Project: Implementing a Simple Database Application

Due: 2024/06/14 (금), 11:59 PM

이번 프로젝트의 목표는 데이터베이스를 이용하는 어플리케이션을 설계하고 구현하는 것이다. **Python** 및 **MySQL**을 이용하여 도서 대출을 시뮬레이션 하는 어플리케이션을 만들어 본다. 이번 프로젝트를 통해 학생들은 어플리케이션과 데이터베이스를 연동하는 방법을 배우게 된다.

1 주요 기능

- 데이터는 관계형 데이터베이스에 저장되어야 한다.
- 주어진 데이터로 데이터베이스를 **초기화** 할 수 있어야 한다. [세부 사항 1]
- 도서 데이터 및 회원 데이터를 **출력**할 수 있어야 한다. [세부 요구사항 2, 3]
- 도서 데이터 및 회원 데이터를 **추가/삭제**할 수 있어야 한다. [세부 사항 4, 5, 6, 7]
- 회원이 도서를 대출할 수 있어야 한다. [세부 사항 8]
- 회원이 도서를 반납하고 평점을 부여할 수 있어야 한다. [세부 사항 9]
- 회원 별로 대출 중인 도서의 정보를 출력할 수 있어야 한다. [세부 사항 10]
- 도서명으로 도서를 검색 할 수 있어야 한다. [세부 사항 11]
- 특정 회원에게 도서를 추천할 수 있어야 한다. [세부 사항 12, 13]
- 프로그램 종료 및 데이터베이스 리셋과 생성이 가능해야 한다. [세부 사항 14, 15]

2 세부 사항

- 특정 입력에 대해 에러가 있을 시 그 즉시 에러 메시지를 출력하고 해당 명령을 종료한다.
- project2-msg.pdf 파일에 기재된 모든 상황에 대응 가능해야 한다.
- *메시지*는 project2-msg.pdf 파일을 기반으로 상황에 맞는 문구를 출력한다.
- 모든 출력은 **첫 번째 컬럼**에 대해 **오름차순**으로 정렬되어야 한다.
- 출력 포맷은 3. 실행 예시를 참조한다.

- **해당 문서에 명시되지 않은 상황에 대해서는 자유롭게 가정하고, 해당 사항은 보고서에 명시한다.**

구현한 도서관 관리 어플리케이션은 다음과 같은 기능을 갖고 있다.

1. **데이터베이스 초기화**
    - 주어진 raw data (data.csv) 로 데이터베이스를 초기화한다.
    - 초기화 완료 시, *성공 메세지*를 출력한다.
        - 주어진 **data.csv** 파일은 특정 시점까지의 회원들의 도서 평가 기록이다.
        - 주어진 **data.csv** 의 각 column이 의미하는 바는 순서대로 아래와 같다.
            - 도서 ID: 평가된 도서의 고유 ID
            - 도서명: 평가된 도서명
            - 저자명: 평가된 도서의 저자명
            - 회원 ID: 해당 도서를 평가한 회원의 고유 ID
            - 회원명: 해당 도서를 평가한 회원 명
            - 평점: 해당 회원이 해당 도서에 남긴 평점
        - 도서관은 각 도서를 최대 1권씩만 보유할 수 있다.
        - 초기화 시 대출 중인 도서는 없으며, 따라서 모든 도서의 대출가능 권수는 1권으로 초기화 된다.
        - 데이터베이스 스키마 및 구성 방식은 프로그램의 기능들을 고려해 자유롭게 설계한다.
            - 데이터베이스 구성에 대한 설명은 보고서에 작성한다.

2. **모든 도서 정보 출력**
    - 데이터베이스에 존재하는 모든 도서의 정보를 출력한다.
        - 각 column은 도서 ID, 도서명, 저자명, 평균 평점, 대출가능 권 수 순으로 출력한다.
        - [참고1 - 평균 평점 규칙]
          도서의 평균 평점은 해당 도서에 남겨진 평점들의 산술평균이다.

          한 회원이 같은 도서에 평점을 여러 번 남기는 경우에는, 가장 최근의 평점 1건만 계산에 반영된다.

          도서에 대한 평점이 존재하지 않는다면 '*None*' 을 출력한다.

- 각 row는 도서 ID를 기준으로 오름차순으로 출력한다.

3. **모든 회원 정보 출력**
   - 데이터베이스에 존재하는 모든 회원의 정보를 출력한다.
   - 각 column은 회원 ID, 이름 순으로 출력한다.
   - 각 row는 회원 ID를 기준으로 오름차순으로 출력한다.

4. **도서 추가**
   - 새로운 도서를 추가한다.
   - 도서명, 저자를 입력한다.
   - 도서명은 최소 1자, 최대 50자까지 입력 가능하다.
     - 1자 미만, 50자 초과하는 도서명을 입력 받으면 *에러 메시지*를 출력한다.
   - 저자명은 최소 1자, 최대 30자까지 입력 가능하다.
     - 1자 미만, 30자 초과하는 도서명을 입력 받으면 *에러 메시지*를 출력한다.
   - 도서명과 저자명의 조합으로 고유성을 갖는다.
     - 도서명과 저자명이 모두 동일한 도서를 추가하려 할 경우 *에러 메시지*를 출력한다
   - 도서의 ID를 부여하는 규칙은 아래와 같다.
     - 도서 ID는 새로운 도서가 추가될 때마다 자동 부여한다.
     - 도서 ID는 1부터 시작하는 *AUTO_INCREMENT* 옵션을 사용한다.
   - 도서 추가 성공 시, *성공 메세지*를 출력한다.

5. **회원 등록**
   - 새로운 회원을 추가한다.
   - 이름을 입력한다.
   - 동일한 이름을 가진 서로 다른 회원이 존재 할 수 있다.
   - 이름은 최소 1글자, 최대 10글자로 제한한다.
     - 이름 제한 조건에 맞지 않는 값을 입력 받으면 *에러 메시지*를 출력한다.

- 편의를 위해 각 회원에 별도의 ID 값을 부여한다.
    - ID는 새로운 회원이 추가될 때마다 자동 부여한다.
    - ID는 1부터 시작하는 *AUTO_INCREMENT* 옵션을 사용한다.
- 회원 등록 성공 시, *성공 메세지*를 출력한다.

6. **도서 삭제**
    - 도서를 삭제한다.
    - 도서 ID를 입력한다.
        - 삭제하려는 ID를 가진 도서가 존재하지 않는다면 *에러 메시지*를 출력한다.
    - 도서를 삭제할 때 해당 도서에 관련된 대출 정보 및 평점 정보도 모두 삭제되어야 한다.
    - 대출 중인 상태의 도서는 삭제할 수 없다.
        - 대출 중인 상태의 도서를 삭제하려고 할 시 *에러 메시지*를 출력한다.
    - 도서 삭제 성공 시, *성공 메세지*를 출력한다.

7. **회원 삭제**
    - 회원을 삭제한다.
    - 회원의 ID를 입력한다.
        - 회원을 삭제할 때 그 회원에 관련된 대출정보 및 평점 정보도 모두 삭제되어야 한다.
        - 삭제하려는 ID를 가진 회원이 존재하지 않는다면 *에러 메시지*를 출력한다.
    - 도서를 대출 중인 상태의 회원은 삭제할 수 없다.
        - 도서를 대출 중인 상태의 회원을 삭제하려고 할 시 *에러 메시지*를 출력한다.
    - 회원 삭제 성공 시, *성공 메세지*를 출력한다.

8. **도서 대출**
    - 회원이 도서를 대출한다.
    - 회원은 동시에 최대 도서 2권까지 대출이 가능하다.
    - 도서 ID, 회원 ID을 입력한다.

        - 해당 ID의 도서가 존재하지 않는다면 *에러 메시지*를 출력한다.
        - 해당 ID의 회원이 존재하지 않는다면 *에러 메시지*를 출력한다.
        - 해당 ID의 도서의 대출 가능 권수가 0권이라면 *에러메세지*를 출력한다.
        - 해당 회원이 이미 최대 개수의 도서를 대출 중인 상태라면 *에러메세지*를 출력한다.
    - 대출된 도서는 대출 가능 권수가 0으로 줄어든다.
    - 도서 대출 성공 시, *성공 메세지*를 출력한다.

9. **도서 반납과 평점 부여**
    - 회원이 도서를 반납하며 동시에 도서를 평가한다.
        - 도서 ID, 회원 ID, 평점(1 이상 5 이하의 정수)을 입력 받는다.
            - 해당 ID의 도서가 존재하지 않는다면 *에러 메시지*를 출력한다.
            - 해당 ID의 회원이 존재하지 않는다면 *에러 메시지*를 출력한다.
            - 평점이 유효한 범위(1 이상 5 이하의 정수)를 벗어나면 *에러 메시지*를 출력한다.
            - 회원이 반납 및 평가하려는 도서가 해당 회원이 대출 중인 도서가 아니라면 *에러 메시지*를 출력한다.
        - 해당 회원이 이미 해당 도서에 남긴 평점이 존재 할 경우에는, 현재 평점으로 덮어쓴다.
        - 반납과 평가가 완료된 도서는 대출 가능 권수가 1권으로 복원된다.
    - 도서 반납 및 평가 성공 시, *성공 메세지*를 출력한다.

10. **회원이 대출 중인 도서 정보 출력**
    - 회원이 대출 중인 모든 도서의 정보를 출력한다.
        - 회원 ID를 입력한다.
        - 해당 회원이 현재 대출 중인 도서 정보를 출력한다.
            - 각 column은 도서 ID, 도서명, 저자명, 평균 평점 순으로 출력한다.
            - 각 row는 도서 ID 오름차순으로 출력한다
            - 평균 평점에 관한 규칙은 세부사항 2 -[참고1 - 평균 평점 규칙] 을 참고한다.
        - 해당 ID의 회원이 존재하지 않는다면 *에러 메시지*를 출력한다.

11. **도서 검색**

- 검색어를 입력받고, **도서명**에 검색어가 포함된 도서를 모두 출력한다.
- 검색은 case insensitive 하여야 한다. (e.g. 'gatsby'로 검색 시 'The Great Gatsby'도 검색결과에 포함되어야 함)
- 각 column은 도서 ID, 도서명, 저자명, 평균 평점, 대출가능 권 수 순으로 출력한다.
- 평균 평점에 관한 규칙은 세부사항 2 -[참고1 - 평균 평점 규칙] 을 참고한다.
- 각 row는 도서 ID를 기준으로 오름차순으로 출력한다.

12. **회원을 위한 도서 추천 1**

특정 회원이 평점을 남긴 적 없는 도서 중 **평균 평점이 가장 높은 도서 1개**와 **가장 평점이 많은 도서 1개**를 추천한다.

- 회원 ID를 입력한다.
  - 해당 ID의 회원이 존재하지 않는다면 *에러 메시지*를 출력한다.
- 평균 평점이 가장 높은 도서와 가장 평점이 많은 도서를 순서대로 출력한다.
  - 각 column은 도서 ID, 도서명, 저자명, 평균 평점 순으로 출력한다.
    - 평균 평점에 관한 규칙은 세부사항 2 -[참고1 - 평균 평점 규칙] 을 참고한다.
  - 이 때, 대출 중이지만 평점을 부여하지 않은 도서도 추천 대상에 포함된다.
  - 평균 평점이 가장 높은 도서가 여럿일 경우, 그 중 ID값이 가장 작은 도서를 추천한다.
  - 가장 평점이 많은 도서가 여럿일 경우, 그 중 ID값이 가장 작은 도서를 추천한다.

13. **회원을 위한 도서 추천 2**

User-based Collaborative Filtering(CF)을 사용해서 회원이 평을 남기지 **<u>않은</u>** 도서 중 **회원이 가장 높은 평점을 부여할 것이라고 예상되는 하나의 도서를 추천**한다.

CF 에도 다양한 방법들이 존재하지만, 본 과제에서는 다음과 같은 **User-based Collaborative Filtering**을 사용해 회원을 위한 도서를 추천한다.

A. 회원이 도서에 부여한 평점을 기반으로 user-item matrix를 구축한다.

- 회원의 수가 n, 도서의 수가 m 인 경우 n×m의 matrix가 생성된다.
- 이 때, 대출 중이지만 평점을 부여하지 않은 도서도 추천 대상에 포함된다.

- 예시: User 4가 아직 평점을 부여하지 않은 Item 2, Item 3에 대한 평점을 예측해서 User 4에게 가장 적합한 Item을 추천하는 것을 목표로 한다.

	Item 1	Item 2	Item 3	Item 4
User 1	5	4	4	
User 2	1	3		1
User 3		2	1	4
User 4	2	?	?	5

B. 회원이 아직 평점을 부여하지 않은 도서에 대해서는 해당 회원이 이미 부여한 평점들의 평균을 임시 평점으로 활용한다.

- 예시: User1의 경우 Item4에 부여한 평점 정보가 없기 때문에, 기존에 부여한 평점들의 평균인 $\frac{5+4+4}{3} = 4.33$ 을 Item4의 임시 평점으로 부여한다.
- 유저가 아직 아무 도서에도 평점을 남기지 않은 경우 해당 유저의 모든 도서에 대한 임시 평점은 0이 된다.

	Item 1	Item 2	Item 3	Item 4
User 1	5	4	4	4.33
User 2	1	3	1.67	1
User 3	2.33	2	1	4
User 4	2	3.5	3.5	5

C. 각 회원이 부여한 평점의 유사도를 측정한다.

- 두 회원이 부여한 평점의 유사도가 높은 경우 두 회원의 취향이 유사하다고 판단한다.
- 다양한 유사도 측정 방식 중, **cosine similarity**를 사용한다.

$$cosine - similarity = \frac{A \cdot B}{\|A\| \cdot \|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

- 모든 회원 사이의 유사도를 구해서 n×n의 similarity matrix를 생성한다.
  - Tip 1: 자기 자신과의 유사도는 항상 1이다. 그러므로 similarity matrix의 대각 성분들은 모두 1이다.
  - Tip 2: User 1과 User 2의 유사도와 User 2와 User 1의 유사도는 동일하기 때문에 similarity matrix는 symmetric하게 구성된다.
- 예시: User1과 User2 사이의 cosine similarity를 구하면, $\frac{5\times1+4\times3+4\times1.67+4.33\times1}{\sqrt{5^2+4^2+4^2+4.33^2}\sqrt{1^2+3^2+1.67^2+1^2}} = \frac{28.1}{32.36} = 0.868 \approx 0.87$

	User 1	User 2	User 3	User 4
User 1	1.0	0.87	0.92	0.94
User 2	0.87	1.0	0.73	0.86
User 3	0.92	0.73	1.0	0.93
User 4	0.94	0.86	0.93	1.0

D. 예측하고자 하는 회원과 나머지 모든 회원과의 similarity를 weight으로 하는 weighted sum 값으로 각 도서에 대한 예측 평점을 구한다.

	Item 1	Item 2	Item 3	Item 4
User 1	5	4	4	4.33
User 2	1	3	1.67	1
User 3	2.33	2	1	4
User 4	2	①	②	5

- 예시: User 4가 아직 평점을 부여하지 않은 Item 2(①)와 Item 3(②)의 예측 평점을 다음과 같이 구할 수 있다.

$$①=\frac{0.94\times4+0.86\times3+0.93\times2}{0.94+0.86+0.93}=\frac{8.2}{2.73}=3.00$$

$$②=\frac{0.94\times4+0.86\times1.67+0.93\times1}{0.94+0.86+0.93}=\frac{6.13}{2.73}=2.25$$

- 추천 후보 (Item 2, Item 3) 중 Item 2의 예상 평점이 3.0으로 다른 모든 추천 대상들의 예상 평점보다 높기 때문에, User 4에게는 Item 2를 추천하게 된다.

	Item 1	Item 2	Item 3	Item 4
User 1	5	4	4	4.33
User 2	1	3	1.67	1
User 3	2.33	2	1	4
User 4	2	3.00	2.25	5

특정 회원에게 도서 추천 시, 아래와 같은 형식으로 입출력이 이루어진다.

- 회원 ID를 입력한다.
  - 해당 ID의 회원이 존재하지 않는다면 *에러 메시지*를 출력한다.
- 해당 회원이 가장 높은 평점을 부여할 것으로 예상되는 도서 정보를 출력한다.
  - 각 column은 도서 ID, 도서명, 저자명, 평균 평점, 예측 평점 순으로 출력한다.
    - 평균 평점에 관한 규칙은 세부사항 2 -[참고1 - 평균 평점 규칙] 을 참고한다.
    - 예측 평점을 구하기 위한 임시 값들은 실제 평점 기록에 영향을 미치지 않는다.
  - 예상된 평점이 같은 경우 ID값이 작은 도서를 추천한다.

14. 프로그램 종료

- 프로그램 종료 후 재시작 시, 데이터베이스는 이전 상태로 보존되어있어야 한다.
- 프로그램 종료 전, *종료 메세지*를 출력한다.

15. 데이터베이스 리셋 및 생성

- 기존에 저장해둔 테이블 및 데이터가 존재 할 경우 모두 삭제한다.
  - 삭제 실시 전 확인 메시지를 띄우고 사용자 입력(y/n)을 받아야 한다.

- 저장된 데이터가 없는 상태에서, 세부 사항 1.을 수행해 데이터베이스를 초기화한다.
- 따라서 세부 사항 1.의 *성공 메세지*가 출력되게 된다.

3 실행 예시

- 프로그램의 출력은 **검은색** 글씨로 표시되어 있다.
- 유저의 입력은 빨간색 글씨로 표시되어 있다.

```
============================================================
1. initialize database
2. print all books
3. print all users
4. insert a new book
5. remove a book
6. insert a new user
7. remove a user
8. check out a book
9. return and rate a book
10. print borrowing status of a user
11. search books
12. recommend a book for a user using popularity-based method
13. recommend a book for a user using user-based collaborative filtering
14. exit
15. reset database
============================================================
Select your action: 1
Database successfully initialized

Select your action: 2
--------------------------------------------------------------------------------
id      title           author          avg.rating              quantity
--------------------------------------------------------------------------------
1       The Matrix      Lana Wachowski   4                      1
2       WALL-E          Andrew Stanton   4                      1
--------------------------------------------------------------------------------

Select your action: 3
--------------------------------------------------------------------------------
id      name
--------------------------------------------------------------------------------
1       Bae Sangwhan
2       Kim Jihoon
3       Choi Byungseo
--------------------------------------------------------------------------------

Select your action: 4
Book title: Alien
Book author: Ridley Scott
One book successfully inserted

Select your action: 6
User name: My User Name
```

```
One user successfully inserted

Select your action: 2
--------------------------------------------------------------------------------
id      title                           author          avg.rating      quantity
--------------------------------------------------------------------------------
1       The Matrix                      Lana Wachowski  4               1
2       WALL-E                          Andrew Stanton  4               1
3       Alien                           Ridley Scott    None            1
--------------------------------------------------------------------------------

Select your action: 11
Query: matrix
--------------------------------------------------------------------------------
id      title                           author          avg.rating      quantity
--------------------------------------------------------------------------------
1       The Matrix                      Lana Wachowski  4               1
--------------------------------------------------------------------------------

Select your action: 3
--------------------------------------------------------------------------------
id      name
--------------------------------------------------------------------------------
1       Bae Sangwhan
2       Kim Jihoon
3       Choi Byungseo
4       My User Name
--------------------------------------------------------------------------------

Select your action: 8
User ID: 1
Book ID: 1
Book successfully checked out

Select your action: 8
User ID: 1
Book ID: 1
Cannot check out a book that is currently borrowed

Select your action: 8
User ID: 1
Book ID: 3
Book successfully checked out

Select your action: 8
User ID: 1
Book ID: 2
User 1 exceeded the maximum borrowing limit

Select your action: 2
--------------------------------------------------------------------------------
id      title                           author          avg.rating      quantity
--------------------------------------------------------------------------------
1       The Matrix                      Lana Wachowski  4               0
2       WALL-E                          Andrew Stanton  4               1
3       Alien                           Ridley Scott    None            0
--------------------------------------------------------------------------------

Select your action: 9
User ID: 1
Book ID: 3
Ratings (1~5): 5
Book successfully returned and rated
```

```
Select your action: 2
-------------------------------------------------------------------------------
id      title                           author          avg.rating       quantity
-------------------------------------------------------------------------------
1       The Matrix                      Lana Wachowski   4                 0
2       WALL-E                          Andrew Stanton   4                 1
3       Alien                           Ridley Scott     5                 1
-------------------------------------------------------------------------------

Select your action: 10
User ID: 1
-------------------------------------------------------------------------------
id      title                           author
-------------------------------------------------------------------------------
1       The Matrix                      Lana Wachowski
-------------------------------------------------------------------------------


Select your action: 12
User ID: 2
-------------------------------------------------------------------------------
Rating-based
-------------------------------------------------------------------------------
id      title                           author          avg.rating       quantity
-------------------------------------------------------------------------------
3       Alien                           Ridley Scott     5                 1
-------------------------------------------------------------------------------
Popularity-based
-------------------------------------------------------------------------------
id      title                           author          avg.rating       quantity
-------------------------------------------------------------------------------
1       The Matrix                      Lana Wachowski   4                 0
-------------------------------------------------------------------------------

Select your action: 13
User ID: 2
-------------------------------------------------------------------------------
id      title                           author          avg.rating       exp.rating
-------------------------------------------------------------------------------
1       The Matrix                      Lana Wachowski   4                 4
-------------------------------------------------------------------------------

Select your action: 14
Bye!
```

4 개발 환경

- Python 3.8+
- MySQL
  - DB 접속 정보
    - Host: astronaut.snu.ac.kr
    - Port: 7001
    - ID: DB학번(예:DB2024_12345)

- Password: ID와 동일

- 접속 후 password 변경 방법
  - set password = password('비밀번호')

5 배점 (100점)

1. **구현 완성도 (90점)**
   - **기능 완성도 (70점)**
     - 채점 과정에서 형식은 동일하지만 다른 테스트 데이터가 쓰일 수 있다.
     - Test case는 연속된 입력으로 구성되어있으며, 연속된 각 입력에 대해 요구사항이 정확하게 동작해야 한다.
   - **데이터베이스 스키마 설계 (10점)**
     - Completeness, Correctness, Minimality, Normality (BCNF & 3NF), 등을 고려하며 설계한다. (강의 슬라이드 6: 47-58pg. 참고)
   - **청결한 코드 (10점)**
     - 적절한 모듈화 및 주석 첨가로 코드 가독성 확보
2. **보고서 (10점)**
   - 본인이 설계한 데이터베이스 스키마 설명 (5점)
   - 프로젝트 결과물 설명 (5점)
     - 핵심 모듈과 알고리즘에 대한 간략한 설명
     - 구현한 내용에 대한 간략한 설명
     - 컴파일과 실행 방법
     - 프로젝트를 하면서 느낀 점
     - 기타 (optional)

       (제시된 기능 중 구현하지 못한 부분이 있다면) 구현하지 못한 내용

       프로젝트 진행하면서 가정한 것들 (추가적으로 처리한 예외처리 등)

6 제출

1. **DB**
   - [중요] 프로젝트 제출시 채점을 위해 자신의 DB 계정 안의 테이블 스키마만 남기고 레코드는 모두 truncate해야 한다.
   - 이를 지키지 않았을 시 채점이 제대로 되지 않아 0점 처리될 수 있으니 주의한다.
2. **프로젝트 코드 압축 파일**
   - **폴더명**: PRJ2_학번 (예: PRJ2_2024-12345.zip)
   - 프로젝트 코드 최상단에서 "**`python run.py`**" 명령어를 통해 실행시킬 수 있어야 한다.
   - **`run.py`**와 **`data.csv`** 파일은 같은 폴더 내에 위치한다.
   - 외부 라이브러리 `numpy`와 `pandas`를 사용해도 되며, 설치가 필요한 **모든** 라이브러리는 requirements.txt 파일에 명시한다.
3. **리포트**
   - **파일명**: PRJ2_학번.pdf (예: PRJ2_2024-12345.pdf)
   - **pdf 포맷**으로 제출

상기한 자료를 압축하여 ETL에 제출

- 제출 ZIP 파일 구조

```
PRJ2_학번.zip
|- PRJ2_학번.zip
|   |- run.py
|   |- (Other files if needed)
|   |- requirements.txt
|   |- data.csv
|- PRJ2_학번.pdf
```

- **기한 끝난 후 과제 제출 시 늦은 시간 별 감점 기준**

  0~24 시간: 10% 감점

  24~48 시간: 20% 감점

48시간 이후: 점수 없음

- **부정 행위는 0점 처리**

  다른 사람의 코드를 참조하는 행위

  이전에 수강한 사람의 코드를 참조하는 행위

  프로그램을 통해 표절 여부 확인 예정