

# Project : MelonMusic

## Package : model

### class JDBCUtil

#### 멤버함수

connection()	conn 객체 확보하는 메서드
disconnection()	conn, pstmt .close 하는 메서드

### class MUserVO

#### 멤버변수(private)

int	uNum	pk(회원번호) / 1부터 시작 ++
String	uid	아이디
String	uPw	비밀번호
String	uName	이름
int	uLastMusic	최근 검색한 노래정보

#### 멤버함수

getter	총 5개
setter	총 5개
toString	1개

#### 테이블

UNUM	INT	PRYMARY KEY	pk(회원번호) / 1부터 시작 ++
UID	VARCHAR(20)	NOT NULL	아이디
UPW	VARCHAR(20)	NOT NULL	비밀번호
UNAME	VARCHAR(20)	NOT NULL	이름
ULASTMUSIC	INT	DEFAULT 0	최근 검색한 노래정보

### class MUserDAO

#### 멤버변수

Connection	conn
PreparedStatement	pstmt

#### 멤버함수

insert(MUserVO){ }	// 아이디, 비밀번호, 이름 입력받고 MUSER테이블에 insert
selectOne(MUserVO){ }	// uLastMusic을 얻기 위한 selectOne
update(MUserVO){ }	// 최근 검색한 노래정보만 업데이트 되는데 숫자로 진행되고
	MMUSIC테이블의 PK값 넣어줌
delete(MUserVO){ }	// 회원탈퇴

### class MMusicVO

#### 멤버변수(private)

int	mNum	pk / 1부터 시작 ++
String	mTitle	노래이름
String	mArtist	가수
int	mHit	조회수

멤버함수	
getter	총 4개
setter	총 4개
toString	1개

테이블			
MNUM	INT	PRYMARY KEY	pk(회원번호) / 1부터 시작 ++
MTITLE	VARCHAR(20)	NOT NULL	노래이름
MARTIST	VARCHAR(20)	NOT NULL	가수
MHIT	INT	DEFAULT 0	조회수

class MMusicDAO	
멤버변수	
Connection	conn
PreparedStatement	pstmt

멤버함수	
insert(MMusicVO){ }	// 데이터 크롤링 할때 사용될 insert
selectAll(MMusicVO){ }	// 1. 전체메뉴를 출력하는 selectAll
	// 2. title로 검색해서 like를 이용해 가져오는 selectAll
	// 3. artist로 검색해서 like를 이용해 가져오는 selectAll
update(MMusicVO){ }	// 조회수를 1 증가하는 update
public boolean hasSample(MmusicVO)	// DB에 데이터 하나라도 있다면 true, 없다면 false

Package : view		
사용자 화면 안내		
초기화면	음악 프로그램	마이페이지
1. 로그인	1. 전체 노래 목록 조회	1. 최근 들은 노래 정보보기
2. 회원가입	2. 노래검색(title/artist)	2. 회원탈퇴
3. 종료	3. 노래추천	3. 마이페이지 나가기
	4.마이페이지	
	5. 로그아웃 (to 초기화면)	

class View		
멤버변수(private)		
int	action	public ( 사용자 선택번호 담기)
Scanner sc = new Scanner(System.in)		

멤버함수	
초기화면 관련 메서드	
public void startUser()	// 초기화면의 시작으로 사용자 화면이 출력될 메서드 (while)
public void loginPage()	// "로그인 페이지입니다" 출력 메서드
public void loginGuide()	// "아이디와 비밀번호를 순서대로 입력해주세요" 출력 메서드
public void writeld()	// "아이디를 입력하여 주세요" 출력 메서드
public String inputString()	// 문자열을 입력받는 메서드
public void writePw()	// "비밀번호를 입력하여 주세요" 출력 메서드
public void checkLogin()	// "아이디 또는 비밀번호를 확인해주세요" 출력 메서드

public void signUpPage()	// "회원가입 메뉴입니다" 출력 메서드
public void signUpGide()	// "아이디 비밀번호 이름을 순서대로 입력하여 주세요" 출력 메서드
public void writeName()	// "이름을 입력하여 주세요" 출력 메서드
public void endUser()	// 초기화면이 종료되는 메서드 (sleep)
<b>멜론 프로그램 관련 메서드</b>	
public void noExistUser()	// "사용자 정보가 존재하지 않아 시작화면으로 이동합니다" 출력 메서드
public void startMelon()	// 멜론 프로그램의 사용자 시작 화면 (while)
public void chartPage()	// "노래목록을 조회합니다" 출력 메서드
public void musicChart (ArrayList<MMusicVO> datas)	// datas를 받아오면 이쁘게 출력할 양식/데이터가 없다면 "데이터가 존재하지 않습니다" 출력 메서드
public void searchMusicPage()	// "노래 검색 메뉴입니다" 출력 메서드
public void writeTA()	// "곡 명이나 가수명을 입력하여 주세요" 출력 메서드
public void searchResult(String str)	// str + "로 검색한 결과입니다" 출력 메서드
public void noExistMusic()	// "노래가 존재하지 않아 메인으로 이동합니다" 출력 메서드
public void choiceMusic()	// "원하시는 노래 번호를 입력하여 주세요" 출력 메서드
public int inputInt()	// 숫자를 입력받는 메서드
public void listenMusic()	// "~ ~ ♪ ♪ ♪ ~ ~" 출력 메서드 (sleep사용 5초정도)
public void musicRecPage()	// "노래 추천 메뉴입니다" 출력 메서드
public void musicInfo(MMusicVO vo)	// 한 곡의 데이터를 받아오면 한곡에 대한 데이터 출력
public void endMelon()	// 멜론 프로그램이 종료되는 메서드
<b>마이 페이지 메서드</b>	
public void startMyPage()	// 마이페이지의 사용자 시작 화면(while)
public void showLastMusic()	// "가장 최근 들은 노래정보를 출력합니다" 출력 메서드
public void userDeletePage()	// "회원 탈퇴 메뉴입니다" 출력 메서드
public void checkDelete()	// "정말로 탈퇴하시겠습니까? Y/N" 출력 메서드 (upper사용할것)
public void suc()	// "회원 탈퇴가 완료되었습니다" 출력 메서드
public void fail()	// "마이 페이지로 돌아갑니다" 출력 메서드
public void endMyPage()	// 마이페이지가 종료되는 메서드
<b>Package : controller</b>	
<b>class Crwaling</b>	
<b>멤버함수</b>	
public static void crawlingdate()	// 데이터 크롤링 하는 메서드
<b>class Controller</b>	
<b>멤버변수</b>	
MUserDAO	uDAO
MMusicDAO	mDAO
View	view
<b>멤버함수</b>	
public Controller()	uDAO, mDAO, view 객체 생성과 DB데이터가 없다면 크롤링 까지 진행하는기본생성자
public void startApp()	// view와 model을 조합해 프로그램이 실행되어지는 메서드