

컴퓨터 네트워크

2016003618 송현수

프로젝트의 목표

1. c언어를 이용하여 웹서버를 구현
2. 서버와 웹브라우저의 동작 방식에 대해 학습

Part1.

서버로 전달된 request message

```
song@song-VirtualBox:~/Desktop/ClientServer_Example$ ./server 8080
Here is the message:
GET /index.html HTTP/1.1
Host: 127.0.0.1:8080
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
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

'헤더 필드이름 : 필드 값'의 구조로 이루어져 있다.

HOST : 127.0.0.1:8080

HOST는 요구된 리소스의 호스트가 127.0.0.1:8080 라는 것을 특정한다.

User-Agent : Mozilla/5.0 --- Firefox/76.0

브라우저마다 웹사이트를 제공하는 것은 힘들다. 따라서 User-Agent를 사용해서 브라우저를 알아낸다.

Accept : text/html, application/---

Accept 필드는 요청에 대한 응답으로 허용되는 타입의 목록을 표시할 수 있다. 클라이언트가 이해 가능한 콘텐츠 타입이 무엇인지를 알려주는 것이다.

Accept-Language : ko-KR,--

요청에 대한 응답으로 어떤 언어를 클라이언트가 이해할 수 있는지, 어떤 것이 더 선호되는 자연 언어들을 나열한다.

Accept-Encoding : gzip, deflate

클라이언트가 이해 가능한 콘텐츠 압축 방식이 무엇인지 알려준다.

Connection : keep-alive

Connection 헤더는 전송이 완료된 후 네트워크 접속을 유지할지 말지를 제어한다.
keep-alive이라서 연결이 끊기지 않고 지속된다.

Upgrade-Insecure-Requests : 1

암호화되고 인증된 응답에 대한 클라이언트의 선호도를 나타내는 신호를 서버에 전송한다. 그리고 CSP 지시문을 성공적으로 처리할 수 있다는 신호를 보낸다.

CSP : Content Security Policy

Cache-Control : max-age = 0

여러 가지 캐싱 옵션들을 줄 수 있다.

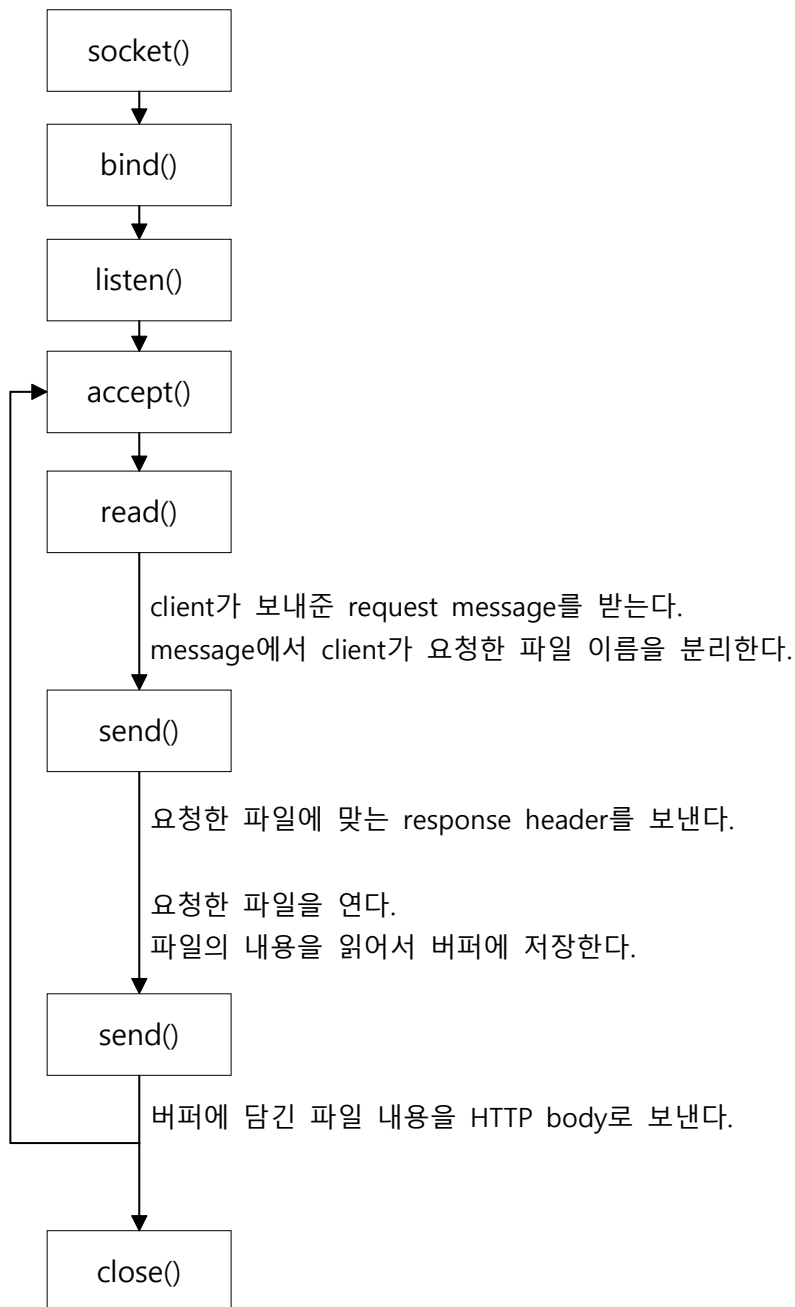
max-age는 리소스가 최신상태라고 판단할 최대 시간을 지정하는 것이다.

출처 (RFC 1945에서 찾지 못한 것은 밑에 사이트에서 찾아서 작성했다.)

<https://tools.ietf.org/html/rfc1945>

<https://developer.mozilla.org/ko/docs/Web/HTTP/Headers>

Part2. 서버 디자인

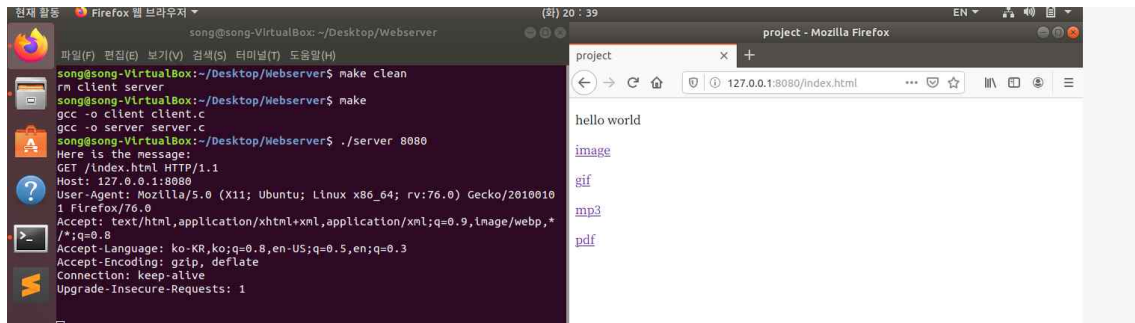


서버 구현 시 어려웠던 점과 해결 방법

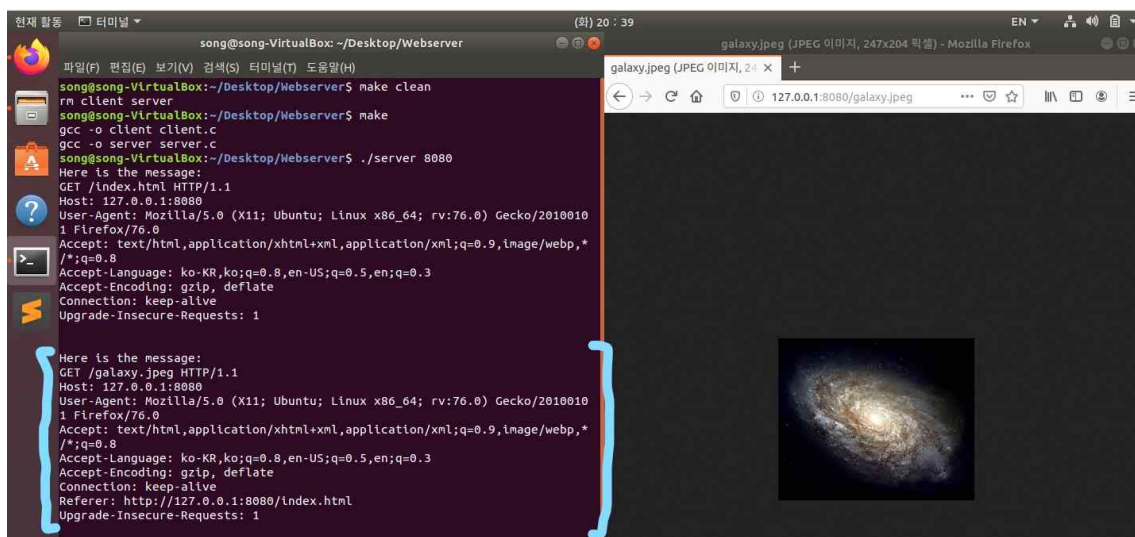
서버 구현 시에 가장 어려웠던 점은 바이너리 파일을 클라이언트로 보내는 것이었다. 처음 구상했던 방식은 헤더와 바디를 `strcat()` 함수를 이용해 합쳐서 한 번에 보낼 계획이었다. 하지만 일반 html 파일과는 달리 이미지나 노래 같은 바이너리 파일을 보낼 때는 제대로 전송이 되지 않았다. 이유를 정확히 찾지는 못했지만 아마도 버퍼를 배열로 만들어서 보낼 때 조금이라도 뒤에 빈 곳이 있으면 오류가 생기는 것처럼 보였다. 따라서 모든 버퍼를 배열이 아닌 포인터로 선언해서 빈 곳 없이 정확히 크기만큼 메모리를 할당해서 리스폰스 헤더와 바디를 따로따로 보내는 것으로 해결하였다.

동작 예시

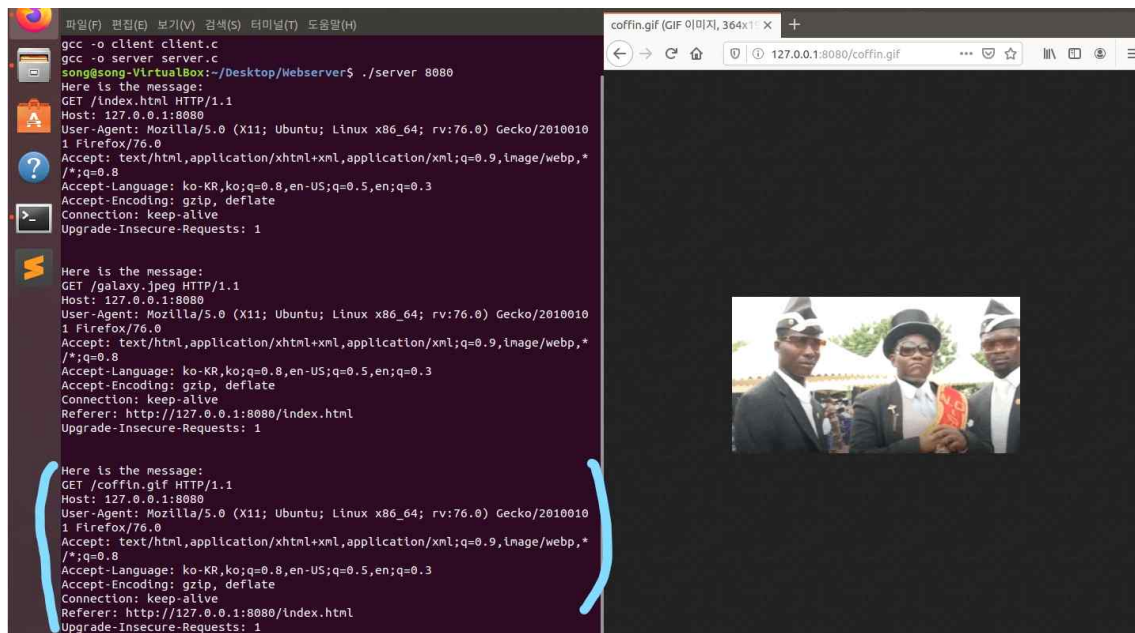
1. 처음 동작



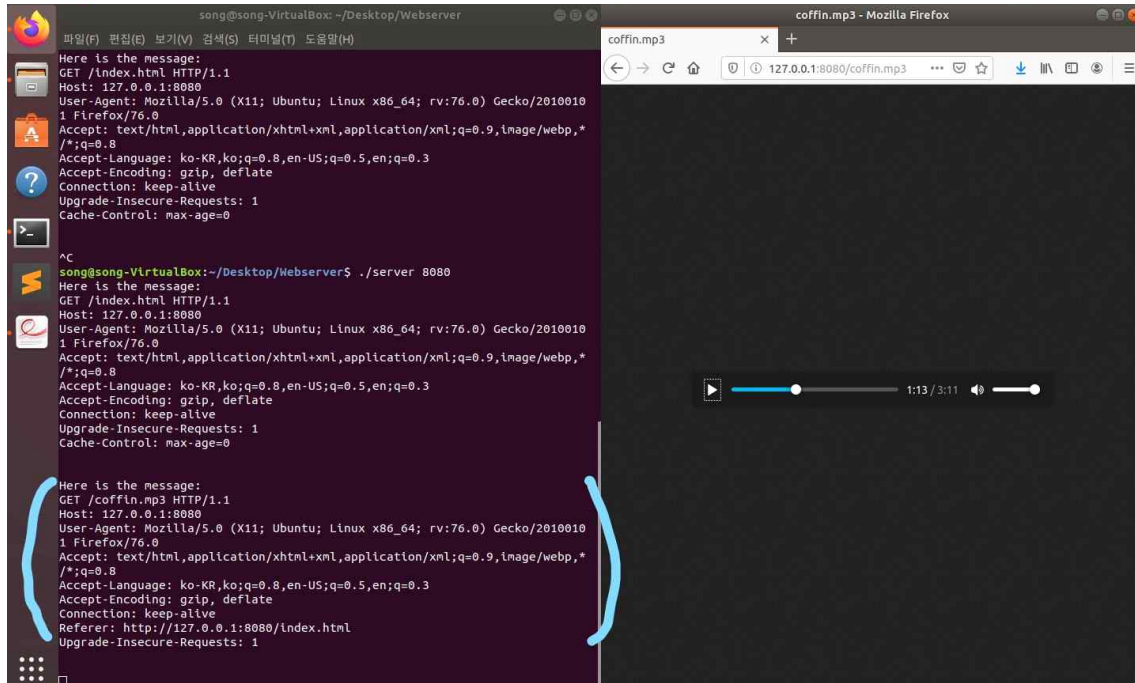
2. 브라우저의 image를 눌렀을 때



3. 브라우저의 gif를 눌렀을 때



4. 브라우저에서 mp3를 눌렀을 때



5. pdf를 눌렀을 때

