

## [실습 5] 파일 조작

파일(file)은 일반 파일과 특수 파일로 구분할 수 있다. 일반 파일은 텍스트 파일과 이진 파일로 구성된다. 특수 파일은 디렉터리, 장치 파일 (device file), 링크 파일, 네트워크 파일, 파이프 등을 말한다. 이번 실습에서는 파일을 복사하거나 파일 이름 변경 혹은 이동하는 명령어를 사용해 보고, 파일의 종류를 확인하는 명령어를 알아본다.

※ 아래 명령어를 이용하여 문제에 답하시오.(1~6)

**cd, cp, mv, ls, tree, file**

☞ firefox로 github 사이트(<https://github.com/wkhong89/usys-class>)에 접속하여 코드를 다운로드 받으시오.

☞ 다운로드 받은 zip 파일을 같은 디렉토리에 압축해제 하시오.

1) 압축 해제된 파일이 있는 위치로 이동 후, tree 명령어를 사용하여 해당 디렉토리의 구조를 보여주시오.

2) 압축 해제된 파일의 디렉토리 chap04를 로그인 계정의 ~/laPiscine/ 디렉토리 아래에 복사한 후 laPiscine 디렉토리 구조를 보여주시오. (tree 명령어 사용)

3) laPiscine 디렉토리 아래에 복사된 chap04의 디렉터리 이름을 day3로 변경하고, 제대로 변경되었는지 결과를 확인하시오.

4) day3 디렉토리 아래에 있는 파일 중 텍스트 파일과 이진 파일을 구분하시오.

5) day3 디렉토리 아래에 txt\_files와 c\_files라는 디렉터리를 생성하고, day3 디렉터리에 있는 확장자 txt인 파일은 모두 txt\_files 디렉터리로, 확장자가 c인 파일은 모두 c\_files로 옮긴 후, 명령이 제대로 수행되었음을 보여주시오.

6) 장치 파일(device file)은 문자 (character) 장치 파일과 블록 (block) 장치 파일로 구성된다. 리눅스 시스템에서 제공하고 있는 문자 장치 파일과 블록 장치 파일을 찾아서 해당 파일이 장치 파일임을 보여주시오.

## [실습 6] 파일 접근 권한과 소유

파일의 읽기, 쓰기, 실행 권한은 소유자(user), 소유자와 동일 그룹(group)에 속한 사용자, 기타(others) 사용자로 구분하여 다르게 부여할 수 있다. 본 실습에서는 파일의 접근 권한을 해석하고, 접근 권한을 변경하는 명령어를 실습한다. 또한, 파일에 대한 소유자와 그룹을 변경해 보고, 소유자와 그룹이 변경되었을 경우 어떤 영향을 미치는지 실습을 통해 확인한다.

※ 아래 명령어를 이용하여 문제에 답하시오 (1~7). 실습 5에서 생성한 디렉터리 ~/laPiscine/day3와 파일이 있어야 한다.

**ls, chmod, chown, chgrp, addgrp, adduser, id, usermod**

1) 파일 hello가 실행 가능한 파일인지 확인하시오. 실행이 되는가? 그렇지 않다면 그 이유를 설명하고, 실행이 가능하도록 파일 속성을 변경하고, 실행 결과를 보이시오.

☞ hello를 실행하기 위해서는 상대 경로명을 사용해야 한다.

2) day3/file1의 소유자를 guest 계정으로 변경하시오. 현재 로그인 계정이 guest가 아닐 때, day3/file1에 파일 쓰기를 시도하면, 운영체제가 어떤 메시지를 보여주는가? 그 이유를 설명하시오.

☞ file1의 접근권한이 644 (rw-r--r--)여야 함.

☞ 단, 시스템에 guest 계정이 있어야 하므로, 없을 경우 계정을 먼저 생성하시오.

3) guest 계정으로 전환(switch) 하시오. ubuntu 계정의 홈 디렉토리로 이동해보자. 이동이 되는가? 이동이 되지 않는다면 그 이유를 설명하시오.

□ 다음 실습 과정(4-6)을 통해 guest 계정에서 ubuntu 홈 디렉토리로 이동할 수 있다.

4) ubuntu 계정과 guest 계정을 동일한 그룹 laPiscine에 두고자 한다. ubuntu 계정으로 전환(switch)하여 laPiscine 그룹을 만들고, laPiscine 그룹이 제대로 생성되었음을 보이시오.

5) ubuntu 계정과 guest 계정을 laPiscine 그룹에 추가하고, id 명령어를 이용하여 현재 로그인 계정과 guest 계정의 기본 그룹(primary group)과 기타 그룹들을 확인하시오.

6) ubuntu 홈 디렉토리를 laPiscine 그룹이 접근할 수 있도록 접근 권한

(permission)을 변경하시오. guest 계정으로 전환 후, ubuntu 홈 디렉토리로 이동해보시오. 이동이 되는가? ~/laPiscine/day3/file1에 'added by guest'를 추가 시도하시오. 추가되는가?

7) 나의 계정을 ubuntu로 전환(switch)하시오. nano를 이용해 'created by ubuntu'를 작성한 후 ~/laPiscine/day3/file2로 저장하시오. guest 계정으로 이 파일에 쓰기 가능한가? 가능하지 않다면 그 이유를 설명하시오. ~/laPiscine/day3/file2에 쓰기 권한을 부여하도록 하기 위해 파일 속성을 어떻게 변경하는 것이 좋을까? ubuntu와 guest가 같은 그룹에 있음을 활용하시오. file2의 접근 권한을 변경한 후 다음 명령어를 입력해 보시오.

```
$ echo "added by $USER" >> file2
```

8) 나의 계정 (guest 계정이 아닌 계정)의 기본 그룹(primary group)을 laPiscine로 변경하시오. (Hint: usermod를 이용하시오.) id 명령어를 이용해 기본 그룹이 어떻게 바뀌었는지 확인하시오. 기본 그룹이 변경되었을 때 내 계정 소유의 파일들에 어떤 변화가 있는지 확인하시오.

## [실습 7] inode와 파일 링크

inode는 유닉스의 파일시스템의 기본 자료구조로써, 파일마다 고유의 inode가 할당된다. 파일이 생성되거나 읽기, 쓰기를 포함해 각종 파일에 대한 조작 정보가 inode에 저장되고 관리된다. 하드 링크와 심볼릭 링크의 차이점을 실습을 통해 확인한다.

※ 아래 명령어를 이용하여 문제에 답하시오(1-6). 실습은 ~/laPiscine/day3에서 이루어진다.

**ls, stat, ln, touch**

- 1) touch 명령어를 이용하여 빈 파일 hello.txt를 생성한 후, inode 번호를 확인하시오.
- 2) hello.txt에 대한 하드 링크 파일 hi.txt를 생성한 후, inode 번호를 확인하시오.
- 3) hello.txt와 hi.txt의 하드 링크 수는 몇 개인가? 하드 링크 수의 의미를 설명하시오.
- 4) hi.txt에 임의의 문자열을 추가한 후, hello.txt 파일과 내용이 동일한지 확인하고, 이유를 설명하시오.
- 5) hello.txt를 삭제하고, 하드 링크의 수를 확인하시오. 하드 링크 수가 줄어든 이유를 설명하시오.
- 6) 새로운 디렉토리 link를 생성한 후, link 디렉토리 아래에 hi.txt의 심볼릭 링크를 생성하시오. hi.txt와 link/hi.txt의 파일 속성에 어떤 차이가 있는지 설명하시오.
- 7) hi.txt를 삭제한 후 link/hi.txt 파일을 열 수 있는가? 그 이유를 설명하시오.