

Introduction to Computer Networks

Assignment 2: Web Server Imitation

소프트웨어학과 김승현

2014310279

1. Development environments

```
(base) → asmt1 sw_vers
ProductName:   Mac OS X
ProductVersion: 10.14.6
BuildVersion:  18G87
(base) → asmt1 python -V
Python 3.7.1
```

2. How to test program

- Decompress 'server.py', 'index.html', 'secret.html', 'cookie.html' and other files to same directory
- Enter 'python3 server.py' in terminal to run server

```
(base) → asmt2 python3 server.py
The server is ready to receive.
```

- Through internet browser(google chrome), go to 'http://ip_address:10080'
- Enter any id and password, then access to 'cookie.html'
- Request other object files until cookie expires

3. Design

a. my_server()

```
def my_server():
    server_host, server_port = '', 10080
    address = (server_host, server_port)
    server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    server_socket.bind(address)
    server_socket.listen(200)
    print('The server is ready to receive.')

    try:
        while True:
            connection_socket, client_addr = server_socket.accept()
            print('Connected to : ', client_addr[0], ':', client_addr[1])
            my_thread = threading.Thread(
                target=req_handler, args=(connection_socket,))
            my_thread.start()

    except KeyboardInterrupt:
        server_socket.close()
```

과제 1과 같이 thread를 이용해 다중 접속, 연속된 request를 처리한다. 서버가 연결 된 후, 들어온 request에 대해서 req_handler()함수로 전달하여 처리한다.

b. req_handler()

- 1) 연결된 socket으로부터 전달된 request를 parsing하여 어떤 요청인지를 체크한다.

```
def req_handler(connection_socket):
    msg = connection_socket.recv(1024).decode()
    request = msg.split(' ')
    req_method = request[0]
    print("[method : ", req_method, ']')
```

- 2) Index.html에 id 및 password를 입력 시 자동으로 Post method를 통해 새로운 요청이 전달되며 이를 msg_parser() 함수를 통해 parsing 하여 사용자가 입력한 id를 알아내고 이 id와 접속한 시간을 datetime을 이용해 구한 후 Set-Cookie 헤더를 통해 cd(cookie id), expired 라는 변수에 기록한다.

```
elif(req_method == 'POST'):
    file_name = request[1]
    if(file_name == '/index.html'):
        user_id = msg_parser(msg, 'id')
        if(user_id == -1):
            send_error(connection_socket, 403, 'User not found')
        print("[User_ID] : ", user_id)
        header = 'HTTP/1.1 302 FOUND\r\nLocation:/secret.html\r\nContent-Type:text/html\r\nSet-Cookie:cd={} \r\n'.format(user_id)
        now = datetime.datetime.now()
        cur = now.minute*60 + now.second
        header += 'Set-Cookie:expired={} \r\n\r\n'.format(cur)
        send_data(connection_socket, header)
```

- 3) Send Data

이후 들어온 file요청에 대해서는 request내 file name의 확장자(extension)를 체크하여 파일을 보내준다.

```
extension = file_name[file_name.rfind(".")+1:]
if extension == "jpeg":
    header = ('HTTP/1.1 200 OK\r\nContent-Type:image/jpeg\r\n\r\n')
elif extension == "html":
    header = ('HTTP/1.1 200 OK\r\nContent-Type:text/html\r\n\r\n')
elif extension == "mp4":
    header = ('HTTP/1.1 200 OK\r\nContent-Type:video/mp4\r\n\r\n')
elif extension == "pdf":
    header = ('HTTP/1.1 200 OK\r\nContent-Type:application/pdf\r\n\r\n')
else:
    header = ('HTTP/1.1 200 OK\r\nContent-Type:application/octet-stream\r\n\r\n')
send_data(connection_socket, header)
while(True):
    data = f.read(1024)
    if not data:
        break
    connection_socket.sendall(data)
```

요청한 파일이 없을 경우 404 error를 보낸다.

```
except FileNotFoundError:  
    send_error(connection_socket, 404, 'File not found')
```

4) Cookie Check

Cookie 만료를 위해서는 먼저 주어진 요청에서 cookie에서 로그인시 쿠키에 입력한 expired 변수의 값(로그인한 시간)을 찾아 is_expired()에 전달한다.

```
exp_time = msg_parser(msg, 'exp')  
exp = is_expired(exp_time)
```

이후 is_expired함수에서 현재 시간과 로그인 당시의 시간의 차를 구해서 30에서 이 차를 빼주어 쿠키 만료까지 남은 시간을 구한다.

```
def is_expired(exp_time):  
    if(exp_time == 0):  
        return -1  
    now = datetime.datetime.now()  
    cur = now.minute*60 + now.second  
    exp = 30 - (cur - int(exp_time))  
    return exp
```

남은 시간이 0 이하일 경우 403 error를 보낸다.

```
if(exp <= 0):  
    send_error(connection_socket, 403, 'you are logged out')
```

5) Cookie.html

Cookie.html에서는 위에서 입력한 cookie 내의 id와 만료까지 남은 시간(exp)를 cookie.html에 인자로 전달하여 이를 보여준다.

```

if(exp <= 0):
    send_error(connection_socket, 403, 'you are logged out')
else:
    f = open(file_name[1:])
    print("[FILE OPENED] : ", file_name)
    data = f.read().format(user_id = user_id, time = exp).encode()
    header = ('HTTP/1.1 200 OK\r\nContent-Length: {}\r\nContent-Type:text/html\r\n\r\n').fo
    send_data(connection_socket, header, True, data)

```

c. send_error()

error의 경우 직접 html을 작성하여 보내주는 방법을 택하였다.

1) Message Format

Message format은 python module중 하나인 HTTPRequestHandler에서 사용하는 error message format을 참조하여 작성하였다.

2) Send_error()

Message format html 형식에 error code와 explain을 전달하여 전송하는 방식을 사용하였다.

```

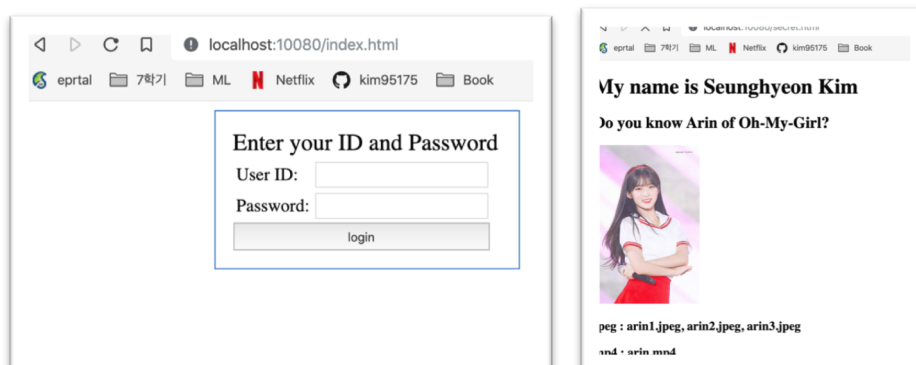
def send_error(conn, code, explain):

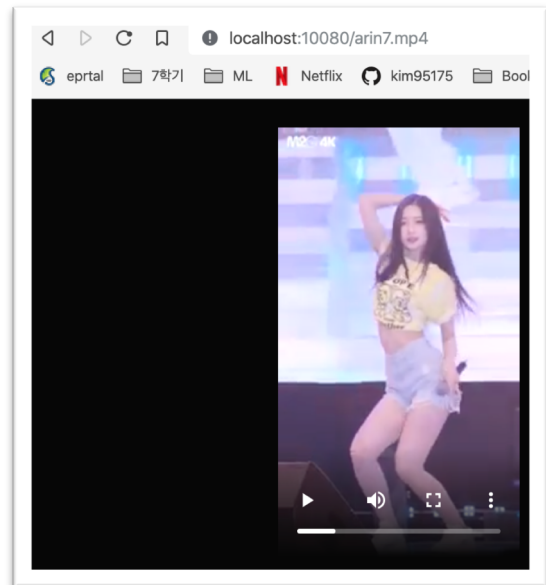
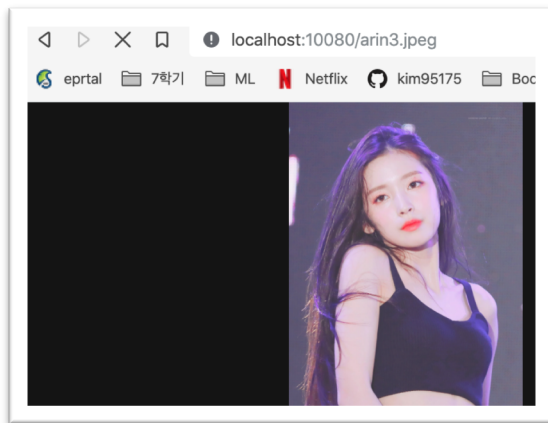
    if code == 403:
        content = error_message.format(code = code, detail = 'Forbidden',explain = explain)
        data = content.encode('UTF-8')
        header = 'HTTP/1.1 403 Forbidden\r\nSet-Cookie:cd={}\r\nSet-Cookie:expired=0\r\n\r\n'.format(None)
    elif code == 404:
        content = error_message.format(code = code, detail = 'Not Found',explain = explain)
        data = content.encode('UTF-8')
        header = 'HTTP/1.1 404 Not Found\r\nContent-Length: {}\r\n\r\n'.format(len(data))
    send_data(conn, header, True, data)

```

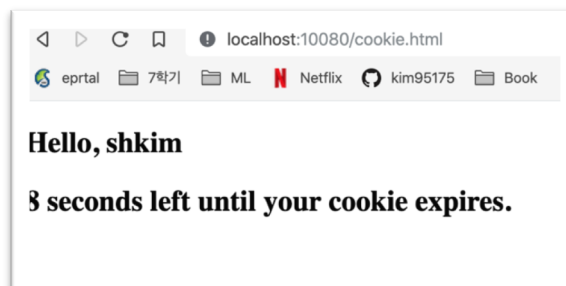
4. Evaluation Scenarios

a. 'index.html', 'secret.html' and other object files

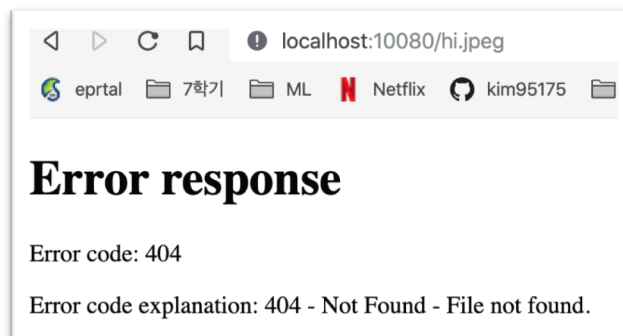




b. 'cookie.html'



c. 404 File Not Found Error when a file is not present



d. 403 Forbidden error when 1) cookie expires 2) or not login

