

Computer Network Project 1

2014037901 나윤환

1. High-level description

```
char *token = strtok(buffer, " ");
char *raw_path = strtok(NULL, " ");
char *path;
printf("\n\n** raw_path: %s\n", raw_path);
if (strcmp(raw_path, "/") == 0)
{
    path = (char *)malloc(strlen("./index.html") + 1);
    strcpy(path, "./index.html");
}
else
{
    path = (char *)malloc(strlen(raw_path) + 1);
    sprintf(path, "%s", raw_path);
}
char *content_type = getContentType(path);
```

```
char *getContentType(char *path_token)
{
    char *exts[] = {".html", ".jpg", ".jpeg", ".png", ".bmp", ".gif", ".mp3", ".pdf", ".ico"};
    char *content_types[] = {"text/html",
                             "image/jpeg",
                             "image/jpeg",
                             "image/png",
                             "image/bmp",
                             "image/gif",
                             "audio/mp3",
                             "application/pdf",
                             "image/x-icon"};
    char *result = content_types[0];
    int len_exts = (int)(sizeof(exts) / sizeof(exts[0]));

    for (int i = 0; i < len_exts; i++)
    {
        if (strstr(path_token, exts[i]) != NULL)
        {
            result = (char *)malloc(strlen(content_types[i]) + 1);
            strcpy(result, content_types[i]);
            break;
        }
    }

    printf("content-type: %s\n", result);
    return result;
}
```

getContentType function 은 HTTP Request 에서 파일 경로에 대한 부분을 parsing 하고 이 parsing 된 path 를 바탕으로 ContentType String 을 return 하는 함수입니다. 매치하는 파일 확장자가 없다면 404.html 을 표시하기 위해서 "text/html"을 return 합니다.

```

    content_type = get_content_type(path);
    int file = open(path, O_RDONLY);
    if (file < 0)
    {
        file = open("./404.html", O_RDONLY);
        content_type = (char *)malloc(strlen("text/html") + 1);
        strcpy(content_type, "text/html");
    }

```

Parsing 된 path 를 기반으로 file 을 open 하는데, 만약에 file 을 open 하지 못한다면 (파일 이름이 잘못 되었거나 지원하지 않는 파일 형식일 경우) 404.html 파일을 표시합니다.

2. Difficulties

2-1. 파일이 없는 경우

Before

```

int file = open(path, O_RDONLY);
if (file < 0) error("Failed to open files");

```

After

```

    content_type = get_content_type(path);
    int file = open(path, O_RDONLY);
    if (file < 0)
    {
        file = open("./404.html", O_RDONLY);
        content_type = (char *)malloc(strlen("text/html") + 1);
        strcpy(content_type, "text/html");
    }

```

원래는 파일을 찾을 수 없는 경우(file < 0)에는 error function 을 실행함으로써 서버가 종료되도록 처리했었습니다. 서버가 종료되지 않도록 하기 위해서 파일을 찾지 못한 경우에는 무조건 404.html 을 return 하도록 프로그램을 개발하였습니다.

2-2. 파일이 Buffer 보다 큰 경우

```

// Write File during sending whole file bytes
while ((file_size = read(file, buffer, BUFF_SIZE)) > 0)
{
    write(client_fd, buffer, BUFF_SIZE);
}

```

파일이 BufferSize 보다 큰 경우에는 while 문을 사용하여 계속해서 Write 할 수 있도록 개발하였습니다.

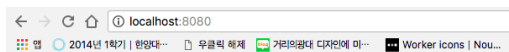
2-2. MP3 파일이 재생이 안되는 오류

```
int pid = fork();
int status;

if (pid < 0)
    error("Failed to fork");
if (pid > 0)
    wait(&status);
```

다른 포맷의 파일은 문제가 없었지만 MP3 파일의 경우에는 화면만 뜬 채 음악이 재생이 안되는 현상이 발생하였습니다. 이 문제를 자식 프로세스를 생성하는 fork() 함수를 사용하여 해결하였습니다.

3. Sample Outputs

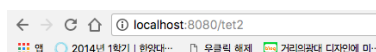


NetWork Project1 - Socket Programming

2014037901 Computer Science 나윤환

URL 뒤에 파일 이름을 입력해주세요

* Index.html

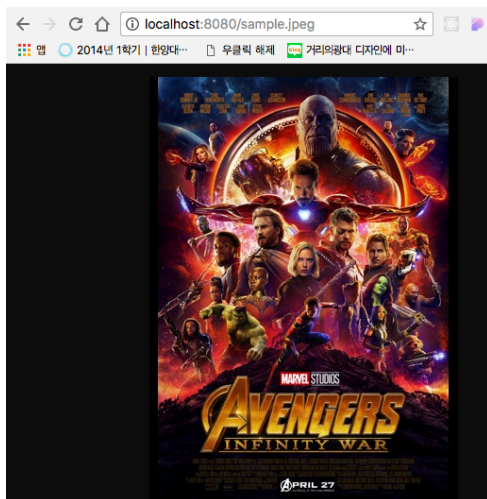


No File

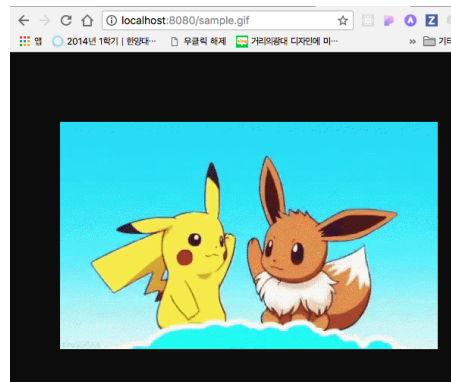
2014037901 Computer Science 나윤환

서버에 해당하는 파일을 찾지 못하였습니다.

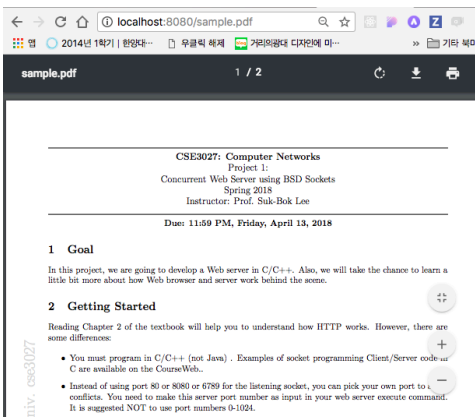
* 404.html



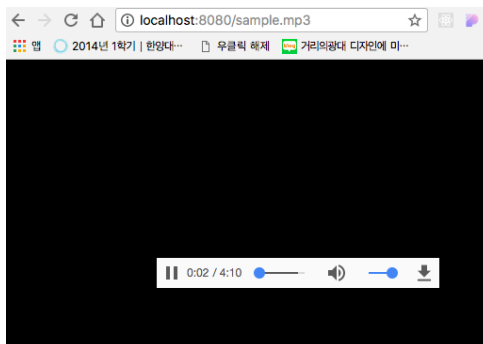
* Sample JPEG



* Sample GIF



* Sample PDF



* Sample MP3