Assignment #AS01 announced on 15 Nov 2022, due at 29/30 Nov 2022

Upload **your server code (.py, .ipynb, or .java)** to the homework submission system. Note that you **MUST write comments** within the codes to explain your codes. Code without comments get zero points.

Q0. Learn socket programming.

0-1. Watch "Berkeley socket programming" video (https://www.youtube.com/watch?v=y5Q4Mc0eO14)

0-2. Read Wiki page "Berkeley sockets" (https://en.wikipedia.org/wiki/Berkeley_sockets).

0-3. Get python/java examples and check socket APIs. You can use one of the language to finish this assignment.
Python server:
https://github.com/hsiaom26/Socket/blob/master/TCP_Socket_Server.ipynb
Python client:
https://github.com/hsiaom26/Socket/blob/master/TCP_Socket_Client.ipynb
Python socket API:
https://docs.python.org/3/library/socket.html
Java server:
https://github.com/hsiaom26/Socket/blob/master/EchoServer.java
Java client:
https://github.com/hsiaom26/Socket/blob/master/EchoClient.java
Java socket API:
https://docs.oracle.com/javase/tutorial/networking/sockets/index.html

Q1. Modify the provided server code from the example and make it a HTTP server that supports the following functions.

- The server runs at **TCP port 8888** on **127.0.0.1** IP address.
- Implement **persistent HTTP** that can transmit **at most two objects** in a single TCP connection.
- Implement a **multi-thread** HTTP server.
- The server can accept a HTTP request with request URL ‘**good.html**’. The server can reply a HTTP reply message with **200 OK** (and any necessary HTTP headers) and an html message containing a simple html as follows: <html><head><link href=”style.css” rel=”stylesheet” type=”text/css”></head><body>good</body></html>
- The server can accept a HTTP request with request URL ‘**style.css**’. The server can reply a HTTP reply message with **200 OK** (and any necessary

HTTP headers, note that its **Content-Type** header must be **text/css**) and an html message containing a simple css file as follows:
Body {color: red;}

- The server can accept a HTTP request with request URL '**redirect.html**'. The server can reply a HTTP reply message with **301 Moved Permanently** and any necessary HTTP headers (including **Location:**). The redirected page is good.html.
- The server can accept a HTTP request with request URL '**notfound.html**'. The server has no such file so it replies a HTTP reply message with **404 Not Found** and any necessary HTTP headers.

Note:

- You only need to upload your server code as a single file to homework submission system.
- TA will use a commercial browser (such as safari, edge and chrome) to connect to your server and see if the expected behavior occurs, i.e., showing the correct good.html (with red word), redirecting from redirected.html to good.html, and showing a 404 on browser.
- Your server should be runnable without any user input. TA will use a script to automatically execute your server codes. You should test your server codes in the same way. Otherwise, no points will be given.