

# 객체지향 프로그래밍 2차 프로젝트

과제번호: 02

실습반: 318호

실습실명/실습조교명: 남경진 조교

이름: 안창희

학번: 201723272

학과: 소프트웨어학과

학년: 3학년

- 서론 (본 과제에 대하여 전체적으로 소개한다. 과제가 무엇인지, 어떻게 해결하였는지 접근방법, 구현하지 못한 부분 등을 설명한다.)

본 과제의 구현은 리눅스 터미널에서의 파일시스템 구조와 흡사한 프로그램을 JAVA 언어를 통해서 만들어내는 것이었다.

본인은 이를 구현하기 위해 해당 기능을 가장 잘 표현할 수 있는 API를 검색하여 공부하였고, java.io 내의 File 클래스와 FileWriter 클래스 개념을 통해 기본적인 파일 입출력과 처리에 대해 배웠다.

또한 BufferdReader 와 InputStramReader 그리고 java.util 내의 StringTokenizer를 통해 과제에서 주어진 터미널 형식과 유사한 입력 방식을 최대한 구현해보려 했다.

각각의 개별적인 기능인 new, del, mkdir, chmod, show, chdir, cwd 등의 커맨드의 구현은 완료를 했으나, 이를 복합적으로 실행했을 때 발생하는 오류를 본인은 해결하지 못했다.

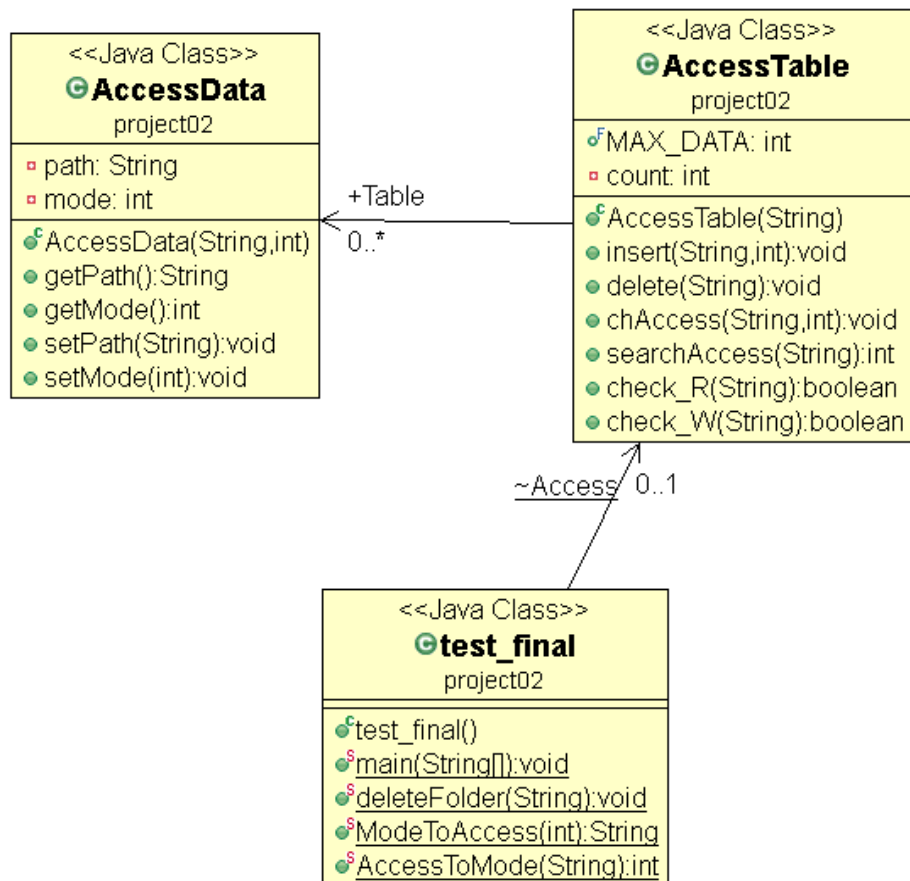
본 과제에서의 가장 큰 어려움은 접근권한 개념의 구현이었는데, 본인은 이를 AccessData를 종합한 AccessTable이라는 개념으로 구현했다. 간략히 말하자면 파일/폴더가 생성될 때마다 파일의 경로와 mode를 짝 지은 객체를 생성해 이 데이터들의 배열인 테이블에 저장하는 방법이다.

개별적인 커맨드의 구현에서는 이 방법이 성공한듯 보였으나, 복합적인 구현인 파일의 생성 및 삭제 후 chdir 커맨드를 통해 폴더를 이동하는 과정에서 이 테이블의 데이터가 원인을 알 수 없게 소실되는 오류가 발생하여 완벽한 기능의 구현은 실패하였다.

또한 설계조건 1인 상속과 다형성을 사용하여 프로그램에 구현하지 못했다.

- 분석/설계 (flowchart, 클래스 다이어그램 등을 이용하여 설명한다.

그림만 그리면 안되고 반드시 설명이 있어야 한다. 구조와 동작 흐름에 대해 설명한다)



프로그램의 클래스 다이어그램은 다음과 같다.

기본적으로 대부분의 작업의 처리가 **test\_final** 클래스의 메인 메소드에서 이루어지며 **AccessTable**에서는 각 파일과 폴더에 대한 접근권한의 입출력과 수정만 이루어지는 데이터베이스와 유사한 구조이다.

**test\_final** 클래스에서는 각 커맨드에 해당하는 입력을 line단위로 받고 커맨드에 따라 입력을 상이한 방식으로 처리하도록 한다.

한 커맨드를 수행한 후 `while`의 무한루프를 통해 다음 line을 기다리도록 했으며 `exit` 커맨드를 입력한 경우에만 루프에서 탈출하게 하였다.

추가로 프로그램 구현중 자가진단을 위한 ls 커맨드를 구현했으며 이는 리눅스에서의 ls를 모방하여 작성했다.

- 실행 결과 화면 (결과에 대한 설명이 있어야 한다. 주의: 반드시 흰색바탕에 검은 글씨로 출력되도록 캡처-토너절약을 위해)

```
Console ✕
test_final [Java Application] C:\wopenjdk-11.0.2_windows-
% new +rw text.txt HelloHell
파일생성 성공. ( 경로 C:\test\ )
% ls
==현재폴더 리스트를 출력합니다.==
text.txt          +rw
% show text.txt
HelloHell
```

## 1. 파일의 생성과 열람

```

test_mat_pava Application: C:\wopenjak\11.0.2_window
% mkdir +rw ttt
폴더가 생성되었습니다.
% ls
==현재폴더 리스트를 출력합니다.==
ttt                +rw                <폴더>
% chdir ttt
% ls
==현재폴더 리스트를 출력합니다.==
문서가 없습니다.
% cwd
현재폴더의 경로는 : C:\test\qtt\  입니다
% new +rw text5.txt aaa
파일생성 성공. ( 경로 C:\test\qtt\ )
% ls
==현재폴더 리스트를 출력합니다.==
text5.txt          +rw
% show text5.txt
|aaa
%

```

## 2. 폴더의 생성과 현재 폴더 이동 후 파일생성 및 열람

+현재 폴더 경로확인

```
Console
test_final [Java Application] C:\wopenjdk-1
% new +rw test.txt 111
파일생성 성공. ( 경로 C:\test\ )
% del test.txt
C:\test\test.txt 파일삭제 완료
% ls
==현재폴더 리스트를 출력합니다.==
문서가 없습니다.
% |
```

### 3. 단순 파일의 삭제

```
% del ttt
C:\test\test\add\bbb\ccc\Report_project01_201723272_안창희.pdf 삭제되었습니다.
C:\test\test\add\bbb\ccc 삭제되었습니다.
C:\test\test\add\bbb\ddd 삭제되었습니다.
C:\test\test\add\bbb 삭제되었습니다.
C:\test\test\add 삭제되었습니다.
C:\test\test 폴더삭제 완료
% ls
==현재폴더 리스트를 출력합니다.==
문서가 없습니다.
%
```

### 4. 다중의 폴더와 파일이 포함된 폴더의 삭제(재귀적으로 구현)

```

% mkdir +rw 11
폴더가 생성되었습니다.
% ls
==현재폴더 리스트를 출력합니다.==
11                +rw                <폴더>
% chdir 11
% cwd
현재폴더의 경로는 : C:\test\11\   입니다
% mkdir +rw 22
폴더가 생성되었습니다.
% chdir 22
% cwd
현재폴더의 경로는 : C:\test\11\22\   입니다
% new +rw test1.txt 33
파일생성 성공. ( 경로 C:\test\11\22\ )
% show test1.txt
33
% chdir ..
% cwd
현재폴더의 경로는 : C:\test\11   입니다
% ls
==현재폴더 리스트를 출력합니다.==
비트에러
22                null                <폴더>
% show 22
%

```

## 5. 여러 개 폴더의 생성 후 부모 폴더로 돌아가 자식 폴더를 show 하는 사례

이렇게 chdir 을 통해 자식으로 이동한 후 다시 부모로 돌아오면 AccessTable 이 소실되는 오류가 발생한다.

- 결론 (이 프로그램에서 배운 점 등과 어려웠던 부분, 미구현 부분에 대해 해결책 등에 대해 기술한다.)

프로그램의 구현 과정에서 File 클래스를 활용하는 법을 배웠고 저번 프로젝트에서도 난항을 겪었으나 사용자로부터 입력을 받을 수 있는 조건과 방법이 기존에 알던 방식보다 훨씬 다양하다는 것을 배웠다.

하지만 BufferdReader나 StringTokenizer를 활용한 입력의 구현에도 일반 파일 생성시 access지정이 없으면 default로 하든지 입력 형식이 잘못 입력되어도 커맨드가 발동하는 등의 에러는 모두 처리하지 못했다.

이 부분에 대해서는 각 커맨드를 while문 내에서 if와 else문 내부를 코딩하는 것이 아니라 각각의 경우에 해당하는 메소드를 세부적으로 정의하고 발동조건을 확립함으로써 해결할 수 있다고 생각한다.

실제로 del 커맨드의 구현에서 초기에는 메인 메소드 내의 코딩으로 모든 것을 구현하려 했지만 이후 deleteFolder 메소드를 추가하여 재귀적으로 구현하여 완성도를 높였다.

또한 상속과 다형성의 구현에서도 user 클래스를 부모로 가지는 superuser 클래스를 정의한 다음 각 커맨드가 실행되는 부분을 메소드화 한 후 실행할 당시 검출하는 접근권한을 superuser에서는 작동하지 않도록 user의 커맨드관련 메소드를 override하는 방식으로 구현하면 충분히 구현 가능할 것이라 설계했으나 시간관계상 구현하지 못하였다.

그리고 최종적으로 폴더 이동상의 AccessTable 소실 문제는 AccessTable을 전역변수로 지정하여 다시금 실험해 볼 생각이지만 근본적인 오류 원인을 알 수 없어 인터넷 검색을 통해 피드백을 받으려고 한다.