

시스템프로그래밍실습

assignment 2-2

학 과: 컴퓨터정보공학부

담당교수: 이기훈 교수님

학 번: 2019202050

성 명: 이강현

1. Introduction

protocol 이 무엇인지 알아보고 그중 http protocol의 개념과 request, response format에 대해 공부한다.

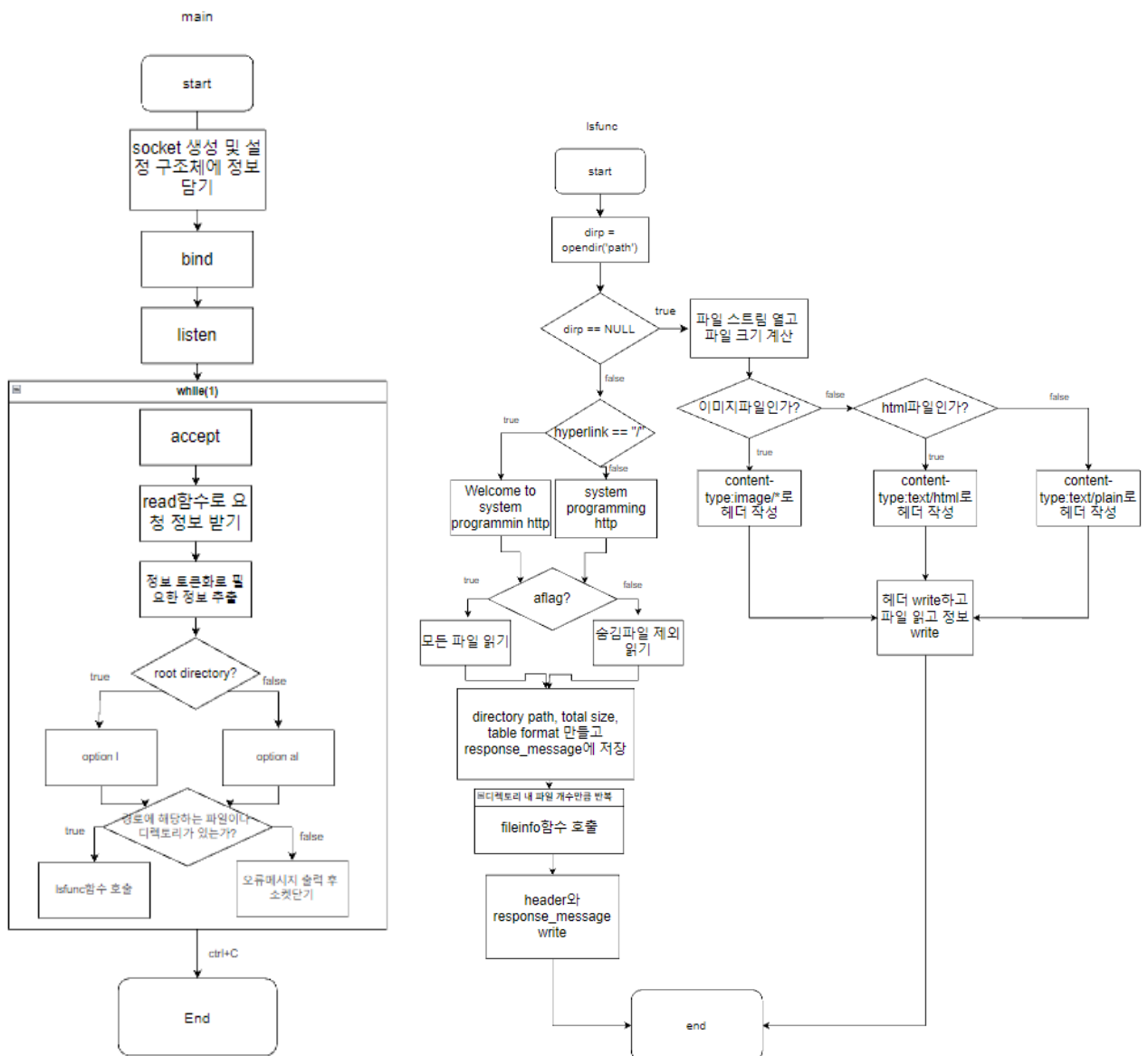
socket programming이 무엇인지 알고 이에 사용되는 여러가지 함수들과 구조체들을 배운다.

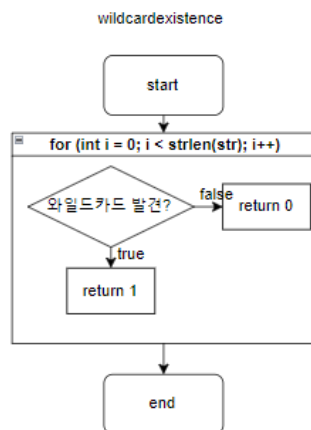
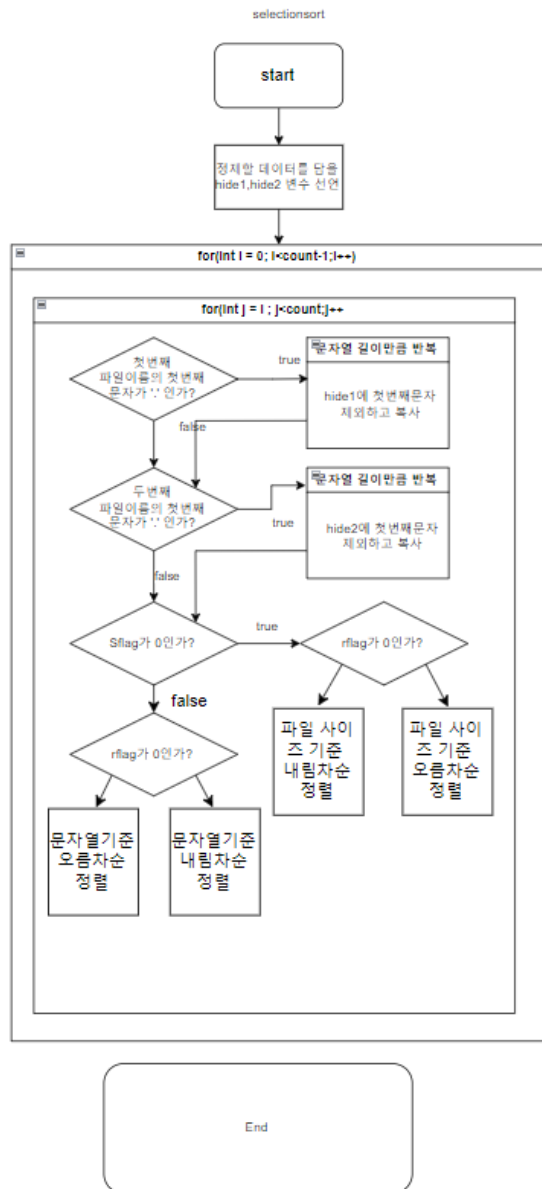
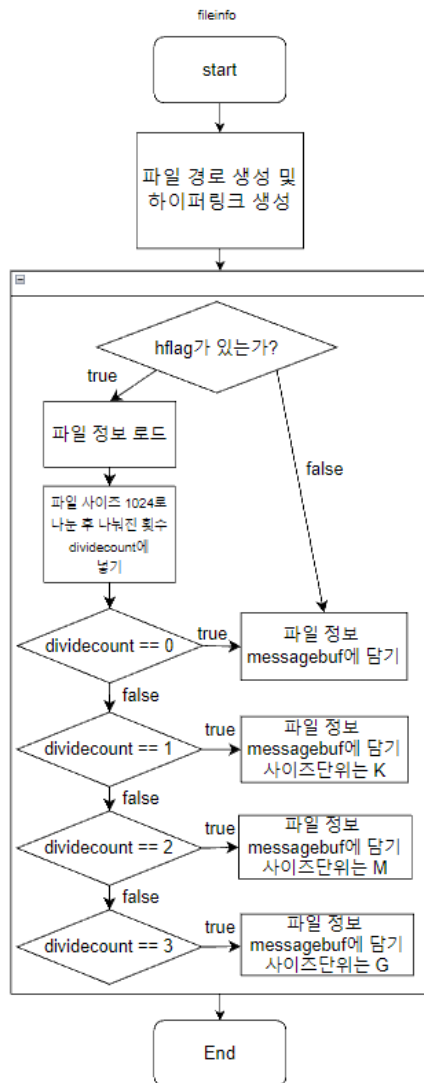
직접 서버를 구축하여 웹 페이지에서 요구하는 정보들을 제공해본다.

서버를 구축하는 과정에서 socket programming에서 사용되는 여러 api들을 익힌다.

하이퍼링크를 사용해보며 url과 하이퍼링크간의 관계를 이해해본다.

2. Flow chart





모든 함수들의 흐름도이다. main 함수는 소켓을 생성하여 소켓을 설정을 마치고 bind,listen을 마무리한 후 while문에서 client의 요청을 기다린다. 그 후 요청이 들어오면 client가 보낸 요청을 해석하고 url을 얻어낸다. 이를 통해 경로를 생성하고 해당경로를 lsfunc함수의 인자로 넣어 준 후 호출한다.

lsfunc 함수는 main 함수에서 넘겨받은 인자에 따라 파일인지 폴더인지를 체크한 후 분기를 나누고 각각 옵션에 따라 요구받은 행동을 취한다.

파일일 경우에는 파일포인터를 선언 후 파일 스트림을 통해 파일을 불러오고 이를 client에게 전송한다.

폴더일 경우에 root directory일 경우에는 l옵션만 적용하여 파일들의 정보를 html형식으로 만들어 보내준다.

그렇지 않을 경우에는 al 옵션은 적용하여 숨김파일에 대한 정보 또한 보낸다.

lsfunc 함수내에서 실행되는 fileaccess함수와 filetype함수는 l옵션에서 제일 먼저 출력되는 파일 타입과 접근권한을 숫자에서 문자데이터로 변환해주는 함수이다.

fileinfo 함수는 lsfunc에서 받은 경로를 통해 하이퍼링크를 생성하고 파일 정보와 함께 버퍼에 담는다.

selectionsort함수는 선택정렬을 구현한 함수로 숨김파일을 구분하지 않고 정렬하기 위해 임시변수를 활용하여 구현하였다.

3. Pseudo Code

```
selectionsort
파일 사이즈 변수선언
파일 경로 만들기 위한 변수선언
문자열 정제 후 저장할 변수 선언

만약 입력 경로가 NULL이 아니라면
    for(파일 개수만큼)
        경로 생성하여 파일 사이즈를 얻어내고 변수에 저장
for(파일개수 -1만큼)
    for(파일개수 -1만큼)
        만약 파일의 앞 문자가 '.' 이라면
            앞문자 제외하여 변수에 저장
        아니라면
            그대로 저장

소문자 모두 대문자로 변경

if(!sflag)
    if(!rflag)
        문자열 기준 오름차순 정렬
    else
        문자열 기준 내림차순 정렬
else
    if(!rflag)
        파일사이즈 기준 내림차순 정렬
        사이즈가 같다면 문자열 기준 정렬
    else
        문자열 기준 오름차순 정렬
        사이즈가 같다면 문자열 기준 정렬
```

```
filetype
입력받은 파일의 mode가 디렉토리라면
    입력받은 문자열의 맨 앞 글자를 d로 변경
입력받은 파일의 mode가 심볼릭 링크라면
    입력받은 문자열의 맨 앞 글자를 l로 변경
|
fileaccess
입력받은 파일의 mode의 user access가 read가 가능하다면
    입력받은 문자열의 2번째를 r로 변경
입력받은 파일의 mode의 user access가 write가 가능하다면
    입력받은 문자열의 3번째를 w로 변경
입력받은 파일의 mode의 user access가 execution이 가능하다면
    입력받은 문자열의 4번째를 x로 변경

입력받은 파일의 mode의 group access가 read가 가능하다면
    입력받은 문자열의 5번째를 r로 변경
입력받은 파일의 mode의 group access가 write가 가능하다면
    입력받은 문자열의 6번째를 w로 변경
입력받은 파일의 mode의 group access가 execution이 가능하다면
    입력받은 문자열의 7번째를 x로 변경

입력받은 파일의 mode의 other access가 read가 가능하다면
    입력받은 문자열의 8번째를 r로 변경
입력받은 파일의 mode의 other access가 write가 가능하다면
    입력받은 문자열의 9번째를 w로 변경
입력받은 파일의 mode의 other access가 execution이 가능하다면
    입력받은 문자열의 10번째를 x로 변경
```

```
lsfunc
입력받은 경로로 디렉토리 열기
디렉토리가 null이라면
    파일 스트림 열고 파일 크기 저장
    if(이미지파일)
        이미지파일 전용 헤더 생성
    elseif(html파일)
        html파일 전용 헤더 생성
    else
        일반파일 전용 헤더 생성
    헤더 파일 write
    파일 바이트단위로 읽고 client에게 보내기
    종료

if(!aflag)
    숨김파일 제외 파일 저장

else
    모두 읽기

if(!lflag)
    정렬 하고 파일들 이름만 출력
else
    정렬하고 파일들 정보를 담은 테이블 생성
    fileinfo 함수를 파일 개수만큼 호출
    헤더파일 write하고 정제된 파일 정보 html형식으로 client에게 보내기
    종료
```

```
fileinfo
구조체 선언
입력받은 경로로 파일 정보 얻어와 구조체에 담기
하이퍼링크 생성을 위해 입력받은 경로 정제
파일 정보를 문자열로 변환
if(hflag)
    파일 정보중 파일 사이즈 부분을 나누어 단위를 붙여줌
파일 정보 messagebuf에 담기

wildcardexistence
wildcard = "*"?"[]"
if(와일드 카드를 입력받은 문자열에서 찾으면)
    return 1
못찾으면
    return 0
```

```
main
소켓 생성
소켓 설정
bind
listen
while(1)
    client accept
    method와 url 토큰으로 얻기
    현재디렉토리를 루트 디렉토리로 간주하고 url이어붙여 경로 생성
    해당 경로에 파일이 존재하지 않는다면 에러전용 헤더와 메시지를 보내주기
    존재한다면 lsfunc함수 호출 후 client disconnect
소켓 닫기
종료
```

4. 결과화면

```
kw2019202050@ubuntu: ~/Desktop/work/Web1_2_D_2019202050
kw2019202050@ubuntu:~/Desktop/work/Web1_2_D_2019202050$ ls
2019202050_web_server.c  good.jpg
Abc.txt                  _good.txt
bcago                    good.txt
bca.html                 KakaoTalk_20230416_224732097.png
dir1                     Makefile
dir2                     text1.txt
dir3                     text4.txt
dir4                     text9.txt
empty.txt                Web2_1_D_2019202050.pdf
final.txt                zoom
god                       zoom.txt
kw2019202050@ubuntu:~/Desktop/work/Web1_2_D_2019202050$ make
gcc -o web_server 2019202050_web_server.c
kw2019202050@ubuntu:~/Desktop/work/Web1_2_D_2019202050$ ./web_server
```

실행예제이다. 실행파일은 web_server로 실행 시 클라이언트를 기다리는 모습이다.

Mozilla Firefox window showing the directory listing for `/home/kw2019202050/Desktop/work/Web1_2_D_2019202050`. The page title is "Welcome To System Programming HTTP".

Directory path: /home/kw2019202050/Desktop/work/Web1_2_D_2019202050
Total: 1240

Name	Permission	Link	Owner	Group	Size	Last Modified
2019202050_web_server.c	-rwxrwx-rw-	1	kw2019202050	kw2019202050	32072	May 1 21:57
Abc.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Mar 31 2:47
bca.html	-rw-rw-r--	1	kw2019202050	kw2019202050	97	May 1 9:10
bcago	lrwxrwxrwx	1	kw2019202050	kw2019202050	8	May 1 9:15
dir1	drwxrwxr-x	3	kw2019202050	kw2019202050	4096	May 1 6:1
dir2	drwxrwxr-x	2	kw2019202050	kw2019202050	4096	Apr 16 4:44
dir3	drwxrwxr-x	2	kw2019202050	kw2019202050	4096	Apr 16 4:44
dir4	drwxrwxr-x	2	kw2019202050	kw2019202050	4096	Apr 16 4:44
empty.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Mar 31 2:50
final.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Mar 31 2:50
god	lrwxrwxrwx	1	kw2019202050	kw2019202050	8	Apr 16 4:24
good.jpg	-rwxrwx-rw-	1	kw2019202050	kw2019202050	28280	May 1 20:37
good.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	13	May 1 9:9
KakaoTalk_20230416_224732097.png	-rwxrwx-rw-	1	kw2019202050	kw2019202050	122087	May 1 9:7
Makefile	-rw-rw-r--	1	kw2019202050	kw2019202050	79	May 2 0:16
text1.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Apr 8 4:54
text4.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Apr 8 4:54
text9.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Apr 8 4:54
Web2_1_D_2019202050.pdf	-rwxrwx-rw-	1	kw2019202050	kw2019202050	1017886	Apr 16 7:4
web_server	-rwxrwxr-x	1	kw2019202050	kw2019202050	27480	May 2 0:16
zoom	lrwxrwxrwx	1	kw2019202050	kw2019202050	8	Apr 16 4:50
zoom.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Mar 31 2:50
_good.txt	-rw-rw-r--	1	kw2019202050	kw2019202050	0	Apr 8 5:18

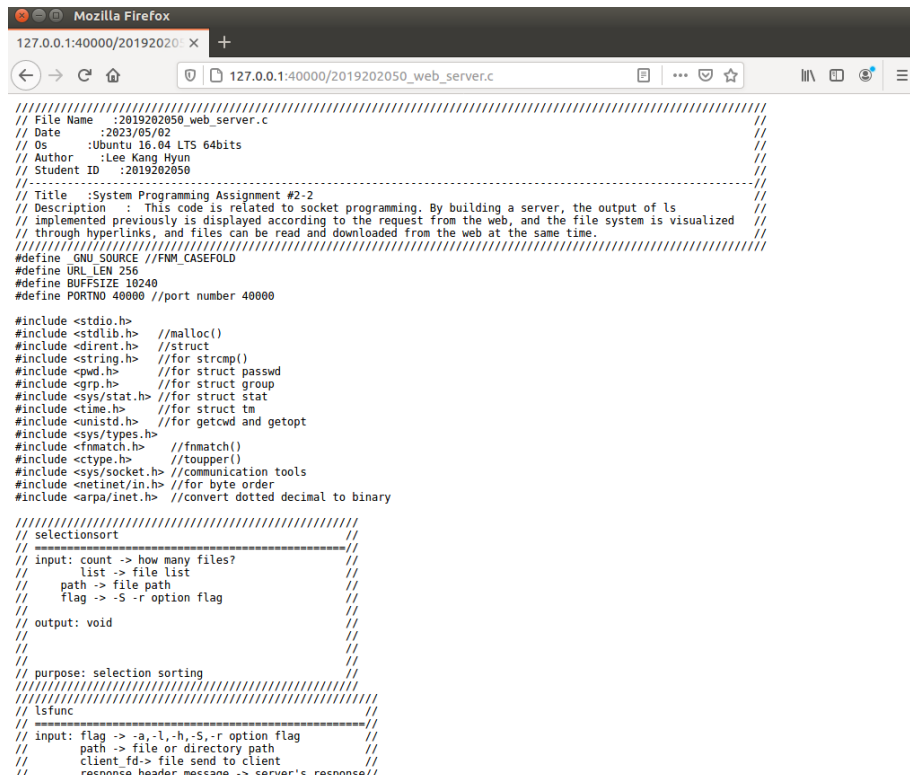
루트 디렉토리의 모습이다. 상위 디렉토리로 접근하지 못하도록 -i 옵션만 적용된 상태이다.

Mozilla Firefox window showing the directory listing for `/home/kw2019202050/Desktop/work/Web1_2_D_2019202050/dir1`. The page title is "System Programming HTTP".

Directory path: /home/kw2019202050/Desktop/work/Web1_2_D_2019202050/dir1
Total: 12

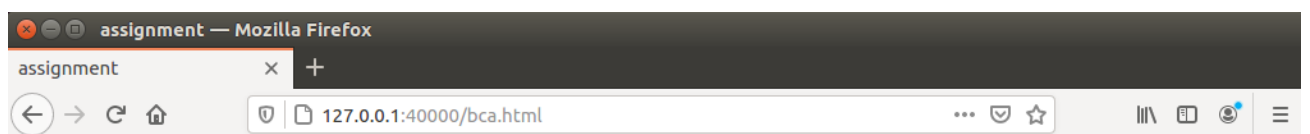
Name	Permission	Link	Owner	Group	Size	Last Modified
..	drwxrwxr-x	3	kw2019202050	kw2019202050	4096	May 1 6:1
..	drwxrwxr-x	6	kw2019202050	kw2019202050	4096	May 2 0:16
hi	drwxrwxr-x	3	kw2019202050	kw2019202050	4096	May 1 6:1

하위 폴더로 하이퍼링크를 통해 들어간 모습이다. 하위 디렉토리이므로 상위 디렉토리로의 접근이 가능하게 `cd -` 옵션이 적용되었다.



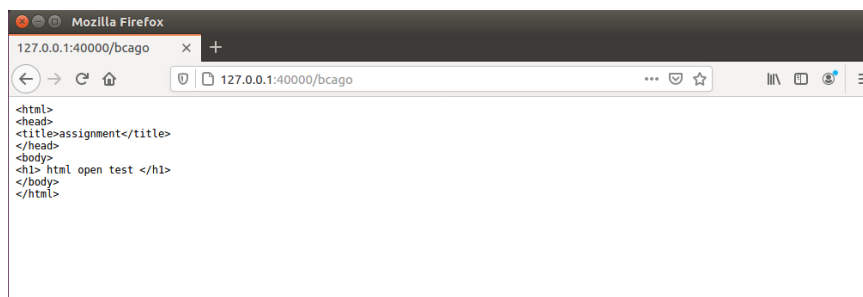
```
#####  
// File Name :2019202050_web_server.c  
// Date :2023/05/02  
// Os :Ubuntu 16.04 LTS 64bits  
// Author :Lee Kang Hyun  
// Student ID :2019202050  
//-----  
// Title :System Programming Assignment #2-2  
// Description : This code is related to socket programming. By building a server, the output of ls  
// implemented previously is displayed according to the request from the web, and the file system is visualized  
// through hyperlinks, and files can be read and downloaded from the web at the same time.  
//-----  
#define GNU_SOURCE //FNM_CASEFOLD  
#define URL_LEN 256  
#define BUFSIZE 10240  
#define PORTNO 40000 //port number 40000  
  
#include <stdio.h>  
#include <stdlib.h> //malloc()  
#include <dirent.h> //struct  
#include <string.h> //for strcmp()  
#include <pwd.h> //for struct passwd  
#include <grp.h> //for struct group  
#include <sys/stat.h> //for struct stat  
#include <time.h> //for struct tm  
#include <unistd.h> //for getcwd and getopt  
#include <sys/types.h>  
#include <fnmatch.h> //fnmatch()  
#include <ctype.h> //toupper()  
#include <sys/socket.h> //communication tools  
#include <netinet/in.h> //for byte order  
#include <arpa/inet.h> //convert dotted decimal to binary  
  
#####  
// selectionsort  
//-----  
// input: count -> how many files?  
// list -> file list  
// path -> file path  
// flag -> -S -r option flag  
//-----  
// output: void  
//-----  
// purpose: selection sorting  
//-----  
// lsfunc  
//-----  
// input: flag -> -a,-l,-h,-S,-r option flag  
// path -> file or directory path  
// client_fd -> file send to client  
// response_header_message -> server's response//
```

source code를 하이퍼링크를 통해 열어본 모습이다. 서버에서 보낸 파일의 정보를 읽을 수 있다.

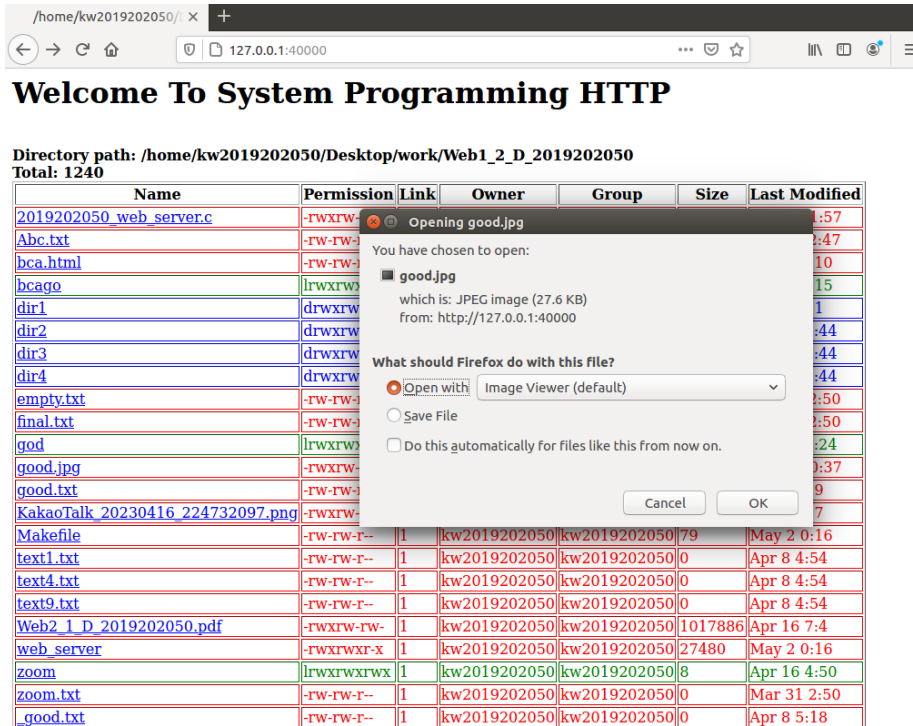


html open test

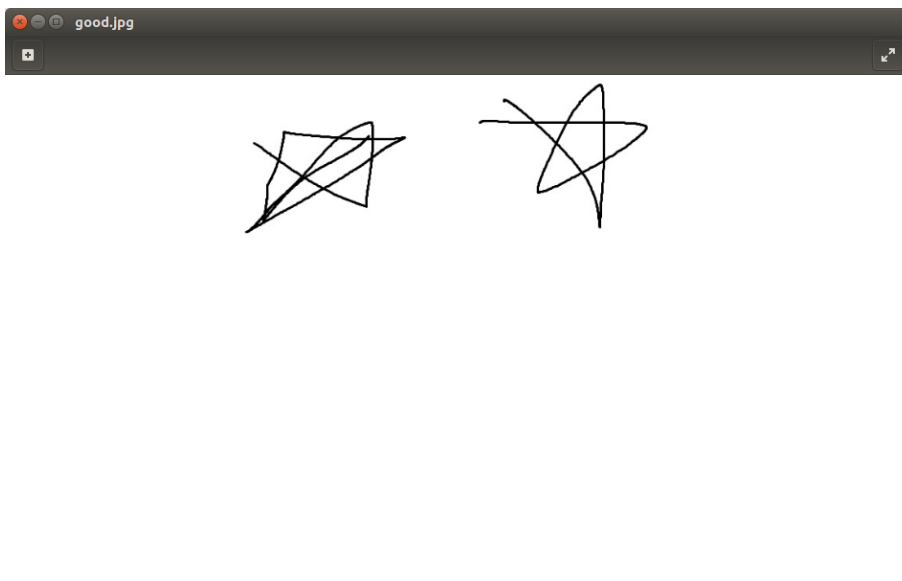
이번엔 html파일을 열어본 모습이다.



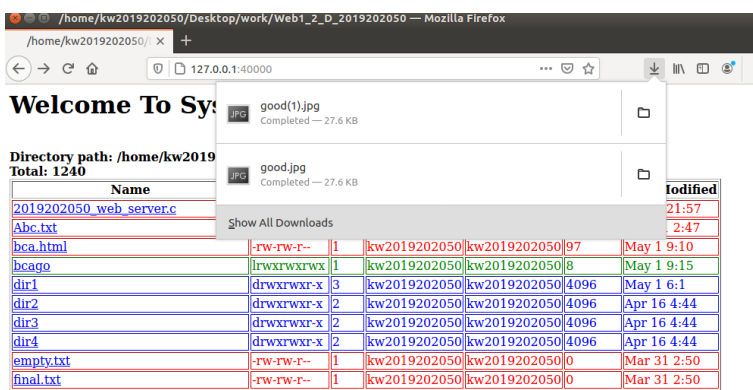
위의 html파일에 심볼릭 링크로 연결된 파일을 열었을 때 파일 자체에 적힌 코드를 확인할 수 있다.



이미지 파일을 클릭했을 때 확인할 수 있는 결과이다. open with를 클릭하면



위와 같이 사진을 볼 수 있고



save를 하게 되면 파일이 다운된 것을 알 수 있다.

Welcome To System Programming HTTP

Directory path: /home/kw2019202050/Desktop/work/Web1_2_D_2019202050
Total: 1240

Name	Permission	Link	Owner	Group	Size	Last Modified
2019202050_web_server.c	-rwxrwx					3:57
Abc.txt	-rw-rw-					3:47
bca.html	-rw-rw-					10
bcago	lrwxrwx					15
dir1	drwxrw					1
dir2	drwxrw					44
dir3	drwxrw					44
dir4	drwxrw					44
empty.txt	-rw-rw-					Mar 31 2:50
final.txt	-rw-rw-					Mar 31 2:50
god	lrwxrwxrwx					Apr 16 4:24
good.jpg	-rwxrw-rw-				28280	May 1 20:37
good.txt	-rw-rw-r--				13	May 1 9:9
KakaoTalk_20230416_224732097.png	-rwxrw-rw-				122087	May 1 9:7
Makefile	-rw-rw-r--				79	May 2 0:16
text1.txt	-rw-rw-r--				0	Apr 8 4:54
text4.txt	-rw-rw-r--				0	Apr 8 4:54

실행파일 또한 파일을 다운할 수 있다.

5. 고찰

이번 과제는 socket programming에 관한 내용이었다. socket이라는 것을 이용하여 file descriptor처럼 사용해서 외부와 통신하는 법을 알 수 있었고 통신을 하기 위해서는 형식과 통신 규약을 지켜야 했는데 이를 통해 protocol에 대한 이해를 할 수 있게 되었다. 또한 socket을 설정하면서 실습시간때 배운 코드를 충분히 이해하고 이를 활용하는 과정에서 각각의 api가 어떻게 작동하는지에 대해 배웠다. 과제를 해결해 나가면서 어려웠던 점은 url을 통해 하이퍼링크를 만드는 과정이 제일 어려웠는데 그중에서도 특히 .과 ..의 하이퍼링크를 만드는 것이 어려웠다. /를 문자열에서 처리하고 href에 .과 ..을 주었을 때 어떻게 작동하는지 서버에서 파일을 열기위해 사용하는 경로와 실제 하이퍼링크에 달아야하는 경로의 차이를 확실히 알아야 했고 문자열을 어떻게 정제해야하는지를 생각해야했다. 또한 message를 먼저적고 header를 나중에 적는 경우에 웹페이지가 구성되지 않았는데 header의 역할과 message의 역할을 알고 통신에서의 response 형식을 지키는 법을 이번 실습을 통해 알게된 것 같다.

6. Reference

시스템프로그래밍실습 강의자료 참조