**Lab6 WebServer**

**Code for the WebServer：**

**from socket import \***

**import datetime**

**serverSocket = socket(AF\_INET, SOCK\_STREAM) # Prepare a sever socket**

**# Fill in start**

**serverPort = 13000**

**serverSocket.bind(('', serverPort))**

**serverSocket.listen(5)**

**# Fill in end**

**while True:**

**# Establish the connection**

**print('Ready to serve...')**

**connectionSocket, addr = serverSocket.accept()**

**print("addr:\n", addr)**

**# Fill in start**

**# Fill in end**

**try:**

**message = connectionSocket.recv(1024) # Fill in start #Fill in end**

**print**

**"message: \n", message**

**filename = message.split()[1]**

**f = open(filename[1:])**

**output\_data = f.read()**

**print("outputdata:", output\_data)**

**now = datetime.datetime.now()**

**# Fill in start #Fill in end**

**# Send one HTTP header line into socket**

**# Fill in start**

**first\_header = "HTTP/1.1 200 OK"**

**# alive ={**

**# "timeout":10,**

**# "max":100,**

**# }**

**header\_info = {**

**"Date": now.strftime("%Y-%m-%d %H:%M"),**

**"Content-Length": len(output\_data),**

**"Keep-Alive": "timeout=%d,max=%d" % (10, 100),**

**"Connection": "Keep-Alive",**

**"Content-Type": "text/html"**

**}**

**following\_header = "\r\n".join("%s:%s" % (item, header\_info[item]) for item in header\_info)**

**print("following\_header:", following\_header)**

**connectionSocket.send("%s\r\n%s\r\n\r\n" % (first\_header, following\_header))**

**# Fill in end**

**# Send the content of the requested file to the client**

**for i in range(0, len(output\_data)):**

**connectionSocket.send(output\_data[i])**

**connectionSocket.close()**

**except IOError:**

**# Send response message for file not found**

**# Fill in start**

**connectionSocket.send(**

**"HTTP/1.1 404 Not Found\r\n"**

**"Content-Type: text/html\r\n\r\n"**

**"<!doctype html><html><body><h1>404 Not Found<h1></body></html>")**

**# Fill in end**

**# Close client socket**

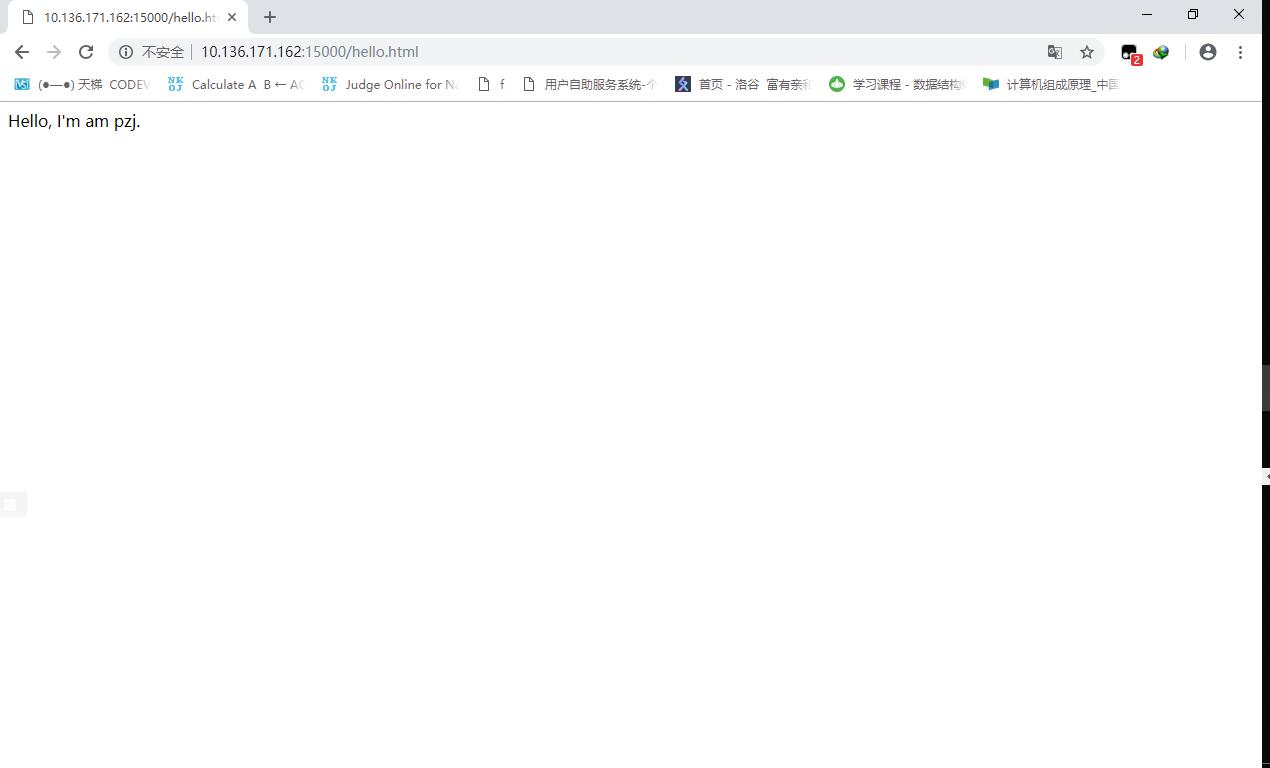
**# Fill in start**

**connectionSocket.close()**

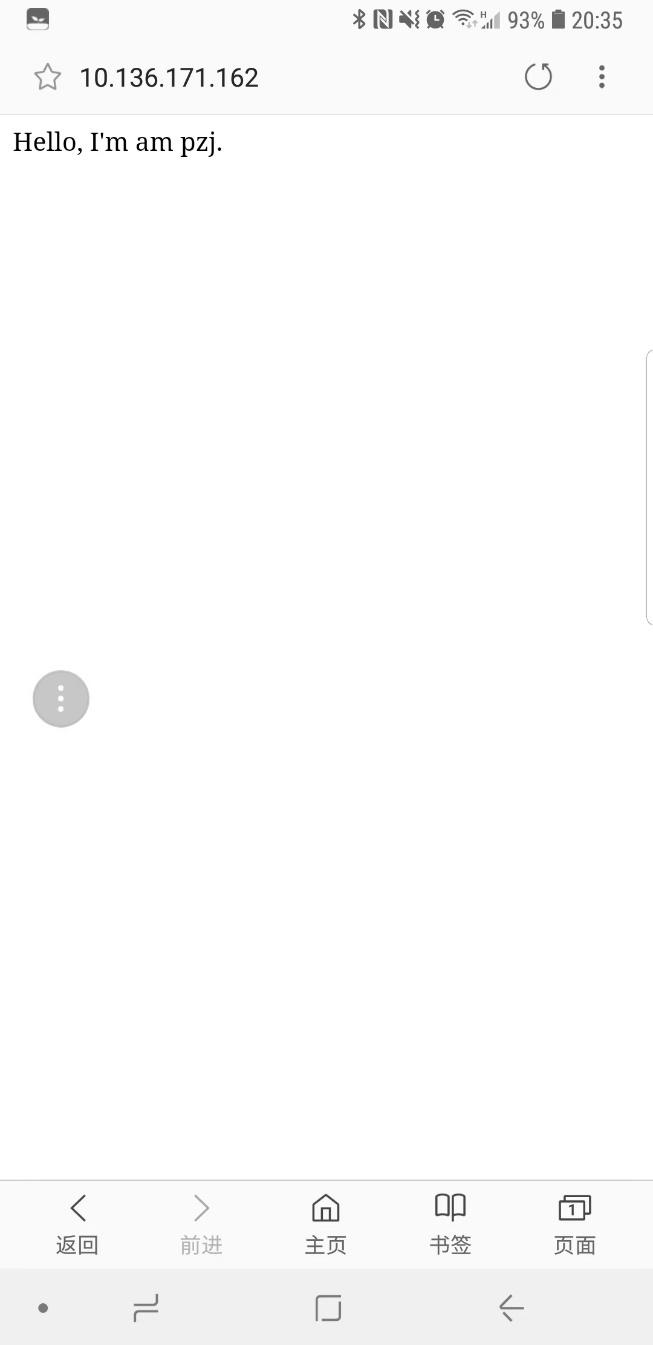
**# Fill in end**

**serverSocket.close()**

**There are some screenshots when I connect this server**

****

**(On the broswer)**

****

**(On my phone’s broswer)**

**Q1. MultiThreading Code：**

**# import socket module**

**from socket import \***

**import threading**

**import time**

**import sys # In order to terminate the program**

**def server(connection\_socket, addr):**

**try:**

**message = connection\_socket.recv(1024).decode()**

**filename = message.split()[1]**

**f = open(filename[1:])**

**output\_data = f.read()**

**# Send one HTTP header line into socket**

**header = 'HTTP/1.1 200 OK\r\nConnection: Keep-Alive\r\nContent-Type: text/html\r\n\r\n'**

**connectionSocket.send(header.encode())**

**# Send the content of the requested file to the client**

**for i in range(0, len(output\_data)):**

**connectionSocket.send(output\_data[i].encode())**

**connectionSocket.send("\r\n".encode())**

**time.sleep(15)**

**connectionSocket.close()**

**except IOError:**

**# Send response message for file not found**

**header = 'HTTP/1.1 404 NotFound\r\n\r\n'**

**connectionSocket.send(header.encode())**

**# Close client socket**

**connectionSocket.close()**

**if \_\_name\_\_ == '\_\_main\_\_':**

**serverSocket = socket(AF\_INET, SOCK\_STREAM)**

**# Prepare a server socket**

**serverSocket.bind(('', 15000))**

**serverSocket.listen(10)**

**while True:**

**# Establish the connection**

**print('Ready to serve...')**

**connectionSocket, addr = serverSocket.accept()**

**thread = threading.Thread(target=server, args=(connectionSocket, addr))**

**thread.start()**

**serverSocket.close()**

**sys.exit() # Terminate the program after sending the corresponding data**

**Q2. My own client code:**

**from socket import \***

**import sys**

**server\_host = sys.argv[1]**

**server\_port = sys.argv[2]**

**filename = sys.argv[3]**

**try:**

**client\_socket = socket(AF\_INET, SOCK\_STREAM)**

**client\_socket.connect((server\_host, int(server\_port)))**

**header = {**

**"first\_header": "HTTP/1.1 200 OK",**

**"Accept": "text/html",**

**"Accept-Language": "en-us",**

**}**

**http\_header = "\r\n".join("%s:%s" % (item, header[item]) for item in header)**

**print(http\_header)**

**client\_socket.send("%s\r\n\r\n" % (http\_header))**

**except IOError:**

**sys.exit(1)**

**final = ""**

**response\_message = client\_socket.recv(1024)**

**while response\_message:**

**final += response\_message**

**response\_message = client\_socket.recv(1024)**

**client\_socket.close()**

**print("final:", final)**