

# Software Project #1-2

2019년 4월 8일

**Due date: 2019년 4월 27일 토요일 23:59:59 까지**

본 프로젝트에서는 책 정보를 총 6개의 컬렉션에 저장하여 관리하는 프로그램을 구현한다. 사용되는 컬렉션의 종류는 다음과 같이 ArrayList, LinkedList, HashSet, TreeSet, HashMap, TreeMap가 있고, 프로그램에 사용되는 입력 파일은 BookList.txt와 command.txt가 있다. BookList.txt는 책 정보를 담고 있는 텍스트 파일로 책 이름과 책 저자가 '/'로 구분되어 입력되어 있다. 프로그램 시작과 동시에 BookList.txt에 존재하는 책 정보가 모든 컬렉션에 입력되어야 한다. 이후에 command.txt에 입력된 프로그램 명령어들을 읽으면서 원하는 명령을 수행한다. 모든 명령어의 동작이 끝났을 경우 프로그램이 종료되면서 Log.txt 파일을 출력한다. 이 파일에는 command.txt에서 읽었던 명령어들이 수행되면서 나온 결과를 요구되는 포맷에 맞게 모두 저장되어 있다.

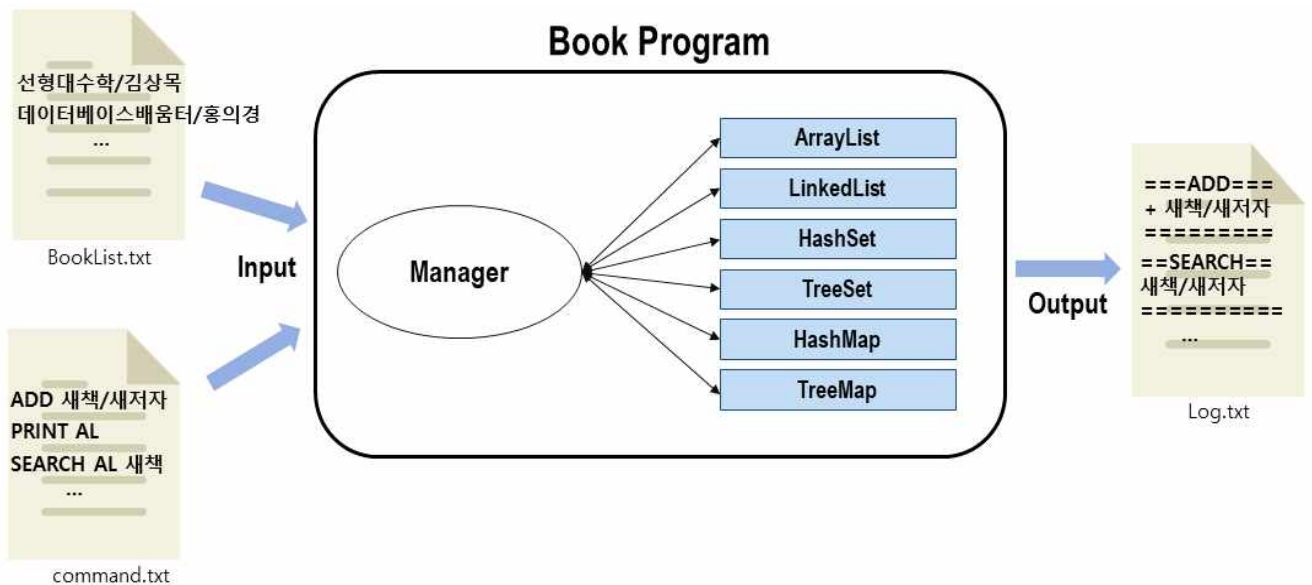


그림 1. Book Program 다이어그램

## □ Program implementation

표 1. 명령어 사용 예 및 기능 설명

명령어	명령어 사용 예 및 기능
ADD	<p>사용 예) ADD 러닝 스파크/홀든 카로</p> <p>새로운 책 정보를 추가하는 명령어로, 책 이름과 저자를 '/' 기준으로 나누어 입력한다. 새로운 책 정보는 해당 프로그램에 속해있는 모든 컬렉션에 동시에 추가되어야 하며, Map 컬렉션 같은 경우에는 key 값으로 책 이름을 사용하도록 설정한다. 책 이름 또는 저자의 정보가 누락되거나 중복된 책 이름이 입력될 때, 그리고 불필요한 인자가 추가로 입력된 경우 에러 코드를 출력한다.</p>
PRINT	<p>사용 예) PRINT AL</p> <p>프로그램에 저장되어있는 책 정보를 모두 출력하는 명령어다. PRINT의 인자로 특정 컬렉션을 지정하여 출력할 수 있다. HashSet, HashMap을 제외하고 모두 책 이름 기준 오름차순으로 정렬한다. 컬렉션에 해당하는 인자 값은 아래와 같다.</p> <p>AL : ArrayList  LL : LinkedList  HS : HashSet  TS : TreeSet  HM : HashMap  TM : TreeMap</p> <p>인자가 입력되지 않았을 때 혹은 없는 인자를 입력했을 때 에러 코드를 출력하고, 불필요한 인자가 추가로 입력된 경우 또한 에러 코드를 출력한다.</p>
SEARCH	<p>사용 예) SEARCH AL 러닝 스파크</p> <p>특정 컬렉션과 책 이름을 인자로 하여 해당 컬렉션에 원하는 책 이름의 정보가 있는지 검색하는 명령어이다. 인자가 입력되지 않았을 때 혹은 책 이름이 컬렉션에 없을 때 에러 코드를 출력하고, 불필요한 인자가 추가로 입력된 경우 또한 에러 코드를 출력한다.</p>

UPDATE	<p>사용 예) UPDATE 러닝 스파크/러닝 하둡</p> <p>책 이름을 수정하는 명령어로, “/”을 기준으로 총 두 개의 인자가 입력으로 들어간다. 첫 번째 인자로 기존 책 이름을 입력하고, 두 번째 인자로 새로 변경할 책 이름을 입력한다. 기존 책 이름이 컬렉션에 존재하지 않을 때 에러 코드를 출력하고, 인자가 입력되지 않거나 불필요한 인자가 추가로 입력된 경우에 에러 코드를 출력한다.</p>
EXIT	<p>사용 예) EXIT</p> <p>프로그램을 종료한다.</p>

## □ Requirements in implementation

- ✓ 모든 명령어는 command.txt에 저장하여 순차적으로 읽고 처리한다.
- ✓ 모든 명령어는 반드시 대문자로 입력한다.  
(ex, PRINT != print )
- ✓ PRINT, SEARCH 시 컬렉션 이름의 대소문자를 구분한다.  
(ex, PRINT al != PRINT AL)
- ✓ 명령어에 인자(Parameter)가 모자라거나 필요 이상으로 입력받을 경우 에러 코드를 출력한다.
- ✓ 책 관리 프로그램에는 중복된 책 이름이 존재하지 않는다.  
- ADD할 때 중복된 책 이름이 존재할 경우 에러를 발생시킨다.
- ✓ BookList.txt에는 최대 100개의 책 정보가 입력될 수 있다고 가정하고, (책 이름)/(책 저자)의 형식을 지키지 않는 경우는 없다고 가정한다.
- ✓ 예외처리에 대해 반드시 에러 코드를 출력한다.
- ✓ 출력은 “출력 포맷”을 반드시 따른다.
- ✓ log.txt 파일에 출력 결과를 반드시 저장한다.  
- log.txt가 이미 존재할 경우 텍스트 파일 가장 뒤에 이어서 추가로 저장한다.

## □ 동작별 에러 코드

동작	에러 코드
ADD	100
PRINT	200
SEARCH	300
UPDATE	400

## □ 출력 포맷

기능	출력 포맷	설명
에러(ERROR)	===== ERROR ===== 100 =====	에러 발생 시 출력 포맷에 맞춰 에러 코드를 출력
ADD	===== ADD ===== + 러닝 스파크/홀든 카로 =====	책 정보 추가 성공 시 출력 포맷에 맞춰 결과를 출력
PRINT	===== PRINT ===== 걸리버 여행기/조너스 스위프트 노인과 바다/어니스트 헤밍웨이 마지막 잎새/오 헨리 ... =====	출력 포맷에 맞춰 결과를 출력 해당 컬렉션에 책 정보가 없을 경우 에러 출력
SEARCH	===== SEARCH ===== 걸리버 여행기/조너스 스위프트 =====	책 정보 탐색에 성공 시 출력 포맷에 맞춰 결과를 출력 책 정보가 없을 경우 에러 출력
UPDATE	===== UPDATE ===== 걸리버 여행기 -> 걸리버 일기 =====	업데이트할 책 정보가 존재할 경우 출력 포맷에 맞춰 결과를 출력 업데이트할 책 정보가 존재하지 않을 경우 에러 출력
EXIT	===== EXIT===== Success =====	출력 포맷에 맞게 출력

## □ 동작 예시

command.txt
ADD 러닝 스파크/홀든 카로 PRINT AL SEARCH AL 러닝 스파크 UPDATE 러닝 스파크/러닝 하둡 PRINT HS EXIT

실행 후 log.txt

=====ADD=====

+ 러닝 스파크/홀든 카로

=====

=====PRINT=====

걸리버 여행기/조너스 스위프트

노인과 바다/어니스트 헤밍웨이

러닝 스파크/홀든 카로

마지막 잎새/오 헨리

앵무새 죽이기/하퍼 리

오만과 편견/제인 오스틴

위대한 개츠비/프랜시스 스콧 피츠제럴드

제인 에어/샬롯 브론테

죽은 시인의 사회/클라인바움

툼 소녀의 모험/마크 트웨인

호밀밭의 파수꾼/제럴드 데이비드 샬린저

1984/조지오웰

=====

=====SEARCH=====

러닝 스파크/홀든 카로

=====

=====UPDATE=====

러닝 스파크 -> 러닝 하둡

=====

=====PRINT=====

걸리버 여행기/조너스 스위프트

앵무새 죽이기/하퍼 리

오만과 편견/제인 오스틴

마지막 잎새/오 헨리

호밀밭의 파수꾼/제럴드 데이비드 샬린저

러닝 하둡/홀든 카로

죽은 시인의 사회/클라인바움

제인 에어/샬롯 브론테

1984/조지오웰

노인과 바다/어니스트 헤밍웨이

툼 소녀의 모험/마크 트웨인

위대한 개츠비/프랜시스 스콧 피츠제럴드

=====

=====EXIT=====

Success

=====

## □ 구현 시 반드시 정의해야하는 Class

- ✓ main.java                                - 메인 클래스
- ✓ BookNode.java                           - 책 노드 클래스
- ✓ myCollection.java                       - 컬렉션을 관리하는 클래스
- ✓ Manager.java                            - Manager 클래스(다른 클래스들의 동작을 관리하여 프로그램을 전체적으로 조정하는 역할을 수행)

## □ Files

- ✓ BookList.txt                            : 프로그램 시작시 입력할 책 정보들이 저장되어 있는 파일
- ✓ command.txt                            : 프로그램을 동작시키는 명령어들을 저장하고 있는 파일
- ✓ log.txt                                  : 프로그램 출력 결과를 모두 저장하고 있는 파일

## □ 제한사항 및 구현 시 유의사항

- ✓ 반드시 제공되는 압축파일(skel\_sp\_p1-2.zip)를 이용하여 구현하며 작성된 소스 파일의 이름과 클래스의 이름과 함수 이름을 변경하지 않는다.
- ✓ 클래스의 함수 및 변수와 파라미터는 자유롭게 추가 구현이 가능하다.
- ✓ 제시된 Class를 각 기능에 알맞게 모두 사용한다.
- ✓ 프로그램 구조에 대한 디자인이 최대한 간결하도록 고려하여 설계한다.
- ✓ 채점 시 코드를 수정해야 하는 일이 없도록 한다.
- ✓ 주석은 반드시 작성한다. (없으면 감점)

## □ 제출기한 및 제출방법

- ✓ 제출기한
  - 2019년 4월 27일 토요일 23:59:59 까지 제출
- ✓ 보고서 작성 형식
  - Introduction : 해당 프로젝트에 대한 설명
  - Algorithm : 구현한 프로그램에 대한 코드 설명 및 동작 설명
  - Result : 구현한 프로그램에 대한 명령어별 결과 캡처
  - Consideration : 프로젝트에 대한 고찰 작성(최소 5줄 이상)
- ✓ 제출방법
  - 소스코드와 보고서 파일(pdf)을 함께 압축하여 u-campus에 제출
    - 확장자가 .java/.pdf가 아닌 파일은 제출하지 않음
    - 보고서 파일 확장자가 pdf가 아닐 시 감점
- ✓ 제출 형식
  - (학번)\_project1-2.zip