C only web server

과목명: 컴퓨터 네트워크

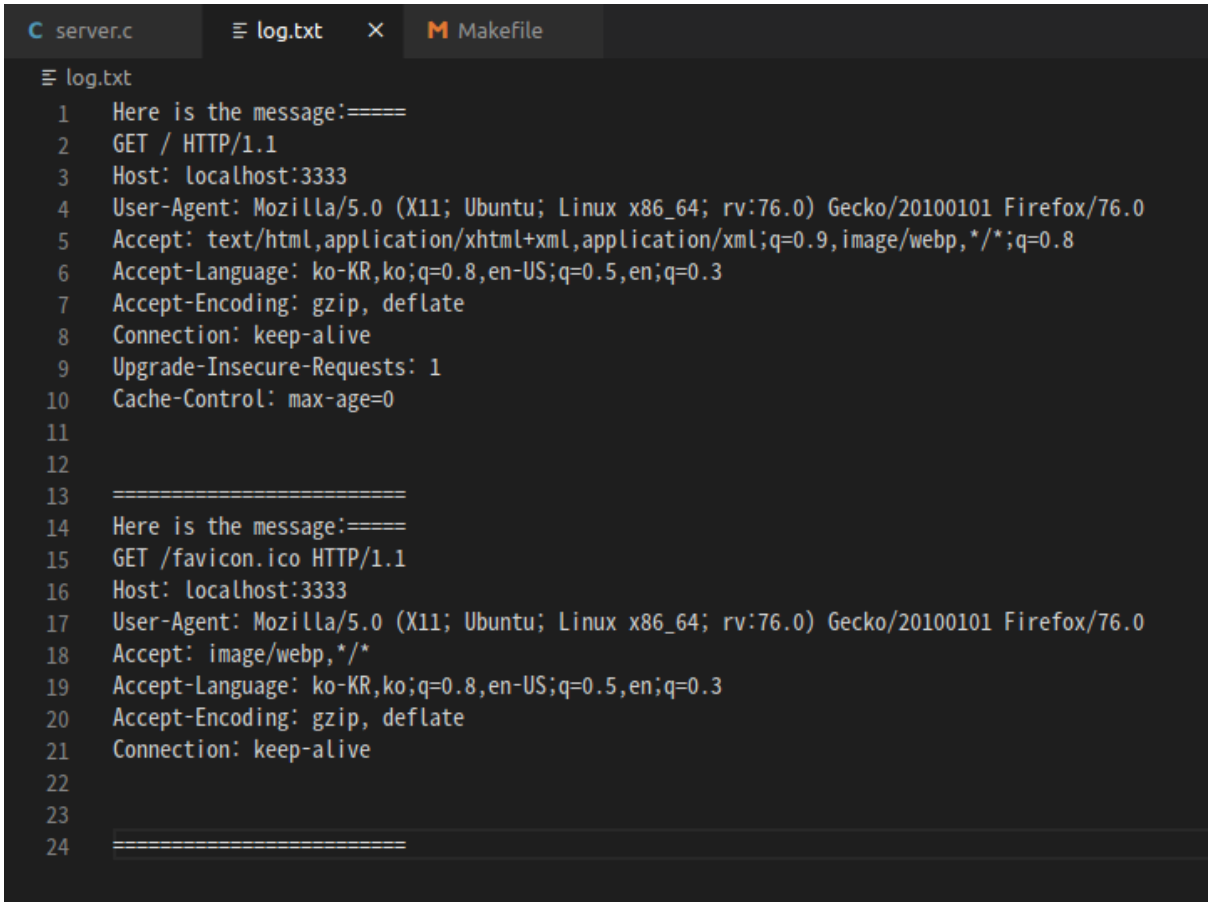
교수명: 은하수

학과명: 소프트웨어학부

학번: 2018044457

이름: 김재훈

Part Ⅰ Client의 request message를 화면에 출력하는 "Web server" 제작



서버에 전달된 request메시지를 각 filed가 무엇을 의미하는지 분석해보았다.

GET / HTTP/1.1

Host: localhost:3333 // request header start

User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:76.0) Gecko/20100101 Firefox/76.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Language: ko-KR,ko;q=0.8,en-US;q=0.5,en;q=0.3

Accept-Encoding: gzip, deflate // request header end

Connection: keep-alive // general header start

Upgrade-Insecure-Requests: 1

Cache-Control: max-age=0 // general header

첫 줄은 요청 방식이 GET임을 나타내고, /파일(보통 index.html)을 요청하고 있으며, HTTP/1.1버전의 형식임을 의미한다.

두번째 줄은 요청한 url이 localhost:3333임을 의미한다.

세번째 줄은 클라이언트가 mozilla/5.0과 호환되는 웹 브라우저를 사용하고 있으며, 사용자 시스템 정보는 X11; Ubuntu; Linux x86_64; rv:76.0이고, Gecko기반의 브라우저인 Firefox 76.0버전을 사용하여 요청을 보냈음을 의미한다.

네번째 줄은 클라이언트에서 text타입의 html파일, application타입의 xhtml+xml파일, 0.9의 우선순위로 인식되는 application타입의 xml파일, image타입의 webp파일, 0.8의 우선순위로 인식되는 다른 모든 타입의 파일을 받아들일 수 있다는 의미이다.

다섯번째 줄은 클라이언트에서 ko-KR, 0.8의 우선순위로 ko, 0.5의 우선순위로 en-US, 0.3의 우선순위로 en 형태의 언어를 이해할 수 있음을 의미한다.

여섯번째 줄은 클라이언트에서 gzip, deflate형태의 압축 알고리즘으로 인코딩된 컨텐츠들을 받을 수 있다는 의미이다.

일곱번째 줄은 현재 전송이 완료된 후 네트워크 접속을 계속 유지하라는 의미이다.

여덟번째 줄은 클라이언트가 더 안전한 h형태의 연결을 지원한다는 것을 서버에 알리고, 서버가 해당 형태의 연결로 리디렉션할 수 있도록 한다.

아홉번째 줄은 클라이언트의 캐싱 방식을 0초 후에 응답캐시가 만료됐다고 인식한다는 의미한다

Entity header는 본문이 없어 포함되지 않은 듯 하다.

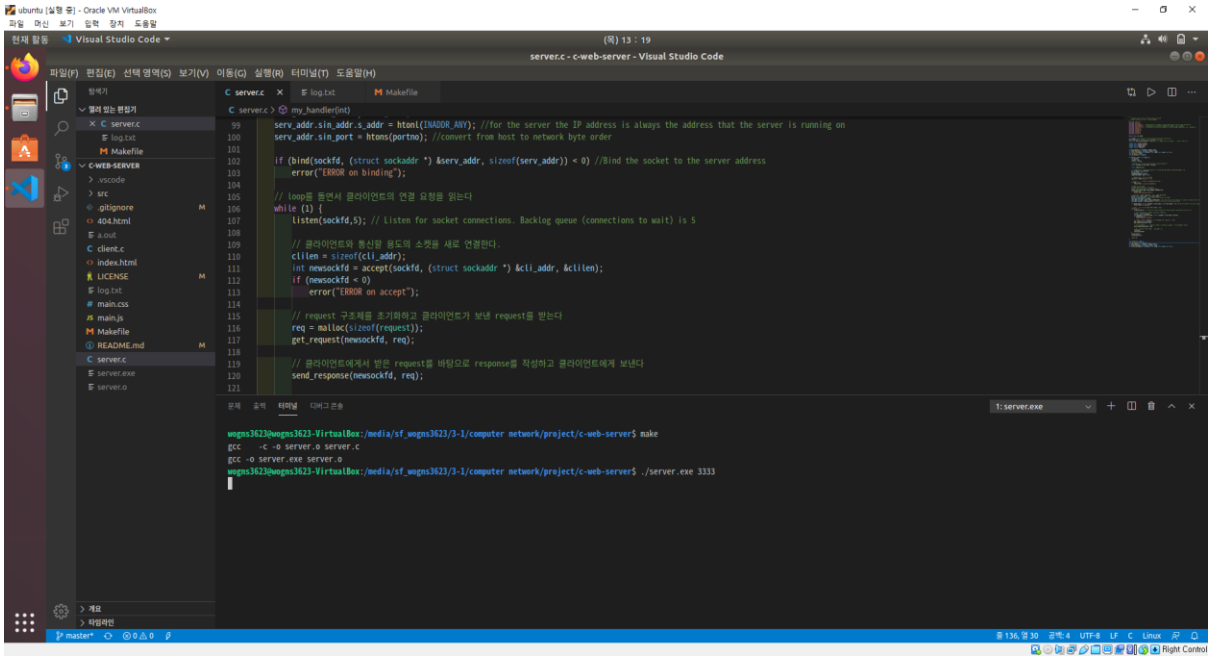
Part Ⅱ Web server가 browser의 request에 response할 수 있도록 확장

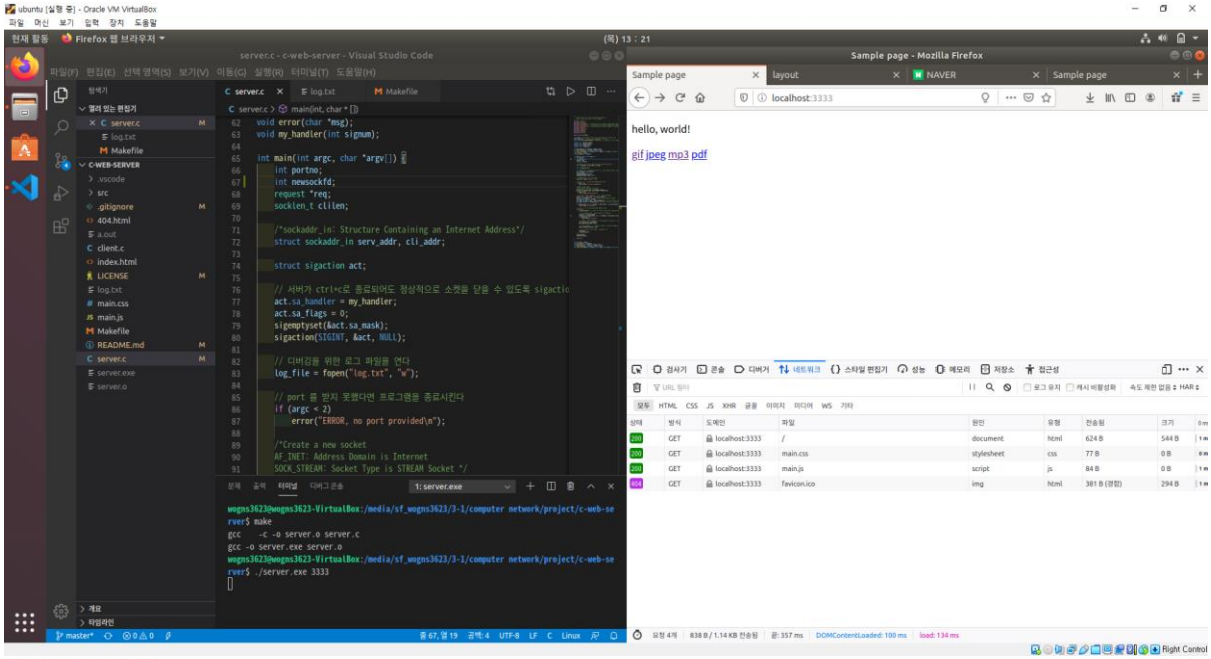
서버는 대략적으로 아래의 순서로 동작한다

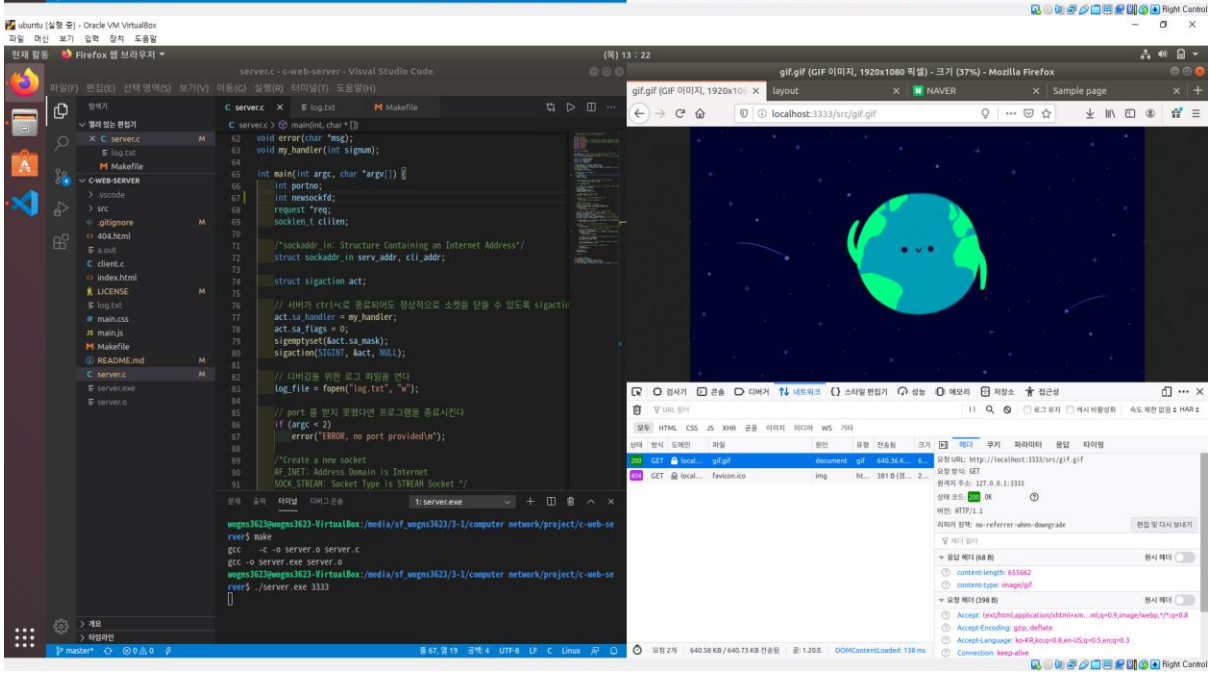
1. 서버 소켓 생성 (sockfd = socket())
2. 입력받은 포트 번호를 이용해 ip주소와 포트 번호를 소켓에 연결 (bind())
3. 클라이언트의 연결 요청 대기 (listen())
4. 클라이언트와 연결되면 통신용 소켓 생성 (newsockfd = accept())
5. Request를 전달받고 해석 (get_request())
6. 전달받은 request를 바탕으로 response작성 (send_response())
7. 클라이언트와의 통신용 소켓 연결종료 (close(newsockfd))
8. 서버를 닫지 않는다면 3번으로 돌아감

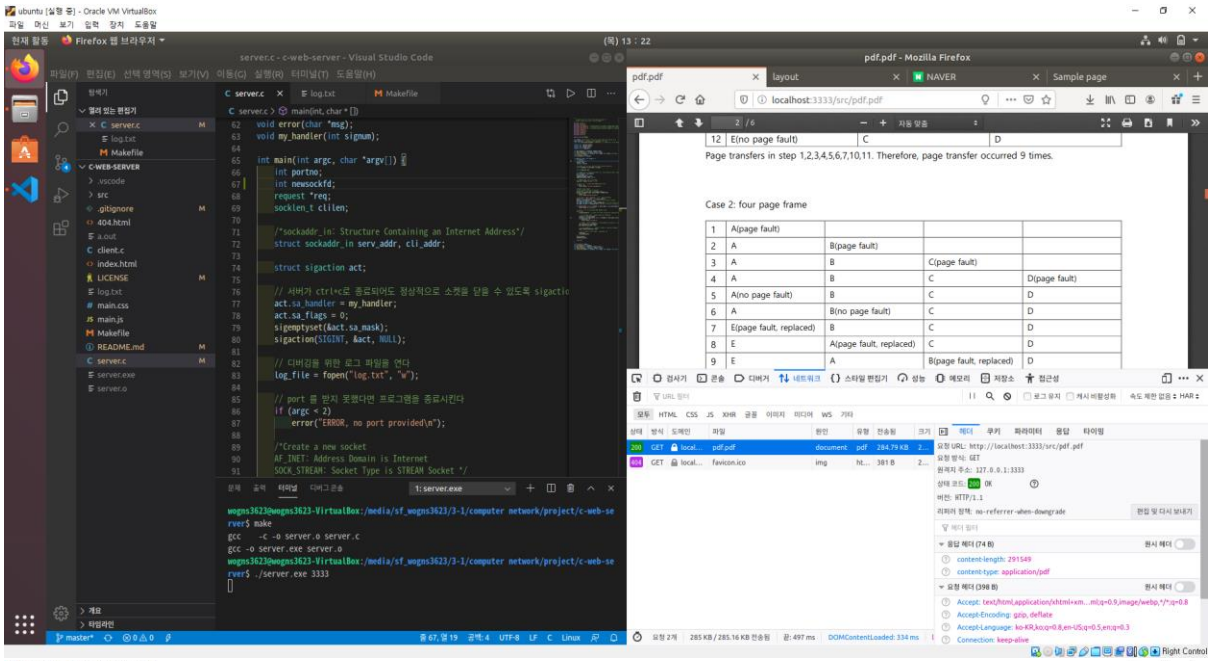
구현중 response를 작성할 때 헤더의 종류가 어떤 것들이 있고, 어떤 헤더를 필수적으로 넣어야 하는지를 몰라 많이 헤멨다. MDN web docs 등을 찾아보며 어떠한 헤더가 있는지, response에는 어떤 해더가 들어가는지에 대해 공부해 서버를 동작시키는데 성공했다.

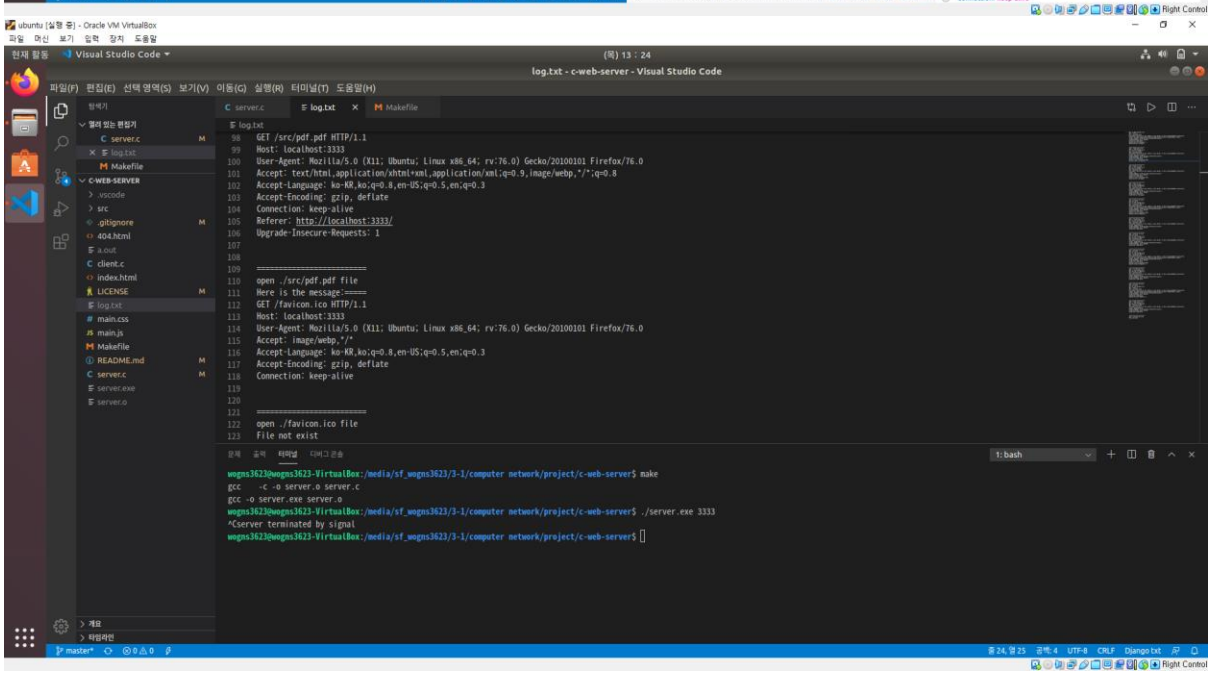
동작 예시




ubuntu [실행 중] - Oracle VM VirtualBox
server.c - c-web-server - Visual Studio Code
Sample page - Mozilla Firefox
hello, world!
gif jpeg mp3 pdf
wogns3623@wogns3623-VirtualBox:/media/sf_wogns3623/3-1/computer network/project/c-web-server$ make
gcc -c -o server.o server.c
gcc -o server.exe server.o
wogns3623@wogns3623-VirtualBox:/media/sf_wogns3623/3-1/computer network/project/c-web-server$ ./server.exe 3333


ubuntu [실행 중] - Oracle VM VirtualBox
server.c - c-web-server - Visual Studio Code
gif.gif (GIF 이미지, 1920x1080 픽셀) - 크기 (37%) - Mozilla Firefox
wogns3623@wogns3623-VirtualBox:/media/sf_wogns3623/3-1/computer network/project/c-web-server$ make
gcc -c -o server.o server.c
gcc -o server.exe server.o
wogns3623@wogns3623-VirtualBox:/media/sf_wogns3623/3-1/computer network/project/c-web-server$ ./server.exe 3333


server.c - c-web-server - Visual Studio Code
pdf.pdf - Mozilla Firefox
Page transfers in step 1,2,3,4,5,6,7,10,11. Therefore, page transfer occurred 9 times.
Case 2: four page frame


log.txt - c-web-server - Visual Studio Code