

포스코 ai-big data 아카데미 21기

## 성적관리 프로그램

분반: B분반

조: 4조

이름 권혁준

이메일 : [gurwns876@naver.com](mailto:gurwns876@naver.com)

명예서약(Honor code)

“나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.”

## Problem: 성적관리 프로그램

### 1. 문제의 개요

본 프로그램을 간략히 설명하면 다음과 같다 .

- 프로그램 실행 시 텍스트 파일로부터 학생들의 성적 목록 작성을 위한 데이터를 읽는다.
- 명시된 7가지 명령어 외의 명령어가 입력될 경우 무시하고 다시 명령어 입력을 대기한다.
- (전체 학생 정보 출력), search (특정 학생 검색), changescore (점수 수정), add (학생 추가), searchgrade (Grade 검색), remove (특정 학생 삭제), quit (종료) 의 7가지 기능이 있다.
- 명시되어 있지 않은 세부사항에 대해서 처리한 방법, 이유 등을 추가 한다.

### 2. 알고리즘, 프로그램 구조 및 설명

본 프로그램 작성을 위한 설계는 다음과 같다.

#### 1. 텍스트 파일 입력받기

- 1-1) 다른 파일일 시 그 파일을 읽는다.
- 1-2) 없을 경우 default값(기본값)으로 students.txt을 읽는다.

#### 2. 입력 받은 데이터를 예시의 내용 및 구성으로 변환하기

- 2-1) 입력받은 학번, 이름, 중간점수, 기말점수를 [Student number] [Wt] [Name] [Wt] [Midterm] [Wt] [Final] [Wn]로 구성되어 하기 위해 탭[Wt]으로 각각의 항목을 구분하고 한 줄이 끝날 마다 줄바꿈[Wn]을 추가시킨다.
- 2-2) 평균과 학점을 추가하기 위해 중간점수와 기말점수에 해당하는 값을 계산하여 구한다.
- 2-3) 평균을 기준으로 오름차순 정렬한다.
- 2-4) 최종적으로 학번, 이름, 중간점수, 기말점수, 평균, 학점을 딕셔너리에 학번을 키워드로 사용하여 학생목록을 만든다.

#### 3. 커맨드 입력받기

- 3-1) while문 무한루프로 명령어를 입력받는다.

#### 4. 기능 구현을 위한 함수 생성

##### 4-1) show 구현

4-1-1) 생성한 딕셔너리를 내림차순으로 key, value로 나누어 출력한다.

##### 4-2) search 구현

4-2-1) 학번이 있으면 학생목록메뉴와 학생목록을 출력한다.

4-2-2) 학번이 없으면 NO SUCH PERSON. 메시지를 출력한다.

##### 4-3) changescore 구현

4-3-1) 학번이 있다면 mid, final 둘 중 하나의 텍스트 값을 입력받는다.

4-3-2) 그 후, 해당 값을 수정하고 그에 맞게 평균, 학점도 수정한다.

4-3-3) 학번이 없다면 NO SUCH PERSON. 메시지 출력, mid, final 텍스트 값이 아니라면 실행되지 않는다.

##### 4-4) add 구현

4-4-1) 학번이 없으면 이름, 중간, 기말을 입력받고, 평균, 학점을 계산하여 학생목록에 추가한다. 그 후 평균에 맞게 내림차순 정렬하고 Student added. 메시지를 출력한다.

4-4-2) 학번이 있다면 ALREADY EXISTS 메시지를 출력한다.

##### 4-5) searchgrade 구현

4-5-1) 학점을 입력 받고 학생목록에 입력받은 학점이 학생 목록에 있는지 검사한다. 있다면 학생목록을 출력한다.

4-5-2) 학점이 A,B,C,D,F가 아니라면 실행을 하지 않고, 맞음에도 불구하고 학생 목록에 학점에 맞는 학생이 없다면 NO RESULTS.를 출력한다.

##### 4-6) remove 구현

4-6-1) 학생목록이 존재한다면 학번을 입력받는다. 없다면 List is empty. 메시지를 출력한다.

4-6-2) 학번이 있다면 해당 학생목록을 삭제한다.

4-6-3) 학번이 없다면 NO SUCH PERSON. 메시지 출력한다.

##### 4-7) quit 구현

4-7-1) 파일을 저장시킬지 여부를 확인한다.

4-7-2) 저장시킨다면 파일명을 입력받고, 학생목록의 각 항목 사이에 wt을 추가하고 목록 사이에는 wn을 추가한다.

4-7-3) 저장시키지 않는다면 실행을 하지 않는다.

4-8) menu함수 구현

4-8-1) 학번, 이름,중간,기말,평균,학점 메뉴를 출력시키는 함수를 만든다.

4-9) 평균,학점 업그레이드 함수 구현

4-9-1) 학생목록이 있는 딕셔너리와 학번을 입력 받는다.

4-9-2) 해당 학번의 평균과 학점을 중간, 기말점수를 계산하여 구한다.

### 3. 프로그램 실행방법 및 예제

```
# show
Student      Name      Midterm  Final  Average Grade
-----
20180002     Lee Jieuna    92      89     90.5      A
20180009     Lee Yeonghee  81      84     82.5      B
20180001     Hong Gildong  84      73     78.5      C
20180011     Ha Donghun   58      68     63.0      D
20180007     Kim Cheolsu  57      62     59.5      F
```

show 입력시 내림차순으로 학생목록을 출력한다.

```
# search
Student ID: 20180007
Student      Name      Midterm  Final  Average Grade
-----
20180007 Kim Cheolsu  57  62  59.5  F
```

#

Search일 때, 학번이 있다면 해당 학생목록을 출력한다.

```
# changescore
Student ID: 20180002
mid/final? mid
Input new score: 90
Student      Name      Midterm  Final  Average Grade
-----
20180002 Lee Jieuna  92 89 90.5 A
Score changed.
20180002 Lee Jieuna  90 89 89.5 B

# 
```

changescore일 때, 학번과 mid,final 값중에 입력이 들어올 때 해당 점수를 수정하고 평균,학점을 수정한다.

```
# add
Student ID: 20181111
Name: jung yongjun
Midterm Score: 90
Final Score: 80
Student added.
# show
Student      Name      Midterm  Final  Average Grade
-----
20180002      Lee Jieuna    90      89      89.5      B
20180009      Lee Yeonghee  81      84      82.5      B
20180001      Hong Gildong  84      73      78.5      C
20180011      Ha Donghun    58      68      63.0      D
20180007      Kim Cheolsu   57      62      59.5      F
20181111      jung yongjun  90      80      85.0      B
# add
Student ID: 20181111
ALREADY EXISTS.
```

add일 때, 학생목록에 학번이 없다면 이름, 중간점수, 기말점수를 입력받고 평균, 학점을 계산하여 학생목록에 추가한다. 학번이 있다면 존재한다는 메시지를 출력한다.

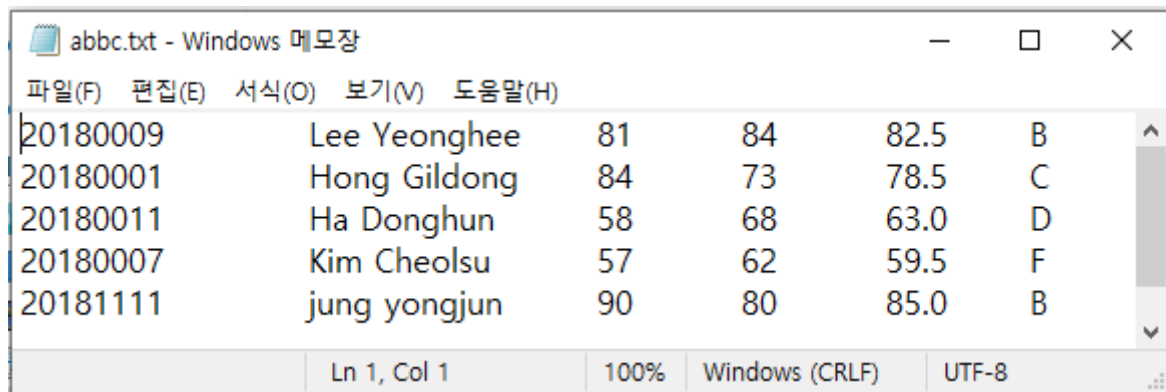
```
# show
Student      Name      Midterm  Final  Average Grade
-----
20180002     Lee Jieuna    90      89     89.5      B
20180009     Lee Yeonghee  81      84     82.5      B
20180001     Hong Gildong  84      73     78.5      C
20180011     Ha Donghun   58      68     63.0      D
20180007     Kim Cheolsu  57      62     59.5      F
20181111     jung yongjun  90      80     85.0      B
# searchgrade
Grade to search: A
NO RESULTS
# searchgrade
Grade to search: B
Student      Name      Midterm  Final  Average Grade
-----
20180002 Lee Jieuna  90 89 89.5 B
20180009 Lee Yeonghee 81 84 82.5 B
20181111 jung yongjun 90 80 85.0 B
# 
```

seachgrade일 때, 학점을 입력하면 그에 맞는 학생목록이 출력된다. A,,B,C,D,F가 아닌 E를 입력받으면 실행을 하지 않는다.

```
# searchgrade
Grade to search: B
Student      Name      Midterm  Final  Average Grade
-----
20180002 Lee Jieuna  90 89 89.5 B
20180009 Lee Yeonghee 81 84 82.5 B
20181111 jung yongjun 90 80 85.0 B
# remove
Student ID: 20180002
Student removed.
# show
Student      Name      Midterm  Final  Average Grade
-----
20180009     Lee Yeonghee  81      84     82.5      B
20180001     Hong Gildong  84      73     78.5      C
20180011     Ha Donghun   58      68     63.0      D
20180007     Kim Cheolsu  57      62     59.5      F
20181111     jung yongjun  90      80     85.0      B
```

remove일 때, 해당 학번이 있다면 그에 해당하는 학생목록을 삭제한다.

```
# show
Student      Name      Midterm  Final  Average Grade
-----
20180009     Lee Yeonghee    81      84      82.5    B
20180001     Hong Gildong    84      73      78.5    C
20180011     Ha Donghun      58      68      63.0    D
20180007     Kim Cheolsu     57      62      59.5    F
20181111     jung yongjun    90      80      85.0    B
# quit
Save data?[yes/no]yes
File name: abbc.txt
```



quit일 때, yes로 파일을 저장시키고 파일명을 입력하면 학생목록을 해당 파일명으로 저장한다.

#### 4. 토론

- seachgrade를 할 때 학점이 있더라도 학생목록이 없을 경우 no result문으로 출력시켜야 하는데, 이때 학점을 입력받고나서 학생목록을 먼저 검사를 해야 된다는 사실을 알게 되었다.
- 학생목록메뉴와 학생목록을 항목마다 출력시킬 때 print문에다가 sep='wt'를 사용해서 세로로 일정한 간격이 되게끔 출력하고 싶었다. 그러나 탭문자는 파이썬 내부적으로 문자열을 포함한 일정공간으로 출력되게끔 설정되어 있어서 깔끔하게 메뉴마다 학생목록이 출력되도록 만들지 못하였다. 그래서 간격을 계산하여 만들었다.
- 이와 마찬가지로의 문제로 quit을 사용하여 파일을 저장시킬 때, 리눅스에서는 괜찮으나 윈도우에서 wt을 포함한 텍스트 파일을 만들 경우 탭문자가 포함되어 있음에도 불구하고 스페이스바 공백 문자 한 개처럼 보이는 경우가 발생했다. 그래서 이름에 문자 1개를 추가해서 간격이 일정하게 만들어 가시성을 높였다.

#### 5. 결론

- 본 프로젝트인 학생 성적 관리 프로그램을 만들면서 딕셔너리를 활용하는 방법에 대해서

더욱 자세하게 알게 되었다. 또한, 터미널창에서 명령어로 .py 파일을 실행시킬 때, sys.argv값을 활용하여 실행시키는 방법을 알게 되었다.