break conn.send(str.encode('receive over'))#向客户端发送已接受的消息 conn.close()#关闭连接 </pre>	

3) 运行概念图。

服务器运行结果:

```
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 23:11:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/LENOVO/AppData/Local/Programs/Python/Python38/jw server.py =
<socket.socket fd=628, family=AddressFamily.AF_INET, type=SocketKind.SOCK_STREAM, proto=0, laddr=('127.0.0.1', 8080), raddr=('127.0.0.1', 59435)> ('127.0.0.1', 59435)
服务器收到消息 Hello World
```

用户端运行结果:

```
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 23:11:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/LENOVO/AppData/Local/Programs/Python/Python38/jw client.py =
recv receive over
>>>
```

二、并发服务器

代码实现

```
from socket import *
from threading import Thread

def recv_data(conn, add):
    print("客户端 {} 已经连接".format(add))
    # 接受数据
    msg='receive over'
    raw_data = conn.recv(1024)
    while raw_data:
        print(f"收到来自 {add} 的数据: {raw_data}")
        conn.send(msg.encode('utf-8'))
        raw_data = conn.recv(1024)
    conn.close()

server = socket(AF_INET, SOCK_STREAM)
    #套接字类型AF_INET, socket.SOCK_STREAM tcp协议
server.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
    #端口复用
server.bind(("127.0.0.1", 8080))
    #绑定IP和端口
server.listen(5)
    #监听客户端
while 1:
    conn, add=server.accept()
    p = Thread(target=recv_data, args=(conn, add))
    p.start()
    # 多线程共享一片内存区域, 所以这里不用关闭
```

运行概念：

```
客户端('127.0.0.1', 59668)已经连接
收到来自('127.0.0.1', 59668)的数据: b'Hello World'
客户端('127.0.0.1', 59672)已经连接
收到来自('127.0.0.1', 59672)的数据: b'Hello World'
客户端('127.0.0.1', 59676)已经连接
收到来自('127.0.0.1', 59676)的数据: b'Hello World'
客户端('127.0.0.1', 59681)已经连接
收到来自('127.0.0.1', 59681)的数据: b'Hello World'
```

实验总结（遇到的问题及解决办法、体会）：

Socket 编程很有趣下学期还想选老师的课。

器材、工具领用及归还负责人： 张楷

实验记录人：（签名）张楷

实验执笔人：（签名）张楷

报告协助人：（签名）张楷

小组成员签名：（签名）

张楷

验收人：

成绩评定：