

네트워크 프로그래밍

과제 3

<멀티 프로세스로 동작하는 간단한 웹 서버 제작>

학과	스마트시스템소프트웨어학과
학번	20160372
이름	방혜원
담당교수님	노동건 교수님

1. 구현 방식

web.c 파일에 웹 서버를 구현하였다.
포트 번호를 설정하여 server를 실행할 수 있도록 하였다.
서버는 listen상태로 두고, accept 시 멀티 프로세스로 실행을 위하여 fork를 한다. 이후 자식 프로세스에서 request_handler가 수행되도록 하였다. (서버는 클라이언트가 중단되어도 죽지 않고, 계속해서 요청을 받아들인다.)

<주요 함수>

request_handler() 함수에서는 소켓을 기반으로 표준 입출력 함수를 사용해주기 위하여 읽기모드의 FILE 포인터와 쓰기모드의 FILE 포인터를 생성해준다. 소켓에서 보내온 GET 요청을 읽어들이고, 요청된 파일이 존재하는지 아닌지를 판별하여 send_error() 함수 혹은 send_data() 함수로 이동할 수 있도록 하였고, 존재하는 파일이라면 파일의 타입을 구할 수 있도록 content_type() 함수를 실행한다.

- 요청한 파일이 존재하지 않을 경우 : send_error()
- 요청한 파일이 존재할 경우 : send_data(), content_type()
- 파일 요청 공간이 비어있을 경우 : file_name을 index.html로 하여 send_data() 함수로 이동

send_data() 함수에서는 상태 코드는 200으로 하고, 요청된 파일의 타입을 입력하고, 파일의 크기를 구한다. HTTP header 정보와 요청한 파일을 소켓 파일 디스크립터에 올려 클라이언트로 전송한다.
요청한 파일이 open되지 않는다면 send_error()함수를 수행한다.

send_error() 함수에서는 요청한 파일이 존재하지 않기 때문에 상태 코드는 404로 하고, file_name을 error.html로 지정한다.
error.html의 타입은 text이므로 타입은 text/html, 파일의 크기를 구하여 HTTP header 정보와 파일을 소켓 파일 디스크립터에 올려 클라이언트로 전송한다.

만약, error.html이 존재하지 않는다면, content를 기입해주어 error occur이라는 메시지가 화면에 보여질 수 있도록 하였다.

content_type() 함수에서는 파일 이름에서 '.' 뒤의 내용을 보고 파일의 종류를 판단하고, 그 종류에 맞는 값을 return 해준다.

2. 처리 가능한 파일

html, png, jpg, jpeg를 처리할 수 있다.

```
.html => text/html  
.png => image/png  
.jpg => image/jpeg  
.jpeg => image/jpeg
```

cherry.png, index.html, error.html, index.jpeg, JY.jpg 파일을 가지고 과제를 수행하였다.