

1. 실행 명령어

JAVA 환경 : jdk-11.0.6

```
$javac MusicPlayer.java
```

```
$java -classpath .;mariadb-java-client-2.7.1.jar MusicPlayer
```

MariaDB 환경 : MariaDB 10.5

```
$Mysql -u root -p musicplayer < musicplayer.sql
```

2. DB 설명

User id, music id, artist id, album id는 모두 auto_increment 만약 삭제 되었을시 키 초기화는 안해준다. - 학생이 졸업을 한다고 그 학번을 그 다음 학번이 가지는것이 아니기 때문에 이것과 같은상황이라고 판단하였다.

3. 수정사항

출시하다, 제작하다, 수록되다 모두 partial - 순차적으로 만들수 있어야 하므로 partial이 되도록 하였다.

플레이하다 - 플레이 횟수 추가(user마다 관리). 플레이 횟수를 여러 명이 한음원을 듣는 것을 합산해야 하므로 플레이 횟수를 추가하였다.

관리하다 - 테이블이 아닌 user의 attribute로 추가함. 관리권한이 있

는 사람은 모든 음원에 접근 가능하므로 테이블이 필요없다고 판단.

사용자 – manage(boolean) 추가, email 추가(로그인을 위해. 아이디는 auto_increment로 지정하여서 로그인을 위해 email을 추가하였다.)

4. 실행 과정

1) 데이터 베이스 접속

2) 첫화면

1. 로그인 login()
2. 회원 가입 signUp()

3) login 후 main menu

1. 음원, 앨범, 아티스트, 목록 보기 showAll()
2. 플레이리스트 메뉴 playList()
3. 음원 차트 musicChart()
4. 검색 search()
5. 관리 manage()

4) playMusic()

사용자가 노래를 플레이

3-1) showAll()

1. 음악 목록 showMusics()
2. 앨범 목록 showAlbums()
3. 아티스트 목록 showArtists()

3-1-1) showMusics()

각 음악 선택 시 musicDetails()

3-1-1-1) musicDetails()

이름, 소속앨범, 장르, 출시일, 추천수, 가사, 내 재생수

1. 추천 Recommendation (추천 또는 추천 취소)
2. 댓글 보기 Show comments
3. 댓글 쓰기 Write comment
4. 플레이리스트에 추가 Add to playlist
5. 전체 가사 보기 Show full lyrics

3-1-2) showAlbums()

각 앨범 선택 시 albumDetails()

3-1-2-1) albumDetails

이름, 출시일, 장르, 좋아요 수, 수록곡

1. 좋아요 like (좋아요 또는 좋아요 취소)
2. play album 시 앨범수록곡을 모두 playMusic()

3-1-3) showArtists()

각 아티스트 선택 시 아티스트의 노래리스트 보여줌

3-2) playlist()

1. 내 재생목록 showPlaylist()
2. 재생목록 만들기 Add playlist
3. 재생목록 삭제하기 Delete playlist
4. public한 모든 재생목록 보기 Show all playlist

3-2-1) showPlaylist()

각 플레이리스트 선택 시 플레이리스트의 노래들을 보여줌

1. 플레이리스트 재생 playPlaylist()
2. 원하는 음원 삭제 Delete music

3-2-1-1) playPlaylist()

1. 플레이리스트에 있는 모든 노래들을 재생 playMusic()

3-3) musicChart()

순위, 제목, 총재생횟수0

음원의 재생수로 1위부터 100위까지 순위를 정함.

재생수 0회 음원은 제외

3-4) search()

입력하는 단어가 있는 음원, 앨범, 아티스트 목록을 출력한다.

3-5) manage()

1. 음원 관리 manageMusic()
2. 유저 관리 manageUsers()
3. 앨범 관리 manageAlbum()
4. 아티스트관리 manageArtist()

3-5-1) manageMusic()

1. 음원 추가 Add music

음원 추가, 음원의 아티스트, 음원이 속한 앨범 연결

가사, 작사가, 작곡가는 StringBuilder 사용.

-end가 나올때까지 정보를 받음.

2. 음원 삭제 Delete music
3. 음원 수정 Update music

음원 제목, 가사 수정

3-5-2) manageUser()

유저 목록 나타냄

각 유저 선택시 유저 정보 수정(이름, 비밀번호)

3-5-3) manageAlbum()

1. 앨범 목록 Show albums

앨범명 - 가수명 - 출시일

2. 앨범 추가 Add album

앨범 추가, 앨범을 만든 가수 연결

3-5-4) manageArtist()

1. 아티스트 목록 Show artists
2. 아티스트 추가 Add artist
3. 그룹 관리 Manage group

3-5-4-1) Show artists

각 아티스트 선택 시 showArtistDetails()

3-5-4-2) Add artist

아티스트를 추가한다.

3-5-4-3) Manage Group

각 그룹을 선택하면 멤버를 볼 수 있다.

1. 멤버 추가 Insert member
2. 멤버 삭제 Delete member

3-5-4-1-1) showArtistDetails()

이름, 데뷔일

1. 그룹일 경우만, 멤버 목록 showArtistMembers()
2. 아티스트의 앨범 목록 showArtistAlbums()
3. 아티스트의 음악 목록 showArtistMusics()

3-5-4-1-1-1) showArtistMembers()

그룹에 소속된 멤버들을 보여준다.

1. 멤버 추가 Add member

3-5-4-1-1-2) showArtistAlbums()

아티스트의 앨범 목록들을 보여준다.

1. 앨범 추가 Add album

3-5-4-1-1-3) showArtistMusics()

아티스트의 노래 목록들을 보여준다.

1. 음원 추가 Add music

5. 실행화면

user추가(회원가입), 앨범추가, 음원추가, 아티스트 추가 화면은 생략

1. 실행

```
C:\Users\B2105>Mysqldump -u root -p musicplayer --databases musicplayer > musicplayer.sql  
Enter password: *****
```

```
C:\Users\B2105>Mysql -u root -p musicplayer < musicplayer.sql  
Enter password: *****
```

```
C:\Users\B2105>javac MusicPlayer.java
```

```
C:\Users\B2105>java -classpath .;mariadb-java-client-2.7.1.jar MusicPlayer
```

2. 로그인, 메인화면

```
-----  
Music Player  
1. Login  
2. Sign Up  
0. Exit  
-----  
* Enter number : 1  
-----  
[ LOGIN ]  
* Enter email : grace915@naver.com  
* Enter password : 1234  
-----  
Hello, grace.  
-----  
[ MAIN MENU ]  
1. Show all  
2. Playlist  
3. Music chart  
4. Search music  
5. Manage  
0. Exit  
-----  
* Enter number :
```


3. Show all musics

```
[ Show All ]
1. Show all Musics
2. Show all albums
3. Show all artists
0. Go Back
-----
* Enter number : 1
-----
[ MUSIC LIST ]
Click on the music to see the details.
1. Love that I recognized at once - Yang Yoseab
2. Caffeine - Yang Yoseab
3. Plz Dont be Sad - Highlight
4. Its still Beautiful - Highlight
5. Life Goes On - BTS
6. Blue & Grey - BTS
7. 그대는 모르쥬 - Yang Yoseab
8. 그 사람 - IU
9. 시간의 바깥 - IU
0. Go Back
* Enter number : 2
-----
[ MUSIC DETAIL ]
Name : Caffeine
album : The Fist Collage
Genre : Ballade
Release Date : 2012-11-26
Recommend : 1
Lyrics : 늦었네 자야 되는데 머릿속에 양은 벌써 다 섰어...
My Play : 31
-----
1. Recommendation
2. Show comments
3. Write comment
4. Add to playlist
5. Show full lyrics
0. Go Back
-----
* Enter number :
```

4. 음악 디테일 메뉴 (중 추천, 댓글쓰기, 댓글보기)

```
[ MUSIC DETAIL ]
Name : 그 사람
album : Love Poem
Genre : Rock/Metal
Release Date : 2019-11-18
Recommend : 1
Lyrics : 그 사람 돌아보지 않아요...
My Play : 0
```

- ```

1. Recommendation
2. Show comments
3. Write comment
4. Add to playlist
5. Show full lyrics
0. Go Back
```

```
* Enter number : 1
```

```
[MUSIC DETAIL]
Name : 그 사람
album : Love Poem
Genre : Rock/Metal
Release Date : 2019-11-18
Recommend : 2
Lyrics : 그 사람 돌아보지 않아요...
My Play : 0
```

- ```
-----
1. Cancel Recommendation
2. Show comments
3. Write comment
4. Add to playlist
5. Show full lyrics
0. Go Back
```

```
* Enter number : 3
Comment : IU!!
```

```
[ MUSIC DETAIL ]
Name : 그 사람
album : Love Poem
Genre : Rock/Metal
Release Date : 2019-11-18
Recommend : 2
Lyrics : 그 사람 돌아보지 않아요...
My Play : 0
```

- ```

1. Cancel Recommendation
2. Show comments
3. Write comment
4. Add to playlist
5. Show full lyrics
0. Go Back
```

```
* Enter number : 2
```

```
[COMMENTS]
IU!! - grace
```

- ```
-----
0. Go Back
* Enter number :
```

5. Show all albums

```
[ Show All ]
1. Show all Musics
2. Show all albums
3. Show all artists
0. Go Back
-----
* Enter number : 2
-----
[ ALBUM LIST ]
Enter number to see the details.
1.Only You - Yang Yoseab - 2020-12-03
2.BE - BTS - 2020-11-20
3.The Fist Collage - Yang Yoseab - 2012-11-26
4.Can you feel it? - Highlight - 2017-03-20

0. Go Back
* Enter number : 1
-----
[ ALBUM DETAIL ]
Name : Only You
Release Date : 2020-12-03
Genre : ballade
like : 1
Love that I recognized at once
a
-----
1. like
2. Play album

0. Go Back
* Enter number :
```

6. 앨범 디테일 메뉴

```
[ ALBUM DETAIL ]
Name : BE
Release Date : 2020-11-20
Genre : Rap/Hiphop, Dance, Ballade, R&B/Soul
like : 1
Life Goes On
Blue & Grey
-----
```

- 1. like
- 2. Play album

```
0. Go Back
* Enter number : 1
-----
```

```
[ ALBUM DETAIL ]
Name : BE
Release Date : 2020-11-20
Genre : Rap/Hiphop, Dance, Ballade, R&B/Soul
like : 2
Life Goes On
Blue & Grey
-----
```

- 1. Cancel like
- 2. Play album

```
0. Go Back
* Enter number : 2
```

```
[ Play Album ]
< Life Goes On >
어느 날 세상이 멈췄어
[ Play Album ]
< Blue & Grey >
Where is my angel
-----
```

```
[ ALBUM DETAIL ]
Name : BE
Release Date : 2020-11-20
Genre : Rap/Hiphop, Dance, Ballade, R&B/Soul
like : 2
Life Goes On
Blue & Grey
-----
```

- 1. Cancel like
- 2. Play album

```
0. Go Back
* Enter number :
```

7. Show all Artists

```
[ Show All ]
1. Show all Musics
2. Show all albums
3. Show all artists
0. Go Back
-----
* Enter number : 3
-----
[ ARTIST LIST ]
Enter number to see the artist's musics
1.Yang Yoseab ( 2009-10-16 )
2.BTS ( 2013-06-13 )
3.V ( 2013-06-13 )
4.Highlight ( 2009-10-16 )

0. Go Back
* Enter number : 1
-----
[ ARTIST'S MUSICS ]
Love that I recognized at once
Caffeine
a

0. Go back
* Enter number :
```

8. playlist menu

```
[ MAIN MENU ]
1. Show all
2. Playlist
3. Music chart
4. Search
5. Manage
0. Exit
-----
* Enter number : 2
-----
[ PLAYLISTS ]
1. My playlists
2. Add playlist
3. Delete playlist
4. Show all playlists
0. Go Back
* Enter number :
```

9. my playlist

```
[ PLAYLISTS ]
1. My playlists
2. Add playlist
3. Delete playlist
4. Show all playlists
0. Go Back
* Enter number : 4
-----
1. favorite3
2. favorite
3. myFavorite
4. bb playlist
0. Go Back
* Enter a number if you want to play a playlist : 3
-----
[Play list - myFavorite ]
1. Play
0. Go Back
-----
(1) Love that I recognized at once
(2) Caffeine
(3) Plz Dont be Sad
(4) Blue & Grey
* Enter number : 1
< Love that I recognized at once >
그대 눈을 바라보면 마음이 떨려요...
< Caffeine >
늦었네 자야 되는데 머릿속에 양은 벌써 다 섰어...
< Plz Dont be Sad >
오늘따라 유난히 웃지않는...
< Blue & Grey >
Where is my angel
-----
```

10. show all playlists

```
[ PLAYLISTS ]
1. My playlists
2. Add playlist
3. Delete playlist
4. Show all playlists
0. Go Back
* Enter number : 4
-----
1. favorite3
2. favorite
3. myFavorite
4. bb playlist
0. Go Back
* Enter a number if you want to play a playlist : 1
-----
[Play list - favorite3 ]
1. Play
0. Go Back
-----
(1) Love that I recognized at once
* Enter number : 1
< Love that I recognized at once >
그대 눈을 바라보면 마음이 떨려요...
-----
[Play list - favorite3 ]
1. Play
0. Go Back
-----
(1) Love that I recognized at once
* Enter number :
```

11. Chart

```
[ MUSIC Chart ]
(1) Love that I recognized at once : 43
(2) a : 37
(3) Life Goes On : 36
(4) Plz Dont be Sad : 30
(5) Caffeine : 29
(6) Blue & Grey : 21
0. Go Back
* Enter number :
```

12. search

```
[ MAIN MENU ]
1. Show all
2. Playlist
3. Music chart
4. Search
5. Manage
0. Exit
-----
* Enter number : 4
-----
[ SEARCH ]
* Enter word to search : be
< Music >
Plz Dont be Sad
Its still Beautiful
< Artist >
< Album >
BE
0. Go Back
* Enter number :
```

13. Manage

```
[ MAIN MENU ]
1. Show all
2. Playlist
3. Music chart
4. Search
5. Manage
0. Exit
-----
* Enter number : 5
-----
[ MANAGE ]
1. Manage Music
2. Manage User
3. Manage album
4. Manage artist
0. Go Back
* Enter number :
```


13-1. Manage 권한 없을 때

```
Music Player
1. Login
2. Sign Up
0. Exit
-----
* Enter number : 1
-----
[ LOGIN ]
* Enter email : b
* Enter password : 4321
-----
Hello, babo.
-----
[ MAIN MENU ]
1. Show all
2. Playlist
3. Music chart
4. Search
5. Manage
0. Exit
-----
* Enter number : 5
You have no authority!
-----
[ MAIN MENU ]
1. Show all
2. Playlist
3. Music chart
4. Search
5. Manage
0. Exit
-----
* Enter number :
```

14. manage album

```
[ MANAGE ]
1. Manage Music
2. Manage User
3. Manage album
4. Manage artist
0. Go Back
* Enter number : 1
-----
[ Manage Music ]
1. Add Music
2. Delete Music
3. Update Music
0. Go Back
* Enter number :
```

15. manage user

```
[ MANAGE ]
1. Manage Music
2. Manage User
3. Manage album
4. Manage artist
0. Go Back
* Enter number : 2
-----
[ MANAGE USER ]
2. Broad
4. gaeun Lee
7. b
8. bb

0. Go Back
* Select user to edit : 7
Original name : b
* Enter new name : babo
Original pw : 1234
* Enter new pw : 4321
-----
[ MANAGE USER ]
2. Broad
4. gaeun Lee
7. babo
8. bb

0. Go Back
* Select user to edit :
```

16. manage album

```
[ MANAGE ]
1. Manage Music
2. Manage User
3. Manage album
4. Manage artist
0. Go Back
* Enter number : 3
-----
[ MANAGE ALBUMS ]
1. Show albums
2. Add album
0. Go Back
* Enter number : 1
-----
[ ALBUM LIST ]
(1) Only You - Yang Yoseab - 2020-12-03
(2) BE - BTS - 2020-11-20
(3) The Fist Collage - Yang Yoseab - 2012-11-26
(4) Can you feel it? - Highlight - 2017-03-20

0. Go Back
* Enter number : 0
-----
[ MANAGE ALBUMS ]
1. Show albums
2. Add album
0. Go Back
* Enter number :
```

17. manage artist

```
[ MANAGE ]
1. Manage Music
2. Manage User
3. Manage album
4. Manage artist
0. Go Back
* Enter number : 4
-----
[ MANAGE ARTISTS ]
1. Show artists
2. Add artist
3. Manage group
0. Go Back
* Enter number : 1
-----
[ ARTIST LIST ]
1. Yang Yoseab
2. BTS
3. V
4. Highlight

0. Go Back
* Select artist to see details:
```

18. artist details

```
[ ARTIST LIST ]
1. Yang Yoseab
2. BTS
3. V
4. Highlight

0. Go Back
* Select artist to see details: 4
-----
[ ARTIST DETAIL ]
Name : Highlight
Debut Date : 2009-10-16
-----
1. Show members
2. Show albums
3. Show musics
0. Go Back
* Enter number : 1
-----
[ ARTIST MEMBERS ]
Yang Yoseab

1. Add member
0. Go Back
* Enter number : 0
-----
[ ARTIST DETAIL ]
Name : Highlight
Debut Date : 2009-10-16
-----
1. Show members
2. Show albums
3. Show musics
0. Go Back
* Enter number : 2
-----
[ ARTIST ALBUMS ]
Can you feel it?

1. Add album
0. Go Back
* Enter number : 0
-----
[ ARTIST DETAIL ]
Name : Highlight
Debut Date : 2009-10-16
-----
1. Show members
2. Show albums
3. Show musics
0. Go Back
* Enter number : 3
-----
[ ARTIST MUSICS ]
Plz Dont be Sad
Its still Beautiful

1. Add music
0. Go Back
* Enter number :
```

19. manage group0

```
[ MANAGE ARTISTS ]
1. Show artists
2. Add artist
3. Manage group
0. Go Back
* Enter number : 3
-----
[ MANAGE GROUP ]
1. Yang Yoseab(2009-10-16)
2. BTS(2013-06-13)
3. V(2013-06-13)
4. Highlight(2009-10-16)
0. Go Back
* Select group to see members :2
<BTS>
V
-----
1. Insert member
2. Delete member
0. Go Back
* Enter number :
```

6. SQL문 명세

중복방지용 sql들을 중복을 찾기위해 사용한것으로 중복방지용 다음에 바로 그다음 sql을 사용한 것은 아니다!

로그인

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM user WHERE email = '%s' AND pw = '%s'", email, pw));
```

회원가입

```
ResultSet rs2 = stmt.executeQuery(String.format("INSERT INTO user VALUES (NULL, '%s', '%s', '%s', '%s', %s)", email, pw, name, registration, manage));
```

전체 음악목록

```
ResultSet rs = stmt.executeQuery("SELECT * FROM music");
```

음원 - 아티스트 목록

```
ResultSet rs1 = stmt.executeQuery(String.format("SELECT * FROM music_release as mr, artist WHERE mr.artist = artist.id and mr.music = %s ", musicId));
```

음원 세부 사항

```
ResultSet rs1 = stmt.executeQuery(String.format("SELECT * FROM music WHERE id = %s", musicId));
```

음원 추천 수

```
ResultSet rs2 = stmt.executeQuery(String.format("SELECT COUNT(*) as count FROM user_recommend WHERE music = %s", musicId));
```

음악 추천 유무 위해

```
ResultSet rs4 = stmt.executeQuery(String.format("SELECT COUNT(*) as count FROM user_recommend WHERE music = %s AND user = %s", musicId, userId));
```

추천 취소

```
ResultSet rs5 = stmt.executeQuery(String.format("DELETE FROM user_recommend WHERE user = %s AND music = %s", userId, musicId));
```

추천

```
ResultSet rs5 = stmt.executeQuery(String.format("INSERT INTO  
user_recommend VALUES (%s, %s)", userId, musicId));
```

내가 플레이 한 횟수

```
ResultSet rs10 = stmt.executeQuery(String.format("SELECT * FROM  
user_play, music WHERE user_play.user = %s AND user_play.music =  
music.id and musicId = %s", userId, musicId));
```

댓글 모음

```
ResultSet rs5 = stmt.executeQuery(String.format("SELECT uc.text,  
user.name FROM user_comment as uc, user WHERE music = %s AND  
uc.user = user.id", musicId));
```

댓글 쓰기

```
ResultSet rs6 = stmt.executeQuery(String.format("SELECT *  
FROM user_comment  
WHERE user = %s AND music = %s", userId, musicId)); // 중복 방지  
ResultSet rs5 = stmt.executeQuery(String.format("INSERT INTO  
user_comment VALUES (%s, %s, '%s')", userId, musicId, inputComment));
```

내 플레이리스트

```
ResultSet rs5 = stmt.executeQuery(String.format("SELECT * FROM playlist  
WHERE user = %s", userId));
```

플레이 리스트에 음원 추가하기

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM  
playlist_registered")); // 중복 방지  
ResultSet rs6 = stmt.executeQuery(String.format("INSERT INTO  
playlist_registered VALUES (%s, '%s', %s)", userId, targetName, musicId));
```

앨범 - 아티스트 목록

```
ResultSet rs = stmt.executeQuery("SELECT * FROM album, album_produce,  
artist WHERE album_produce.artist = artist.id AND album_produce.album =  
album.id");
```

앨범 세부사항

```
ResultSet rs1 = stmt.executeQuery(String.format("SELECT * FROM album  
WHERE id = %s", albumId));
```

앨범 좋아요 수

```
ResultSet rs2 = stmt.executeQuery(String.format("SELECT COUNT(*) as  
count FROM user_like WHERE album = %s", albumId));
```

앨범에 수록된 곡들

```
ResultSet rs3 = stmt.executeQuery(String.format("SELECT * FROM music  
WHERE album = %s", albumId));
```

앨범 좋아요 유무 위해

```
ResultSet rs4 = stmt.executeQuery(String.format("SELECT COUNT(*) as  
count FROM user_like WHERE album = %s AND user = %s", albumId,  
userId));
```

좋아요 취소

```
ResultSet rs6 = stmt.executeQuery(String.format("DELETE FROM user_like  
WHERE user = %s AND album = %s", userId, albumId));
```

좋아요

```
ResultSet rs6 = stmt.executeQuery(String.format("INSERT INTO user_like  
VALUES (%s, %s)", userId, albumId));
```

전체 아티스트

```
ResultSet rs = stmt.executeQuery("SELECT * FROM artist");
```

아티스트가 출시한 음악들

```
ResultSet rs3 = stmt.executeQuery(String.format("SELECT * FROM music,  
music_release, artist WHERE artist.id = %s AND music_release.artist =  
artist.id AND music_release.music = music.id", artistId));
```

내 플레이리스트

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM playlist  
WHERE user = %s", userId));
```

플레이리스트 추가

```
ResultSet rs = stmt.executeQuery(String.format("INSERT INTO playlist  
VALUES (%s, '%s', %s)", userId, name, publicNum));
```

플레이리스트 삭제

```
ResultSet rs2 = stmt.executeQuery(String.format("DELETE FROM playlist  
WHERE user = %s AND name = '%s'", userId, targetName));
```


공개가능한 플레이리스트 목록

```
ResultSet rs = stmt.executeQuery("SELECT * FROM playlist WHERE public = 1 ");
```

플레이리스트에 포함된 음원 목록

```
ResultSet rs1 = stmt.executeQuery(String.format("SELECT * FROM playlist_registered, music WHERE playlist_registered.music = music.id AND user = %s AND playlist_registered.name = '%s'", userId, targetName));
```

플레이리스트에서 음원 삭제

```
ResultSet rs2 = stmt.executeQuery(String.format("DELETE FROM playlist_registered WHERE user = %s AND name = '%s' AND music = %s", userId, targetName, musicId));
```

플레이리스트 하나 선택

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM playlist_registered WHERE user = %s AND name = '%s'", userId, name));
```

음원 재생

```
ResultSet rs2 = stmt.executeQuery(String.format("INSERT INTO user_play VALUES (%s, %s, %s)", userId, musicId, 1)); //첫 재생  
ResultSet rs2 = stmt.executeQuery(String.format("UPDATE user_play SET count = count+1")); //이미 재생한적 있을 때
```

음악 차트 - 플레이수(모든 count 합산)

```
ResultSet rs = stmt.executeQuery("SELECT count, music, SUM(count) FROM user_play GROUP BY music ORDER BY count DESC");
```

검색- 음원, 아티스트, 앨범

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM music WHERE name LIKE '%%%s%%'", query));  
ResultSet rs2 = stmt.executeQuery(String.format("SELECT * FROM artist WHERE name LIKE '%%%s%%'", query));  
ResultSet rs3 = stmt.executeQuery(String.format("SELECT * FROM album WHERE name LIKE '%%%s%%'", query));
```

관리권한 확인

```
ResultSet rs = stmt.executeQuery(String.format("SELECT manage FROM user WHERE id = %s ", userId));
```

음원 추가

```
ResultSet rs2 = stmt.executeQuery(String.format("INSERT INTO music  
VALUES (NULL, '%s', '%s', '%s', '%s', %s, '%s')", name, genre, lyrics, date,  
album, shortenLyrics));
```

방금 추가한 음원 아이디 확인 위해

```
ResultSet rs3 = stmt.executeQuery(String.format("SELECT id FROM music  
order by id desc"));  
rs3.first();
```

가수 추가

```
ResultSet rs4 = stmt.executeQuery(String.format("INSERT INTO  
music_release VALUES(%s, %s)", artist, musicId);
```

작사가 추가

```
ResultSet rs5 = stmt.executeQuery(String.format("INSERT INTO  
music_writer VALUES ('%s', %s)", name, musicId));
```

작곡가 추가

```
ResultSet rs5 = stmt.executeQuery(String.format("INSERT INTO  
music_composer VALUES ('%s', %s)",name,musicId));
```

음원 삭제

```
ResultSet rs2 = stmt.executeQuery(String.format("DELETE FROM music  
WHERE id = %s", musicId));
```

음원 정보 수정

```
ResultSet rs4 = stmt.executeQuery(String.format("UPDATE music SET name  
= '%s', lyrics = '%s', shortenLyrics = '%s' WHERE id = %s", newName,  
newLyrics, shortenLyrics, musicId));
```

유저 정보

```
ResultSet rs = stmt.executeQuery("SELECT * FROM user WHERE manage =  
0");
```

유저 정보 수정

```
ResultSet rs2 = stmt.executeQuery(String.format("UPDATE user SET name  
= '%s', pw = '%s' WHERE id = %s", newName, newPw, userId));
```

앨범 추가

```
ResultSet rs3 = stmt.executeQuery(String.format("INSERT INTO album  
VALUES(null, '%s', '%s', '%s')", name, date, genre));
```

방금 추가한 앨범 아이디를 찾기 위해

```
ResultSet rs2 = stmt.executeQuery(String.format("SELECT id FROM album  
order by id desc"));  
rs2.first();
```

앨범을 제작하다

```
ResultSet rs4 = stmt.executeQuery(String.format("INSERT INTO  
album_produce VALUES(%s, %s)", albumId, artistId));
```

아티스트 목록

```
ResultSet rs = stmt.executeQuery("SELECT * FROM artist");
```

아티스트 추가

```
ResultSet rs = stmt.executeQuery(String.format("INSERT INTO artist  
VALUES (NULL, '%s', '%s')", name, date));
```

아티스트가 그룹에 속할시 소속하다 추가

```
ResultSet rs2 = stmt.executeQuery(String.format("INSERT INTO  
artist_affiliated VALUES(%s, %s)", artistId, groupId));
```

그룹에 소속한 가수

```
ResultSet rs1 = stmt.executeQuery(String.format("SELECT * FROM  
artist_affiliated as aa, artist WHERE team = %s AND aa.artist = artist.id",  
groupId));
```

그룹에서 가수를 삭제

```
ResultSet rs4 = stmt.executeQuery(String.format("DELETE FROM  
artist_affiliated WHERE artist = %s AND team = %s ", artistId, groupId));
```

아티스트 세부사항

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM artist  
WHERE id = %s", artistId));
```

그룹인지 아닌지 판단 위해

```
ResultSet rs1 = stmt.executeQuery(String.format("SELECT * FROM  
artist_affiliated WHERE team = %s", artistId));  
rs1.last();
```

아티스트의 앨범 목록

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM  
album_produce, album WHERE artist = %s AND album_produce.album =  
album.id", artistId));
```

아티스트의 음원 목록

```
ResultSet rs = stmt.executeQuery(String.format("SELECT * FROM  
music_release, music WHERE artist = %s AND music_release.music =  
music.id", artistId));
```