

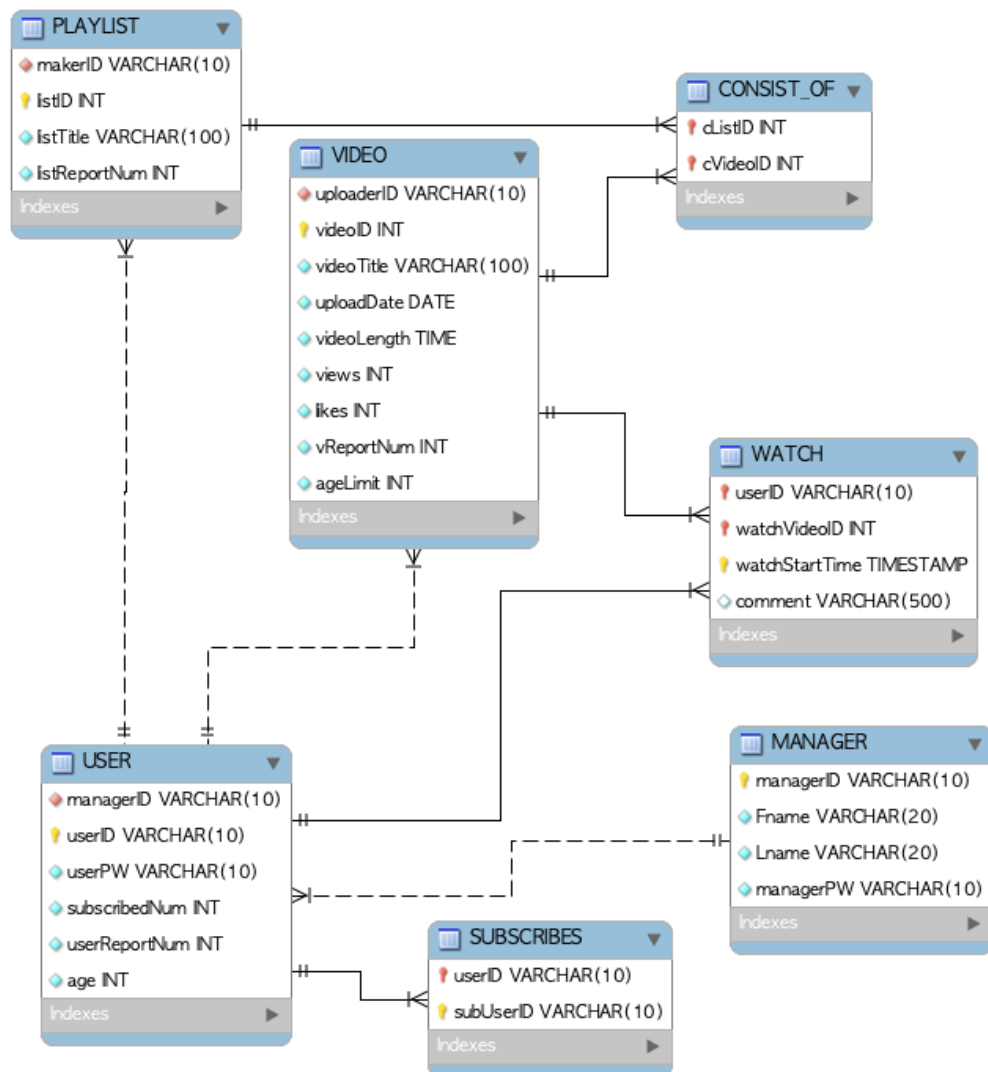
## Project 2 - 동영상 업로드 및 공유 플랫폼 프로그램 개발

컴퓨터소프트웨어학부 2021025205 강태욱

### 1. 프로그램 설명

- 이 프로그램은 Manager 프로그램과 User 프로그램으로 나누어진다.
- Manager 프로그램은 관리자 프로그램으로, 동영상 업로드 및 공유 시스템을 관리하는데 사용된다.
- User 프로그램은 사용자 프로그램으로, 동영상 업로드 및 공유 플랫폼을 이용하는데 사용된다.

### 2. Crow-foot Diagram



### 3. 프로그램 설명 - Manager

- a. 관리자는 특정 사용자들을 관리한다. 관리자는 자신이 관리하지 않는 사용자의 정보를 임의로 수정할 수 없다.
- b. 관리자는 특정 사용자의 영상 업로드 내역, 시청 기록, 사용자가 만든 영상 재생 리스트를 열람할 수 있다.
- c. 관리자는 신고 기록이 100 회 이상인 이용자의 계정, 영상, 영상 재생 리스트를 삭제할 수 있다. 이때, 사용자의 계정이 삭제된다면 그 사용자가 올린 영상, 만든 영상 재생 리스트는 같이 삭제된다.
- d. 관리자는 자신이 관리하는 사용자의 시청 내역을 삭제할 수 있다.
- e. 관리자는 자신이 관리하는 영상의 나이 제한을 임의로 설정할 수 있다.
  - i. 이 부분은 요구사항 분석 단계에서 명시되지 않은 것으로, 개발 과정에서 추가로 구현하였다.
- f. 관리자는 자신이 관리하는 이용자의 관리자를 타인으로 변경할 수 있다.
- g. 관리자는 자신의 비밀번호, 이름을 변경할 수 있다.
- h. Manager 프로그램을 통해 새로운 관리자를 생성할 수 있다.
- i. Manager 프로그램을 통해 기존 관리자를 삭제할 수 있다.
- j. Manager ID 를 통해 manager 를 구분한다.

### 4. 프로그램 설명 - User

- a. 사용자는 다른 사용자가 올린 영상을 볼 수 있다. 이때, 사용자의 나이가 특정 영상의 나이 제한에 걸린다면 사용자는 그 영상을 볼 수 없다.
  - i. 이 부분은 요구사항 분석 단계에서 명시되지 않은 것으로, 개발 과정에서 추가로 구현하였다.
- b. 사용자는 다른 사용자가 만든 영상 재생 리스트를 사용할 수 있다.
- c. 사용자는 영상을 업로드하고 자신의 업로드 영상 목록을 관리할 수 있다.
  - i. 한 사용자는 자신이 올린 영상들과 중복된 이름의 영상을 올리지 못하는 것으로 바꾸었다. 즉, 각 영상은 사용자 ID 와 영상 제목으로 서로를 구분할 수 있다.
- d. 사용자는 본인이 업로드한 영상을 삭제할 수 있다.
- e. 사용자는 다른 사용자가 올린 영상에 좋아요를 남길 수 있다.
- f. 사용자는 다른 사용자를 구독 및 구독 취소할 수 있다.
- g. 사용자는 특정 영상에 댓글을 남길 수 있다. 다른 사용자는 영상에 달린 댓글을 볼 수 있다.
  - i. 영상에 대한 댓글의 개수를 제한하는 것이 큰 의미가 없을 것 같아 영상의 댓글 개수 제한(1000 개)을 없앴다.
- h. 사용자는 특정 영상을 신고할 수 있다.
- i. 사용자는 자신만의 영상 재생 리스트를 만들 수 있다.

- i. 한 사용자는 자신이 만든 영상 재생 리스트와 중복된 이름의 영상 재생 리스트를 만들지 못한다. 즉, 각 영상 재생 리스트는 리스트 ID 와 제작자 ID 로 서로를 구분할 수 있다.
- j. 사용자는 영상 재생 리스트 안의 영상을 볼 수 있다.
  - i. 영상을 보는 과정에서 영상 재생 리스트의 총 길이를 나타내고자 했으나, 자료형의 문제로 이 부분을 구현하지 못하였다.
  - ii. 영상 재생 리스트 내의 영상 개수를 제한하는데 큰 의미가 있을 것 같지 않아 영상 재생 리스트의 영상 개수 제한(100 개)을 없앴다.
  - iii. 나이 제한에 걸리는 영상은 보여지지 않도록 구현하였다.
- k. 사용자는 영상 재생 리스트 내의 영상을 영상 업로드 날짜, 영상 길이, 재생 횟수를 기준으로 정렬하여 볼 수 있다.
  - i. 영상 재생 리스트 내의 영상 순서에 대한 제약(stack)을 없앴다.
- l. 사용자는 특정 영상 재생 리스트를 신고할 수 있다.
- m. 사용자는 영상 관람 시 업로더의 구독자 수를 볼 수 있다.
- n. 사용자는 영상 관람 시 업로더를 신고할 수 있다.
- o. 사용자는 자신이 구독하는 채널을 볼 수 있다.
- p. 사용자는 자신의 시청 기록을 볼 수 있다.
  - i. 시청 기록은 사용자 ID, 영상 ID, 시청 시작 시각을 통해 서로 구분된다.
- q. User ID 를 통해 User 를 구분한다.

## 5. 프로그램 코드에 대한 설명 - JDBC Class

- a. 이 Class 는 jdbc 드라이버를 연동하기 위해 사용하는 Class 이다.
- b. Manager Class 와 User Class 모두 이 Class 를 이용해 MySQL 에 query 를 던진다.
- c. getStatement() method 는 VIDEO\_PLATFORM Schema 와 본 프로그램을 연결하는 statement 를 반환한다.
- d. closeConnection method 는 statement 을 닫고, mysql 과의 connection 을 끊어 query 로 인한 변화 사항이 DB 에 완전히 반영되도록 한다.

## 6. 프로그램 코드에 대한 설명 - Manager Class

- a. startManager() : 로그인, 매니저 생성, 매니저 삭제, 프로그램 종료 기능을 숫자로 입력받아 그 역할을 하는 method 로 연결시켜 주는 method 이다. 잘못된 입력이 들어오면 경고 메시지 출력 후 다시 숫자를 입력받는다.
- b. loginManager() : manager account 에 로그인 하는 기능을 구현한 method 이다. DB 에 입력 받은 ID,비밀번호와 일치하는 계정 정보가 있다면 그 계정으로 manager 프로그램에 로그인한다. 만약 그렇지 않다면, ID 와 비밀번호를 다시 입력 받거나 startManager method 로 돌아간다.

- c. afterLogin() : Users, Videos, Playlists 목록을 보고 관리하는 기능, Users, Videos, Playlists 를 삭제하는 기능, manager 정보를 수정하는 기능, 프로그램 종료 기능을 각각의 숫자로 입력받아 그 역할을 하는 method 로 연결시켜 주는 method 이다. 잘못된 입력이 들어오면 경고 메세지 출력 후 다시 숫자를 입력 받는다.
- d. show() : Users, Videos, Playlists 를 보여주고 그들을 관리하는 기능을 하는 method 로 연결시켜주는 method 이다.
  - i. showUserList() : 관리자가 관리하는 User 의 목록을 보여주는 method 이다. 목록을 보여준 후 managerUser method 를 통해 사용자를 관리한다.
    - 1. manageUser() : User 의 관리자 변환, User 의 영상 업로드 목록 보여주기, User 의 시청 기록을 보여주는 기능을 하는 method 로 연결시켜주는 method 이다.
    - 2. transferUserManagement() : User 의 관리자를 변환시키는 method 이다. user 의 ID 와 새로운 관리자의 ID 를 입력받아 이를 DB 에 반영한다.
    - 3. showUploadList() : User 가 올린 영상의 목록을 보여주는 method 이다. 각각의 영상 제목, 영상 ID, 업로드 날짜, 영상 길이, 조회수, 좋아요 수, 신고 횟수, 나이 제한을 보여준다. 이 method 에서는 영상을 삭제할 수 없다.
    - 4. sshowViewHistory() : User 의 시청 기록을 보여주는 method 이다. 각각의 영상 제목, 영상 ID, 업로드 날짜, 영상 길이, 영상을 본 시각을 보여준다. 이후 입력을 통해 deleteViewHistory method 를 이용해 시청 기록을 삭제할 수 있다.
    - 5. deleteViewHistory() : User 의 시청 기록을 삭제한다. 만약 user 의 시청 기록이 없다면, 관련 메세지를 출력한다. 성공적으로 기록을 삭제하면 관련 메세지를 출력한다.
  - ii. showVideo() : 관리자가 관리하는 사용자들이 올린 모든 영상의 목록을 출력한다. 영상의 제목, 영상 ID, 업로더 ID, 업로드 날짜, 영상 길이, 조회수, 좋아요 수, 신고 횟수, 나이 제한을 보여준다. 모든 영상의 정보 조회가 끝나면 입력을 통해 setAgeLimit method 를 통해 영상의 나이 제한을 설정할 수 있다.
    - 1. setAgeLimit() : 특정 영상의 나이 제한을 설정할 수 있다.
  - iii. showPlaylist() : 관리자가 관리하는 사용자들이 만든 영상 재생 리스트의 목록을 출력한다. 영상 재생 리스트의 제목, 리스트 ID, 제작자 ID, 신고 횟수를 보여준다. 그 후 영상 재생 리스트 안의 영상의 정보를 보여준다.
- e. delete() : Users 삭제, Videos 삭제, Playlists 삭제의 기능을 수행하는 method 를 실행시키는 method 이다.
  - i. deleteUser() : 관리자가 관리하는 사용자 중 신고 횟수가 100 이상인 사용자의 목록을 출력하고, 입력 받은 사용자 ID 를 DB 에서 찾아 이를 삭제한다.
  - ii. deleteVideo() : 관리자가 관리하는 사용자가 올린 영상 중 신고 횟수가 100 이상인 영상의 목록을 출력하고, 입력 받은 영상 ID 를 DB 에서 찾아 이를 삭제한다.

- iii. deletePlaylist() : 관리자가 관리하는 사용자가 만든 영상 재생 리스트 중 신고 횟수가 100 이상인 영상의 목록을 출력하고, 입력 받은 영상 재생 리스트 ID 를 DB 에서 찾아 이를 삭제한다.
- f. updateManager() : manager 의 비밀번호 변경, 이름 변경의 기능을 하는 method 를 실행하는 method 이다.
  - i. changePW() : manager 의 비밀번호를 변경한 후 이를 DB 에 반영한다. 10 자 이상의 비밀번호를 입력 받으면 오류 메시지를 출력하고 다시 새로운 비밀번호를 입력받는다.
  - ii. changeFname() : manager 의 first name 을 변경한 후 이를 DB 에 반영한다.
  - iii. changeLname() : manager 의 last name 을 변경한 후 이를 DB 에 반영한다.
- g. createManager() : 새로운 manager 를 생성한다. 10 자 이내의 ID, 비밀번호를 입력 받은 후 DB 에 중복된 이름의 manager 가 있는지 확인한 후 manager account 를 생성한다.
- h. deleteManager() : 'manager'를 제외한 모든 관리자의 목록을 보여준 후, 이 중 특정 manager 를 삭제한다. 이때, 삭제되는 manager 가 관리하는 Users, Videos, Playlists 의 관리자는 'manager'로 설정된다.

## 7. 프로그램 코드에 대한 설명 - User Class

- a. startUser() : 로그인, 사용자 계정 생성, 프로그램 종료 기능을 숫자로 입력받아 그 역할을 하는 method 로 연결시켜 주는 method 이다. 잘못된 입력이 들어오면 경고 메시지 출력 후 다시 숫자를 입력 받는다.
- b. loginUser() : user account 에 로그인 하는 기능을 구현한 method 이다. DB 에 입력 받은 ID,비밀번호와 일치하는 계정 정보가 있다면 그 계정으로 user 프로그램에 로그인한다. 만약 그렇지 않다면, ID 와 비밀번호를 다시 입력 받거나 startUser method 로 돌아간다.
- c. afterLogin() : Users, Videos, Playlists 목록을 보고 관리하는 기능, Users, Videos, Playlists 를 삭제하는 기능, manager 정보를 수정하는 기능, 프로그램 종료 기능을 각각의 숫자로 입력받아 그 역할을 하는 method 로 연결시켜 주는 method 이다. 잘못된 입력이 들어오면 경고 메시지 출력 후 다시 숫자를 입력 받는다.
- d. show() : 영상 목록을 보여주고 특정 영상을 시청하는 기능을 하는 method, 영상 재생 리스트의 목록을 보여주고 특정 영상 재생 리스트를 이용하는 기능을 하는 method 를 실행하는 method 이다. 이 method 에서는 이전 화면으로 돌아갈 수 있다.
- e. showVideo() : DB 에 업로드 된 모든 영상의 목록을 보여준다. 각 영상의 제목, 업로드 ID, 영상 길이, 조회수를 보여준다. 이때, 사용자의 나이가 어려 나이 제한에 걸리는 영상은 보이지 않는다. 영상의 목록을 모두 본 다음에는 watchVideo method 를 통해 특정 영상을 볼 수 있다.
  - i. watchVideo() : 영상의 제목과 업로더 ID 를 입력해 영상을 시청한다. 이때, 영상의 이름, 업로더 ID, 업로더의 구독자 수, 영상을 올린 날짜, 영상 길이, 조회수, 좋아요 수를 보여준다. 사용자의 나이로 인해 나이 제한에 걸리는 영상은 볼 수 없다. 영상을

- 시청하면 해당 영상의 조회수가 1 오르며, 사용자의 시청 기록에 해당 영상의 ID 와 영상 시청 시각이 저장된다. 영상 정보를 다 보여준 후에는 영상에 달린 댓글을 보여준다. 그 후 interactVideo method 를 통해 영상과 관련된 상호작용을 한다.
- ii. interactVideo(int, String, String) : 특정 영상에 좋아요를 누르는 기능, 업로드를 구독/구독 취소하는 기능, 댓글을 남기는 기능, 영상을 영상 재생 리스트에 담는 기능, 영상을 신고하는 기능, 업로더를 신고하는 기능을 하는 method 를 실행하는 method 이다.
  - iii. showComments(int).: 특정 영상에 달린 댓글의 내용과 댓글을 단 사용자의 ID 를 모두 보여준다.
  - iv. likeVideo(int) : 특정 영상에 좋아요를 남기는 기능을 하는 method 이다.
  - v. subscribe(String, boolean) : 특정 사용자를 구독하거나 구독 취소하는 기능을 구현한 method 이다. 특정 사용자를 구독하면 영상 업로더의 구독자가 1 늘어나고, SUBSCRIBES 테이블에 해당 관계가 저장된다. 특정 사용자를 구독 취소하면 영상 업로더의 구독자가 1 감소하고, SUBSCRIBES 테이블에 해당 관계가 삭제된다.
  - vi. leaveCommnet(int, String) : 특정 영상에 댓글을 남기는 기능을 하는 method 이다.
  - vii. putPlaylist(int) : showUserPlaylist method 를 통해 자신이 제작한 영상 재생 리스트의 목록을 보여준 뒤, 영상 재생 리스트의 이름을 입력 받아 그 리스트에 해당 영상을 넣는다. 이때, 재생 리스트에 영상을 추가하지 못하는 상황이면 에러 메시지를 출력한다.
  - viii. report(int, String) : 인자로 전달받은 ID 를 가진 영상 혹은 영상 재생 리스트를 신고하는 기능을 하는 method 이다. 인자로 전달받은 문자열이 “video”이면 ID 가 영상 ID 를 뜻하므로 특정 영상에 대한 신고를 수행한다. 인자로 전달받은 문자열이 “playlist”이면 ID 가 영상 재생 리스트 ID 를 뜻하므로 특정 영상 재생 리스트에 대한 신고를 수행한다.
  - ix. report(String) : 특정 User 를 신고하는 기능을 수행하는 method 이다. 만약 신고할 수 있는 사용자가 없다면, 에러 메시지를 출력한다.
- f. showPlaylist() : DB 에 업로드 된 모든 영상 재생 리스트의 목록을 출력한다. 출력 이후 watchPlaylist() method 를 통해 특정 영상 재생 리스트의 콘텐츠에 접근할 수 있다.
- i. watchPlaylist() : 사용자로부터 리스트 이름과 제작자 이름을 전달받은 후 selectOrder method 를 통해 목록 정렬 순서를 전달받는다. 이후 정렬 순서에 맞게 영상 제목, 업로더 ID, 영상 길이, 조회수, 나이 제한의 영상 정보를 보여준다.
  - ii. selectOrder() : 영상 재생 리스트 조회 시 영상의 정렬 순서를 정하는 method 이다. 이 method 를 통해 정렬 순서를 기본 순서, 최신 순, 오래된 순, 영상 길이가 긴 순, 조회수가 높은 순으로 정렬하여 영상의 목록을 조회할 수 있다.
  - iii. interactPlaylist(int) : 특정 영상 재생 리스트 안의 영상을 보거나, 영상 재생 리스트를 신고하는 기능을 수행하는 method 이다. 각각 watchVideo method, report method 를 통해 해당 기능을 수행한다.

- g. `manageChannel()` : 계정의 채널 정보 조회, 구독하는 채널 목록 조회, 영상 업로드, 업로드한 영상 삭제, 영상 업로드 목록 조회, 시청 기록 조회, 영상 재생 리스트 제작, 영상 재생 리스트 삭제, 만든 영상 재생 리스트 목록 조회의 기능을 하는 `method` 를 연결시켜주는 `method` 이다.
  - i. `showUserInfo()` : 계정 주인의 사용자 ID, 구독자 수, 나이를 보여준다.
  - ii. `showSubs()` : 계정 주인이 구독하고 있는 사용자의 ID 를 보여준다.
  - iii. `uploadVideo()` : 영상을 업로드하는 `method` 이다. 사용자로부터 영상 이름, 영상의 길이, 나이 제한을 입력 받아 이를 토대로 DB 에 영상을 업로드한다. 영상 업로드 시간은 자동으로 설정된다. 만약 이미 DB 에 중복되는 영상이 존재한다면 에러 메시지를 출력한다.
  - iv. `deleteVideo()` : `showUploadList` `method` 를 통해 자신이 업로드한 영상의 목록을 보고, 이 중 업로드한 영상을 삭제하는 기능을 하는 `method` 이다.
  - v. `showUploadList()` : 자신이 업로드한 영상들의 정보를 보여주는 기능을 하는 `method` 이다. 영상의 제목, 업로드 날짜, 영상 길이, 조회수, 좋아요 수, 나이 제한을 보여준다. 만약 사용자가 업로드한 영상이 있다면 `true` 를 `return` 한다.
  - vi. `showViewHistory()` : 사용자의 시청 기록을 보여주는 기능을 하는 `method` 이다. 영상 제목, 업로더 ID, 영상 시청 시각을 보여준다.
  - vii. `makePlaylist()` : 영상 재생 리스트를 제작하는 `method` 이다. 사용자로부터 제작할 영상 재생 리스트의 제목을 전달받아 이를 제작한 후 DB 에 반영한다. 만약 중복되는 영상 재생 리스트가 있다면 에러 메시지를 출력한다.
  - viii. `deletePlaylist()` : 사용자가 만든 영상 재생 리스트 중 하나를 삭제하는 기능을 하는 `method` 이다.
  - ix. `showUserPlaylist()` : 사용자가 만든 영상 재생 리스트의 이름을 목록으로 보여주는 기능을 하는 `method` 이다. 만약 사용자가 만든 영상 재생 리스트가 있다면 `true` 를 `return` 한다.
- h. `updateUser()` : 사용자 계정의 비밀번호를 변경하거나, 계정을 삭제하는 기능을 하는 `method` 와 연결시켜주는 역할을 하는 `method` 이다.
  - i. `changePW()` : 사용자 계정의 새로운 비밀번호를 입력 받고 이를 DB 에 반영하는 `method` 이다. 10 자 이상의 비밀번호를 입력 받으면 오류 메시지를 출력하고 다시 새로운 비밀번호를 입력 받는다.
  - ii. `deleteUser()` : 사용자의 계정을 삭제하는 `method` 이다. 계정 삭제에 대한 의사를 사용자에게 다시 물어본 뒤, 계정 삭제의 의사가 있다면 계정을 삭제한 후 이를 DB 에 반영한다. 계정 삭제 후 `startUser` `method` 를 실행해 초기 화면으로 돌아간다.
- i. `createUser()` : 새로운 사용자 계정을 생성하는 `method` 이다. 사용자로부터 ID, 비밀번호, 나이를 입력 받아 계정을 생성한다. 만약 같은 이름의 ID 가 이미 DB 에 존재한다면, 오류 메시지를 출력하고 사용자로부터 새로운 ID 를 전달받는다. 계정 생성에 성공하면 `startUser` `method` 를 실행해 초기 화면으로 돌아간다.

8. 수행되는 기능 스크린샷 - 프로그램의 기능이 많은 관계로 기능의 일부만 예시로 실행하였다.

- a. 개발 및 테스트 환경 : MacBook Air M1, terminal 에서 실행
- b. 수행되는 기능 - Manager
  - i. java -jar Manager.jar 를 통해 프로그램을 실행한다.

```
~ $ java -jar Manager.jar
----- Manager Program -----
1. Login
2. Create new manager
3. Delete manager
4. Quit program
Please enter a number to select a mode : |
```

- ii. 로그인 후 User 목록을 조회하는 예시

```
----- Show & Manage Users | Videos | Playlists -----
1. Show & Manage Users
2. Show & Manage Videos
3. Show Playlists
4. Return to previous menu
Please enter a number to select a mode : 1

----- Show & Manage User List -----
User ID : 2021025205, Age : 10, subNum : 0, Reports : 0
User ID : user1, Age : 23, subNum : 0, Reports : 100
```

- iii. user1 의 업로드 목록을 조회하는 예시

```
----- Show User's Upload List -----
Please enter user's ID : user1
## user1's Upload List ##
Video Title : qooo, Video ID : 4, Upload Date : 2022-11-30, Video Length : 00:14:05, Views : 0, Likes : 0, Reports : 0, Age Limit : 0
Video Title : video1, Video ID : 3, Upload Date : 2022-11-30, Video Length : 00:13:12, Views : 0, Likes : 0, Reports : 0, Age Limit : 15
Video Title : Hello, Video ID : 5, Upload Date : 2012-11-05, Video Length : 00:00:54, Views : 0, Likes : 0, Reports : 0, Age Limit : 0
Video Title : Me at the zoo, Video ID : 1, Upload Date : 2008-02-04, Video Length : 00:05:05, Views : 0, Likes : 0, Reports : 100, Age Limit : 19
----- You cannot delete a video at this screen. If you want to delete the video, go to 'delete' page -----
```

- iv. user1 의 시청 기록을 조회하고 삭제한 예시

```
----- Show & Manage User's View History -----
Please enter user's ID : user1
## user1's View History ##
Video Title : Hello, Video ID : 2, Upload Date : 2009-06-09, Video Length : 00:00:08, Watch at 2022-11-30 15:50:47
Video Title : Hello, Video ID : 2, Upload Date : 2009-06-09, Video Length : 00:00:08, Watch at 2014-06-16 14:35:57
----- Done -----

----- Manage View History -----
1. Delete user's view history
Any keys except 1. Do nothing
Enter a number for the following task : 1
----- Deletion Complete -----

## User Management ##
1. Change a user's manager
2. Show user's upload list
3. Show user's view history
4. Do nothing
Enter a number for the following task : |
```

- v. 2021025205 의 관리자를 manager2 로 변경한 예시

```
----- Transfer User Management -----
Please enter user's ID : 2021025205
Please enter the ID of the user's new manager : manager2
```



vi. 관리자가 관리하는 영상의 목록을 보여주는 예시

```
----- Show Video's Information List -----  
Video Title : Hello, Video ID : 5, Uploader ID : user1, Upload Date : 2012-11-05, Video Length : 00:00:54, Views : 0, Likes : 0,  
Video Title : Me at the zoo, Video ID : 1, Uploader ID : user1, Upload Date : 2008-02-04, Video Length : 00:05:05, Views : 0, Lik  
imit : 19  
Video Title : qooo, Video ID : 4, Uploader ID : user1, Upload Date : 2022-11-30, Video Length : 00:14:05, Views : 0, Likes : 0, R  
Video Title : video1, Video ID : 3, Uploader ID : user1, Upload Date : 2022-11-30, Video Length : 00:13:12, Views : 0, Likes : 0,
```

vii. Hello 영상의 나이 제한을 설정하는 예시

```
----- Manage Videos -----  
1. Set age limit  
Any keys except 1. Do nothing  
Enter a number for the following task : 1  
Please enter video's ID you want to manage : 5  
Please enter an age limit of this video : 10  
----- Change 5's age limit to 10 -----
```

viii. 관리자가 관리하는 재생 리스트의 목록을 보여주는 예시

```
----- Show Playlist's Information List -----  
List Title : playlist, List ID : 1, Maker ID : user1, Report : 100  
-----  
Video Title : Me at the zoo, Video ID : 1, Uploader : user1, Upload Date : 2008-02-04, Video Length : 00:05:05, Views : 0, Likes : 0, Age Limit : 19  
Video Title : Hello, Video ID : 2, Uploader : user2, Upload Date : 2009-06-09, Video Length : 00:00:08, Views : 0, Likes : 0, Age Limit : 0  
-----
```

ix. 신고가 100 이상인 영상을 삭제하는 예시

```
----- Delete Videos -----  
Video Title : Me at the zoo, Video ID : 1, Uploader ID : user1, Report : 100  
Enter the ID of the Video you want to delete : 1  
## 1 Deleted ##
```

x. 신고가 100 이상인 재생 리스트를 삭제하는 예시

```
----- Delete Playlists -----  
List Title : playlist, List ID : 1, Maker ID : user1, Report : 100  
Enter the ID of the Playlist you want to delete : 1  
## 1 Deleted ##
```

xi. 신고가 100 이상인 사용자를 삭제하는 예시

```
----- Type the ID of User you want to delete -----  
User ID : user1, Report : 100  
Enter the ID of the User you want to delete : user1  
## user1 Deleted ##
```

xii. 관리자 계정을 생성하는 예시

```
----- Create New Manager -----
Please enter new manager's ID (Max length = 10) : manager10
Please enter new manager's PW (Max length = 10) : aoslwj10
Please enter new manager's first name : Tom
Please enter new manager's last name : Tom
----- Creation Success! Return to Main -----

----- Manager Program -----
1. Login
2. Create new manager
3. Delete manager
4. Quit program
Please enter a number to select a mode : 1
Please enter your manager ID : manager10
Please enter your manager Password : aoslwj10

----- Manager Menu - Manager : manager10 -----
1. Show & Manage Users | Videos | Playlists
2. Delete Users | Videos | Playlists
3. Update manager information
4. Quit program
Please enter a number to select a mode :
```

xiii. 관리자 계정을 삭제하는 예시

```
----- Delete Manager -----
1. manager1
2. manager10
3. manager2
4. manager3
5. manager4
6. manager5
Please enter a manager ID you want to delete : manager1
## manager1 Deleted ##

----- Manager Program -----
1. Login
2. Create new manager
3. Delete manager
4. Quit program
Please enter a number to select a mode : 1
Please enter your manager ID : manager1
Please enter your manager Password : aoslwj1

----- Login Failed -----
1. Try again
2. Return to Main
All numbers except 1, 2. Quit Program
Please enter a number :
```

c. 수행되는 기능 - User (DB 재생성 이후 기능 시현을 진행했다.)

i. java -jar User.jar 를 통해 프로그램을 실행한다.

```
~ $ java -jar User.jar
----- User Program -----
1. Login
2. Sign Up
3. Quit program
```

ii. 영상 목록 조회, 영상 시청 기능 예시

```
----- Show Videos -----
Video Title : Me at the zoo, Uploader ID : user1, Video Length : 00:05:05, Views : 0, Age Limit : 19
Video Title : Hello, Uploader ID : user2, Video Length : 00:00:08, Views : 0, Age Limit : 0
Video Title : video1, Uploader ID : user1, Video Length : 00:13:12, Views : 0, Age Limit : 15
Video Title : qooo, Uploader ID : user1, Video Length : 00:14:05, Views : 0, Age Limit : 0
Video Title : Hello, Uploader ID : user1, Video Length : 00:00:54, Views : 0, Age Limit : 0

*** If you want to watch a video, type '1'. If not, type something else. ***
Please enter a key to select a mode : 1

## Wacth a Video ##
Please enter a name of video you want to watch : Hello
Please enter a uploader ID of video you want to watch : user2

----- Hello -----
Uploader ID : user2, Subscribers : 0, Upload Date : 2009-06-09, Length : 00:00:08, Views : 0, Likes : 0
----- Comments -----
user1 : WOWOWOWOW
-----
```

iii. 영상 좋아요, 업로더 구독 예시

```
## Interaction ##
1. Like this video
2. Subscribe uploader
3. Leave a comment
4. Put this video in a playlist
5. Report this video
6. Report uploader
7. Return to previous menu
Please enter a number to select a mode : 1
---- Like! ----

## Interaction ##
1. Like this video
2. Subscribe uploader
3. Leave a comment
4. Put this video in a playlist
5. Report this video
6. Report uploader
7. Return to previous menu
Please enter a number to select a mode : 2
----- Subscribe user1 -----
```

iv. 댓글 남기기 기능

```
Please leave a comment : gooooooooood
---- You leaved a comment! ----
user1 : gooooooooood
```

v. 기능 실행 결과

```
----- Hello -----
Uploader ID : user2, Subscribers : 1, Upload Date : 2009-06-09, Length : 00:00:08, Views : 1, Likes : 1
----- Comments -----
user1 : wowwowwow
user1 : gooooooooood
-----
```

vi. 조회수 순으로 정렬하여 영상 재생 리스트 보는 기능

```
----- Show PlayLists -----
List Title : my list, Maker ID : hyu
List Title : playlist, Maker ID : user1

## Watch a Playlist ##
Please enter a name of playlist you want to watch : playlist
Please enter a maker ID of playlist you want to watch : user1

----- Select sort order -----
1. Default order
2. Upload Date (most recent)
3. Upload Date (oldest)
4. Video length (longest)
5. Most popular
Please select the number of sort order : 5

----- playlist -----
Video Title : Hello, Uploader ID : user2, Video Length : 00:00:08, Views : 2, Age Limit : 0
Video Title : Me at the zoo, Uploader ID : user1, Video Length : 00:05:05, Views : 0, Age Limit : 19

*** Interact Playlist ***
1. Watch a video
2. Report a playlist
3. Return to previous menu
Please enter a key to select a mode : |
```

vii. 채널 정보 보여주는 기능

```
----- Show Channel Info -----
User ID : user1, Subs Number : 2, Age : 23
```

viii. 구독하고 있는 채널 보여주는 기능

```
----- List of Subscribing Channels -----
2021025205
hyu
user2
```

ix. 영상 업로드 기능

```
----- Upload a Video -----
Please enter a video name : goood
Please enter a video's length (yy:mm:ss) : 00:13:13
Please enter an age limit of the video : 0
----- Upload Complete! -----
```

x. 업로드 한 영상 조회 기능

```
## user1's Upload List ##
Video Title : goood, Upload Date : 2022-12-10, Video Length : 00:13:13, Views : 0, Likes : 0, Age Limit : 0
Video Title : qooo, Upload Date : 2022-11-30, Video Length : 00:14:05, Views : 0, Likes : 0, Age Limit : 0
Video Title : video1, Upload Date : 2022-11-30, Video Length : 00:13:12, Views : 0, Likes : 0, Age Limit : 15
Video Title : Hello, Upload Date : 2012-11-05, Video Length : 00:00:54, Views : 0, Likes : 0, Age Limit : 0
Video Title : Me at the zoo, Upload Date : 2008-02-04, Video Length : 00:05:05, Views : 1, Likes : 0, Age Limit : 19
```

xi. 제작한 영상 재생 리스트 조회 기능

```
## user1's Playlist List ##  
Playlist Title : playlist
```

xii. 계정 삭제 기능

```
----- Delete Your Account -----  
IF YOU WANT TO DELETE THIS ACCOUNT FOR SURE, TYPE 'Y'  
Y  
----- user1 Deleted. Return to initial menu -----  
  
----- User Program -----  
1. Login  
2. Sign Up  
3. Quit program  
Please enter a number to select a mode : 1  
Please enter your ID : user1  
Please enter your Password : dbwj1  
  
----- Login Failed -----  
1. Try again  
2. Return to Main  
All numbers except 1, 2. Quit Program  
Please enter a number : |
```

xiii. 계정 생성 기능

```
----- Sign Up -----  
Please enter new user's ID (Max length = 10) : user10  
Please enter new user's PW (Max Length = 10) : dbwj10  
Please enter new user's age : 15  
----- Creation Success! Return to Main -----  
  
----- User Program -----  
1. Login  
2. Sign Up  
3. Quit program  
Please enter a number to select a mode : 1  
Please enter your ID : user10  
Please enter your Password : dbwj10  
  
----- User Menu - User : user10 -----  
1. Show & Watch Videos | Playlists  
2. Manage Your Channel  
3. Update User Information  
4. Quit program  
Please enter a number to select a mode :
```

## 9. 사용되는 SQL 문 명세 - Manager

### a. loginManager method

#### i. SELECT managerID, managerPW from MANAGER;

1. MANAGER 테이블에서 매니저 계정의 ID 와 매니저 계정의 비밀번호를 가져온다.
2. 매니저 계정 로그인 시 입력과 일치하는 계정을 찾기 위해 사용된다.

### b. showUserList method

#### i. SELECT userID, age, subscribedNum, userReportNum

FROM MANAGER, USER

WHERE MANAGER.managerID = USER.managerID AND

MANAGER.managerID = '(managerID)';

1. 관리자가 관리하는 사용자들의 정보를 가져오기 위해 사용된다.
2. 사용자의 ID, 나이, 구독자 수, 신고 수를 가져온다.

### c. transferUserManagement method

#### i. UPDATE USER SET managerID = '(managerID)'

WHERE userID = '(userID)' AND managerID = '(managerID)';

1. 키보드를 통해 입력 받은 (userID) 사용자의 관리자를 입력 받은 (managerID) 관리자로 변경하는 쿼리이다.

### d. showUploadList method

#### i. SELECT videoTitle, videoID, uploadDate, videoLength, views, likes,

vReportNum, ageLimit FROM USER, VIDEO

WHERE USER.managerID = '(managerID)' AND userID = '(userID)' AND userID = uploaderID ORDER BY uploadDate DESC;

1. 자신이 관리하는 사용자 중 사용자 ID 가 (userID) 사용자인 사용자의 업로드한 영상들을 최근에 올린 영상 순으로 정렬하여 가져온다.
2. 영상 제목, 영상 ID, 업로드 일자, 영상 길이, 조회수, 좋아요 수, 신고 수, 나이 제한 정보를 가져온다.

### e. showViewHistory method

#### i. SELECT videoTitle, videoID, uploadDate, videoLength, watchStartTime

FROM USER, WATCH, VIDEO

WHERE USER.managerID = '(managerID)' AND USER.userID = '(userID)' AND

USER.userID = WATCH.userID AND watchVideoID = videoID

ORDER BY watchStartTime DESC;

1. 자신이 관리하는 사용자 중 사용자 ID 가 (userID)인 사용자의 시청한 영상 목록을 영상 시청 시작 시점을 기준으로 내림차순 정렬하여 가져온다.
2. 영상 제목, 영상 ID, 업로드 날짜, 영상 길이, 영상 시청 시작 시점에 대한 정보를 가져온다.

- f. deleteViewHistory method
  - i. DELETE FROM WATCH WHERE userID = '(userID)';
    - 1. 사용자 ID 가 (userID)인 사용자의 시청한 영상 엔티티를 모두 삭제한다.
- g. showVideo method
  - i. SELECT videoTitle, videoID, uploaderID, uploadDate, videoLength, views, likes, vReportNum, ageLimit FROM VIDEO, USER WHERE managerID = '(managerID)' AND userID = uploaderID;
    - 1. 사용자의 관리자 ID 가 (managerID)인 사용자 중 (userID) 사용자가 업로드한 영상 목록을 가져온다.
    - 2. 영상 제목, 영상 ID, 업로더 ID, 업로드 날짜, 영상 길이, 조회수, 좋아요 수, 신고 수, 나이 제한에 대한 정보를 가져온다.
- h. setAgeLimit method
  - i. UPDATE VIDEO SET ageLimit = '(limit)' WHERE videoID = (videoID) AND uploaderID IN (SELECT userID FROM USER WHERE managerID = '(managerID)');
    - 1. 입력 받은 영상 ID 가 (videoID)인 영상의 나이 제한을 입력 받은 (limit)으로 설정한다.
- i. showPlaylist method
  - i. SELECT listTitle, listID, makerID, listReportNum FROM PLAYLIST, USER WHERE managerID = '(managerID)' AND makerID = userID;
    - 1. 사용자의 관리자 ID 가 (managerID)인 사용자가 제작한 영상 재생 리스트의 정보를 가져온다.
    - 2. 리스트 제목, 리스트 ID, 제작자 ID, 신고 수를 가져온다.
- j. showListVideos method
  - i. SELECT videoTitle, videoID, uploaderID, uploadDate, videoLength, views, likes, ageLimit FROM VIDEO, CONSIST\_OF WHERE cListID = (listID) AND cVideoID = videoID;
    - 1. (listID) 영상 재생 리스트를 구성하는 영상들의 목록을 가져온다.
    - 2. 각 영상의 제목, 영상 ID, 업로더 ID, 업로드 날짜, 영상 길이, 조회수, 좋아요 수, 나이 제한 정보를 가져온다.
- k. deleteUser method
  - i. SELECT userID, userReportNum FROM MANAGER, USER WHERE MANAGER.managerID = '(managerID)' AND MANAGER.managerID = USER.managerID AND userReportNum >= 100;
    - 1. 관리자가 관리하는 사용자 중 신고 수가 100 이상인 사용자의 ID, 신고 수 목록을 가져온다.

- ii. DELETE FROM USER  
WHERE managerID = '(managerID)' AND userID = '(deleteUserID)';  
1. 사용자 중 (deleteUserID)의 사용자 ID 를 가진 사용자를 삭제한다.
- l. deleteVideo method
  - i. SELECT videoTitle, videoID, uploaderID, vReportNum FROM VIDEO, USER  
WHERE managerID = '(managerID)' AND userID = uploaderID AND  
vReportNum >= 100;  
1. 관리자가 관리하는 영상 중 신고가 100 개 이상인 영상의 목록을 가져온다.  
2. 영상 제목, 영상 ID, 업로더 ID, 신고 수를 가져온다.
  - ii. DELETE FROM VIDEO WHERE videoID = (deleteVideoID) AND uploaderID IN  
(SELECT userID FROM USER WHERE managerID = '(managerID)') AND  
vReportNum >= 100;  
1. 관리자가 관리하는 영상 중 영상 ID 가 (deleteVideoID)인 영상을 삭제한다.
- m. deletePlaylist method
  - i. SELECT listTitle, listID, makerID, listReportNum  
FROM MANAGER, USER, PLAYLIST  
WHERE MANAGER.managerID = '(managerID)' AND MANAGER.managerID =  
USER.managerID AND userID = makerID AND listReportNum >= 100;  
1. 관리자가 관리하는 영상 중 신고가 100 개 이상인 영상 재생 리스트의 목록을  
가져온다.  
2. 리스트 제목, 리스트 ID, 제작자 ID, 신고 수 정보를 가져온다.
  - ii. DELETE FROM PLAYLIST WHERE listID = (deletePlaylistID) AND makerID In  
(SELECT userID FROM USER WHERE managerID = '(managerID)') AND  
listReportNum >= 100;  
1. 관리자가 관리하는 영상 재생 리스트 중 리스트 ID 가 (deletePlaylistID)인  
영상 재생 리스트를 삭제한다.
- n. changePW method
  - i. UPDATE MANAGER SET managerPW = '(managerPW)'  
WHERE managerID = '(managerID)';  
1. 관리자 ID 가 (managerID)인 관리자의 계정 비밀번호를 (managerPW)로  
설정한다.
- o. changeFname method
  - i. UPDATE MANAGER SET Fname = '(Fname)'  
WHERE managerID = '(managerID)';  
1. 관리자 ID 가 (managerID)인 관리자의 first name 을 (Fname)로 설정한다.



- p. changeLname method
  - i. UPDATE MANAGER SET Lname = '(Lname)'  
WHERE managerID = '(managerID)';
    - 1. 관리자 ID 가 (managerID)인 관리자의 last name 을 (Lname)로 설정한다.
- q. createManager method
  - i. SELECT managerID FROM MANAGER;
    - 1. 모든 관리자의 ID 를 가져온다.
  - ii. INSERT INTO MANAGER(managerID, Fname, Lname, managerPW)  
VALUES ('(managerID)', '(Fname)', '(Lname)', '(managerPW)');
    - 1. 새로운 관리자를 MANAGER 테이블에 등록한다.
- r. deleteManager method
  - i. SELECT managerID FROM MANAGER;
    - 1. 모든 관리자의 ID 를 가져온다.
  - ii. UPDATE USER SET managerID = 'manager' WHERE managerID =  
'(dManagerID)';
    - 1. 삭제하고자 하는 관리자의 사용자의 관리자를 'manager'로 변경한다.
  - iii. DELETE FROM MANAGER WHERE managerID = '(dManagerID)';
    - 1. (dManagerID) 관리자를 MANAGER 테이블에서 삭제한다.

## 사용되는 SQL 명세 - User

- a. loginUser method
  - iv. `SELECT userID, userPW, age FROM USER;`
    - 1. 모든 사용자의 ID, 비밀번호, 나이를 불러온다.
- b. showVideo method
  - v. `SELECT videoTitle, uploaderID, videoLength, views, ageLimit FROM VIDEO, USER WHERE ageLimit <= age AND userID = '(userID)';`
    - 1. 사용자 ID 가 (userID)인 사용자에게 사용자의 나이 제한에 걸리지 않는 영상 목록을 보여준다.
    - 2. 영상 제목, 업로더 ID, 영상 길이, 조회수, 나이 제한을 가져온다.
- c. watchVideo method
  - vi. `SELECT ageLimit, videoID, videoTitle, uploaderID, subscribedNum, uploadDate, videoLength, views, likes FROM VIDEO, USER WHERE videoTitle = '(videoTitle)' AND uploaderID = '(uploaderID)' AND uploaderID = userID;`
    - 1. (uploaderID) 사용자가 올린 (videoTitle) 제목의 영상에 대한 정보를 불러온다.
  - vii. `UPDATE VIDEO SET views = views + 1 WHERE videoID = (videoID);`
    - 1. (videoID) 영상의 조회수를 1 늘린다.
  - viii. `INSERT INTO WATCH VALUES ('(userID)', (videoID), '(time)', NULL);`
    - 1. (userID) 사용자가 (videoID) 영상을 (time) 시점에 시청했음을 알리는 튜플을 WATCH 테이블에 추가한다.
- d. showComments method
  - ix. `SELECT userID, comment FROM WATCH WHERE watchVideoID = (videoID);`
    - 1. (videoID) 영상의 댓글과 댓글을 단 사용자의 ID 를 가져온다.
- e. interactVideo method
  - x. `SELECT * FROM SUBSCRIBES WHERE userID = '(userID)' AND subUserID = '(uploaderID)';`
    - 1. (userID)가 (subUserID) 이용자를 구독하고 있는지 여부를 확인하는데 사용되는 쿼리이다.
- f. likeVideo method
  - xi. `UPDATE VIDEO SET likes = likes + 1 WHERE videoID = (videoID);`
    - 1. (videoID) 영상의 좋아요 수를 1 늘린다.

- g. subscribe method
  - xii. UPDATE USER SET subscribedNum = subscribedNum + 1  
WHERE userID = '(uploaderID)';  
1. (uploaderID) 사용자의 구독자 수를 1 늘린다.
  - xiii. INSERT INTO SUBSCRIBES VALUES ('(userID)', '(uploaderID)');  
1. SUBSCRIBES 테이블에 구독 관계 튜플을 추가한다.
  - xiv. UPDATE USER SET subscribedNum = subscribedNum - 1  
WHERE userID = '(uploaderID)';  
1. (uploaderID) 사용자의 구독자 수를 1 줄인다.
  - xv. DELETE FROM SUBSCRIBES  
WHERE userID = '(userID)' AND subUserID = '(uploaderID)';  
1. SUBSCRIBES 테이블에서 해당 튜플을 삭제한다.
- h. leaveComment method
  - xvi. UPDATE WATCH SET comment = '(comment)' WHERE userID = '(userID)' AND  
watchVideoID = (videoID) AND watchStartTime = '(time)';  
1. 영상을 시청할 때 해당 영상에 댓글 (comment)를 남기는 쿼리이다.
- i. putPlaylist method
  - xvii. SELECT listID FROM PLAYLIST WHERE listTitle = '(listN)' AND makerID =  
'(userID)';  
1. 사용자 자신이 만든 영상 재생 리스트의 ID 를 가져온다.
  - xviii. INSERT INTO CONSIST\_OF VALUES ((listID), (videoID));  
1. (listID) 재생 리스트에 (videoID) 영상을 넣는다.
- j. report method
  - xix. UPDATE VIDEO SET vReportNum = vReportNum + 1 WHERE videoID = (ID);  
1. (ID) 영상의 신고 수를 1 늘린다.
  - xx. UPDATE PLAYLIST SET listReportNum = listReportNum + 1  
WHERE listID = (ID);  
1. (ID) 영상 재생 리스트의 신고 수를 1 늘린다.
  - xxi. UPDATE UER SET userReportNum = userReportNum + 1  
WHERE userID = '(userID)';  
1. (ID) 유저의 신고 수를 1 늘린다.
- k. showPlaylist method
  - xxii. SELECT listTitle, makerID FROM PLAYLIST;  
1. 모든 영상 재생 리스트의 제목, 제작자 ID 를 가져온다.
- l. watchPlaylist method
  - xxiii. SELECT listID, videoTitle, uploaderID, videoLength, views, ageLimit  
FROM USER, VIDEO, CONSIST\_OF, PLAYLIST

WHERE ageLimit <= age AND userID = '(userID)' AND makerID = '(makerID)'  
AND listTitle = '(name)' AND listID = cListID = AND cVideoID = videoID;

1. 영상 재생 리스트를 구성하는 영상 중 사용자의 나이 제한에 걸리지 않는 영상을 제외한 모든 영상의 목록을 가져온다.

m. selectOrder method

xxiv. ORDER BY uploadDate DESC;

1. 영상 재생 리스트의 영상 순서를 최신 순으로 정렬하는데 사용되는 쿼리의 부분이다.

xxv. ORDER BY uploadDate ASC;

1. 영상 재생 리스트의 영상 순서를 오래된 순으로 정렬하는데 사용되는 쿼리의 부분이다.

xxvi. ORDER BY videoLength DESC;

1. 영상 재생 리스트의 영상 순서를 영상 길이가 긴 순으로 정렬하는데 사용되는 쿼리의 부분이다.

xxvii. ORDER BY views DESC;

1. 영상 재생 리스트의 영상 순서를 조회수가 높은 순으로 정렬하는데 사용되는 쿼리의 부분이다.

n. showUserInfo method

xxviii. SELECT userID, subscribedNum, age FROM USER WHERE userID = '(userID)';

1. (userID) 사용자의 ID, 구독자 수, 나이 정보를 가져온다.

o. showSubs method

xxix. SELECT subUserID FROM SUBSCRIBES WHERE userID = '(userID)';

1. (userID) 사용자가 구독하는 사용자의 ID 의 목록을 가져온다.

p. uploadVideo method

xxx. INSERT INTO VIDEO ('uploaderID', 'videoTitle', 'uploadDate', 'videoLength',  
'views', 'likes', 'vReportNum', 'ageLimit')  
VALUES '(userID)', '(videoName)', '(uploadDate)', '(videoLen)', 0, 0, 0,  
(ageLimit));

1. VIDEO 테이블에 입력 받은 영상 정보를 바탕으로 영상을 업로드하는데 사용되는 쿼리이다.

q. deleteVideo method

xxxi. DELETE FROM VIDEO WHERE videoTitle = '(videoTitle)' AND uploaderID =  
'(userID)';

1. VIDEO 테이블에서 영상 제목이 (videoTitle)이고 업로더의 ID 가 (userID)인 영상을 삭제한다.

- r. showUploadList method
  - xxxii. SELECT videoTitle, uploadDate, videoLength, views, likes, ageLimit  
FROM VIDEO WHERE uploaderID = '(userID)' ORDER BY uploadDate DESC;
    - 1. 영상 업로더가 (userID)인 영상의 목록을 최신순으로 정렬하여 가져온다.
- s. showViewHistory method
  - xxxiii. SELECT videoTitle, uploaderID, watchStartTime FROM WATCH, VIDEO  
WHERE userID = '(userID)' AND watchVideoID = videoID  
ORDER BY watchStartTime DESC;
    - 1. (userID) 사용자가 시청한 영상의 목록을 최근에 본 영상 순으로 정렬하여 가져온다.
- t. makePlaylist method
  - xxxiv. INSERT INTO `PLAYLIST` (`makerID`, `listTitle`, `listReportNum`)  
VALUES ('(userID)', '(listName)', 0);
    - 1. PLAYLIST 테이블에 입력 받은 영상 재생 리스트의 정보를 바탕으로 영상 재생 리스트를 만드는데 사용되는 쿼리이다.
- u. deletePlaylist method
  - xxxv. DELETE FROM PLAYLIST WHERE listTitle = '(listTitle)' AND makerID = '(userID)';
    - 1. PLAYLIST 테이블에서 리스트 제목이 (listTitle)이고 제작자 ID 가 (userID)인 영상 재생 리스트를 삭제한다.
- v. showUserPlaylist method
  - xxxvi. SELECT listTitle FROM PLAYLIST WHERE makerID = '(userID)';
    - 1. 리스트 제작자 ID 가 (userID)인 영상 재생 리스트의 영상 제목들을 불러온다.
- w. changePW method
  - xxxvii. UPDATE USER SET userPW = '(userPW)' WHERE userID = '(userID)';
    - 1. (userID) 사용자의 계정 비밀번호를 (userPW)로 변경한다.
- x. deleteUser method
  - xxxviii. DELETE FROM USER WHERE userID = '(userID)';
    - 1. (userID) 사용자의 계정을 삭제한다.
- y. createUser method
  - xxxix. SELECT userID FROM USER;
    - 1. USER 테이블에서 모든 userID 목록을 가져온다.
  - xl. SELECT managerID FROM MANAGER ORDER BY RAND() LIMIT 1;
    - 1. 관리자들의 ID 를 랜덤으로 하나 가져온다.
  - xli. INSERT INTO USER VALUES ('(managerID)', '(userID)', '(userPw)', 0, 0, age);
    - 1. 입력받은 정보를 바탕으로 새로운 사용자 계정 정보를 USER 테이블에 삽입해 새로운 사용자를 만든다.