- 기능

STATUS (회원등급)

SELECTALL()

SELECTONE()

초기데이터 삽입 - INSERT()

UPDATE()

DELETE()

MEMBER (회원)

SELECTALL()

로그인 - SELECTONE()

회원가입 - INSERT()

회원정보변경 - UPDATE()

회원탈퇴 - DELETE()

AIRINFO (항공정보)

항공권 목록 출력, 인기순 출력, 시간순 출력 - SELECTALL()

항공권 선택 - SELECTONE()

항공권 삽입(크롤링) - INSERT()

항공권 잔여좌석, 시간, 가격 변경(관리자) - UPDATE()

항공권 삭제(관리자) - DELETE()

AIROPT (항공옵션)

선택한 항공권 옵션목록 - SELECTALL()

- SELECTONE()

초기데이터 삽입 - INSERT()

- UPDATE()

- DELETE()

AIRDET (항공상세)

선택한 항공권 목록 - SELECTALL()

- SELECTONE()

크롤링할 때 같이 삽입 - INSERT()

잔여 좌석 변경 - UPDATE()

- DELETE()

AIRBOOK (항공예매)

예매한 항공권 목록 - SELECTALL()

예매한 항공권 - SELECTONE()

항공권 예매 - INSERT()

예약 변경 - UPDATE()

예약 취소 - DELETE()

==========================================================================

- DB

AIRINFO

- AIRINFO\_NUM -> INT (PK) (AUTO\_INCREMENT)

- AIRINFO\_AIRLINE -> VARCHAR(30) (NOT NULL)

- AIRINFO\_ROUTE -> VARCHAR(30) (NOT NULL)

- AIRINFO\_START -> TIME (NOT NULL)

- AIRINFO\_RETURN -> TIME (NOT NULL)

- AIRINFO\_PRICE -> INT (NOT NULL)

CREATE TABLE AIRINFO(

AIRINFO\_NUM INT PRIMARY KEY AUTO\_INCREMENT,

AIRINFO\_AIRLINE VARCHAR(30) NOT NULL,

AIRINFO\_ROUTE VARCHAR(30) NOT NULL,

AIRINFO\_START TIME NOT NULL,

AIRINFO\_RETURN TIME NOT NULL,

AIRINFO\_PRICE INT NOT NULL,

AIRINFO\_DATE DATE NOT NULL DEFAULT (CURRENT\_DATE)

);

AIROPT

- AIROPT\_NUM -> INT (PK) (AUTO\_INCREMENT)

- AIROPT\_GRADE -> VARCHAR(30) (NOT NULL)

- AIROPT\_PRICE -> INT (NOT NULL)

CREATE TABLE AIROPT(

AIROPT\_NUM INT PRIMARY KEY AUTO\_INCREMENT,

AIROPT\_GRADE VARCHAR(30) NOT NULL,

AIROPT\_PRICE INT NOT NULL

);

AIRDET

- AIRDET\_NUM -> INT (PK) (AUTO\_INCREMENT)

- AIRINFO\_NUM -> INT (FK) (NOT NULL)

- AIROPT\_NUM -> INT (FK) (NOT NULL)

- AIRDET\_CNT -> INT (NOT NULL)

CREATE TABLE AIRDET(

AIRDET\_NUM INT PRIMARY KEY AUTO\_INCREMENT,

AIRINFO\_NUM INT NOT NULL,

AIROPT\_NUM INT NOT NULL,

AIRDET\_CNT INT NOT NULL,

FOREIGN KEY(AIRINFO\_NUM) REFERENCES AIRINFO(AIRINFO\_NUM) ON DELETE CASCADE,

FOREIGN KEY(AIROPT\_NUM) REFERENCES AIROPT(AIROPT\_NUM) ON DELETE CASCADE

);

AIRBOOK

- AIRBOOK\_NUM -> INT (PK) (AUTO\_INCREMENT)

- MEMBER\_ID -> INT (FK) (NOT NULL)

- AIRDET\_NUM -> INT (FK) (NOT NULL)

- AIRBOOK\_CNT -> INT (NOT NULL)

- AIRBOOK\_DATE -> DATE (DEFAULT SYSDATE) (NOT NULL)

CREATE TABLE AIRBOOK(

AIRBOOK\_NUM INT PRIMARY KEY AUTO\_INCREMENT,

MEMBER\_ID INT NOT NULL,

AIRDET\_NUM INT NOT NULL,

AIRBOOK\_CNT INT NOT NULL,

AIRBOOK\_DATE DATE DEFAULT (CURRENT\_DATE), arr.util.date

FOREIGN KEY(MEMBER\_ID) REFERENCES MEMBER(MEMBER\_ID) ON DELETE CASCADE,

FOREIGN KEY(AIRDET\_NUM) REFERENCES AIRDET(AIRDET\_NUM) ON DELETE CASCADE

);

MEMBER

- ID -> VARCHAR(30) (PK)

- PW -> VARCHAR(30) (NOT NULL)

- NAME -> VARCHAR(30) (NOT NULL)

- STATUS\_NUM -> INT (FK) (NOT NULL)

CREATE TABLE MEMBER(

MEMBER\_ID VARCHAR(30) PRIMARY KEY,

MEMBER\_PW VARCHAR(30) NOT NULL,

MEMBER\_NAME VARCHAR(30) NOT NULL,

STATUS\_NUM INT NOT NULL DEFALUT 1,

FOREIGN KEY(STATUS\_NUM) REFERENCES STATUS(STATUS\_NUM) ON DELETE CASCADE

);

STATUS

- STATUS\_NUM -> INT (PK)

- STATUS\_NAME -> VARCHAR(10) (NOT NULL)

CREATE TABLE STATUS(

STATUS\_NUM INT PRIMARY KEY AUTO\_INCREMENT,

STATUS\_NAME VARCHAR(10) NOT NULL

);

==========================================================================

VO

● MemberVO (회원)

private String mId // 회원 아이디 (PK)

private String mPw // 회원 비밀번호

private String mName // 회원 이름

private int sNum // 회원등급 번호 (FK)

● StatusVO (회원등급)

private int sNum // 회원등급 번호 (PK)

private String sName // 회원등급 이름

● AirInfoVO (항공정보)

private int airInfoNum // 항공정보 번호 (PK)

private String airInfoAirline // 항공사

private String airInfoRoute // 비행경로

private Date airInfoStart // 출발시간

private Date airInfoReturn // 도착시간

private airInfoPrice // 가격

● AirOptVO (항공옵션)

private int airOptNum // 항공옵션 번호 (PK)

private String airOptgrade // 좌석등급

private airOptprice // 추가금액

● AirDetVO (항공상세)

private int airDetNum // 항공상세 번호 (PK)

private int airInfoNum // 항공정보 번호 (FK)

private int airOptNum // 항공옵션 번호 (FK)

private int airDetCnt // 잔여좌석 개수

● AirBookVO (항공예매)

private int airBooknum // 항공예매 번호 (PK)

private int mId // 회원 아이디 (FK)

private int airDetNum // 항공상세 번호 (FK)

private int airBookCnt // 예매좌석 개수

private Date airBookDate // 예매 날짜

==========================================================================

DAO

● MemberDAO

public boolean insert(MemberVO mVO)

- 회원가입 : DB에 회원 정보 추가

JDBCUtil.connect() 호출

// 사용자가 입력한 id, pw, name을

// 각각 mVO의 mId, mPw, mName에 받아옴

"INSERT INTO MEMBER (MEMBER\_ID, MEMBER\_PW, MEMBER\_NAME) VALUES('mId', 'mPw', mName);"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public boolean update(MemberVO mVO)

- 회원 정보 수정 : DB에 존재하는 회원의 정보(pw, name)를 수정해서 저장

JDBCUtil.connect() 호출

// 사용자가 입력한 id, pw, name을

// 각각 mVO의 mId, mPw, mName에 받아옴

"UPDATE MEMBER SET MEMBER\_PW='mPw', MEMBER\_NAME='mName' WHERE MEMBER\_ID='mId';"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public boolean delete(MemberVO mVO)

- 회원 탈퇴 : DB에 존재하는 회원을 삭제함

JDBCUtil.connect() 호출

// 사용자가 입력한 id를

// 각각 mVO의 mId에 받아옴

"DELETE FROM MEMBER WHERE MEMBER\_ID='mId';"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public ArrayList<MemberVO> selectAll(MemberVO mVO)

- 해당 기능 없음

null로 반환

————————————————————————————————————————————

public MemberVO selectOne(MemberVO mVO)

- 로그인 : DB에 존재하는 회원인지 확인

JDBCUtil.connect() 호출

// 사용자가 입력한 id, pw를

// 각각 mVO의 mId, mPw에 받아옴

"SELECT MEMBER\_ID, MEMBER\_PW, MEMBER\_NAME FROM MEMBER WHERE MEMBER\_PW='mId' AND MEMBER\_PW='mPw';"

// ㄴ> 쿼리문 실행

if(rs.next())로 임시 객체에 저장

JDBCUtil.disconnect() 호출

임시객체로 반환

// 예외 시 null로 반환

--------------------------------------------------------------------------------------------------------------------

● AirInfoDAO

public boolean insert(AirInfoVO aiVO)

- 항공권 삽입 : DB에 항공권 정보 추가

JDBCUtil.connect() 호출

// 크롤링한 데이터를

// aiVO의 각각의 필드에 받아옴

"INSERT INTO AIRINFO (AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ...) VALUES('airInfoAirline', 'airInfoRoute', ...);"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public boolean update(AirInfoVO aiVO)

- 항공권 시간, 가격 변경(관리자) : DB에 존재하는 항공권 정보를 수정해서 저장

JDBCUtil.connect() 호출

// 사용자가 입력한 aiNum, 변경할 airInfoStart or airInfoReturn or airInfoPrice를

// 각각 aiVO의 aiNum, airInfoStart, airInfoReturn, airInfoPrice에 받아옴

"UPDATE AIRINFO SET AIRINFO\_START='airInfoStart' WHERE AIRINFO\_NUM='aiNum';"

"UPDATE AIRINFO SET AIRINFO\_RETURN='airInfoReturn' WHERE AIRINFO\_NUM='aiNum';"

"UPDATE AIRINFO SET AIRINFO\_PRICE='airInfoPrice' WHERE AIRINFO\_NUM='aiNum';"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public boolean delete(AirInfoVO aiVO)

- 항공권 삭제(관리자) : DB에 존재하는 항공권을 삭제함

JDBCUtil.connect() 호출

// 사용자가 입력한 aiNum를

// 각각 aiVO의 aiNum에 받아옴

"DELETE FROM AIRINFO WHERE AIRINFO\_NUM='aiNum';"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public ArrayList<AirInfoVO> selectAll(AirInfoVO aiVO)

- 항공권 목록 출력, 인기순 출력, 시간순 출력, 시간 필터 검색

JDBCUtil.connect() 호출

// 사용자가 입력한 airInfoStart를

// 각각 aiVO에 받아옴

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO;"

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO ORDER BY AIRINFO\_START ASC;

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO WHERE AIRINFO\_START < airInfoStart;

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO WHERE AIRINFO\_START > airInfoStart;

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO 예매한 항공정보 개수 순;

// ㄴ> 쿼리문 실행

while(rs.next())로 임시 객체에 저장

JDBCUtil.disconnect() 호출

임시객체로 반환

// 예외 시 null로 반환

————————————————————————————————————————————

public AirInfoVO selectOne(AirInfoVO aiVO)

- 항공권 선택 : 선택한 항공권을 가져옴

JDBCUtil.connect() 호출

// 사용자가 입력한 aiNum를

// 각각 aiVO의 aiNum에 받아옴

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO WHERE AIRINFO\_NUM='aiNum';"

// ㄴ> 쿼리문 실행

if(rs.next())로 임시 객체에 저장

JDBCUtil.disconnect() 호출

임시객체로 반환

// 예외 시 null로 반환

================================================================================================

● AirOptDAO

public boolean insert(AirOptVO aoVO)

- 항공옵션 삽입 : DB에 항공권 정보 추가

- 해당 기능 없음

false로 반환

————————————————————————————————————————————

public boolean update(AirOptVO aoVO)

- 해당 기능 없음

false로 반환

————————————————————————————————————————————

public boolean delete(AirOptVO aoVO)

- 해당 기능 없음

false로 반환

————————————————————————————————————————————

public ArrayList<AirOptVO> selectAll(AirOptVO aoVO)

- 선택한 항공권 옵션목록

JDBCUtil.connect() 호출

// 전체 항공권 옵션목록

"SELECT AIROPT\_GRADE, AIROPT\_PRICE FROM AIROPT;" // 수정

// ㄴ> 쿼리문 실행

while(rs.next())로 임시 객체에 저장

JDBCUtil.disconnect() 호출

임시객체로 반환

// 예외 시 null로 반환

————————————————————————————————————————————

public AirOptVO selectOne(AirOptVO aoVO)

- 해당 기능 없음

null로 반환

=============================================================================

AIRDET (항공상세)

선택한 항공권 목록 - SELECTALL()

- SELECTONE()

크롤링할 때 같이 삽입 - INSERT()

잔여 좌석 변경(관리자) - UPDATE()

- DELETE()

● AirDetVO (항공상세)

private int airDetNum // 항공상세 번호 (PK)

private int airInfoNum // 항공정보 번호 (FK)

private int airOptNum // 항공옵션 번호 (FK)

private int airDetCnt // 잔여좌석 개수

● AirDetDAO

public boolean insert(AirDetVO adVO)

- 항공상세 삽입 : DB에 항공상세 추가

JDBCUtil.connect() 호출

// 크롤링한 데이터를

// aiVO의 각각의 필드에 받아옴

"INSERT INTO AIRINFO (AIRDET\_AIRLINE, AIRDET\_ROUTE, ...) VALUES('airInfoAirline', 'airInfoRoute', ...);"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public boolean update(AirDetVO adVO)

- 항공권 잔여좌석(관리자) : DB에 존재하는 항공권 정보를 수정해서 저장

JDBCUtil.connect() 호출

// 사용자가 입력한 aiNum, 변경할 airInfoStart or airInfoReturn or airInfoPrice를

// 각각 aiVO의 aiNum, airInfoStart, airInfoReturn, airInfoPrice에 받아옴

"UPDATE AIRINFO SET AIRINFO\_START='airInfoStart' WHERE AIRINFO\_NUM='aiNum';"

"UPDATE AIRINFO SET AIRINFO\_RETURN='airInfoReturn' WHERE AIRINFO\_NUM='aiNum';"

"UPDATE AIRINFO SET AIRINFO\_PRICE='airInfoPrice' WHERE AIRINFO\_NUM='aiNum';"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public boolean delete(AirDetVO adVO)

- 항공권 삭제(관리자) : DB에 존재하는 항공권을 삭제함

JDBCUtil.connect() 호출

// 사용자가 입력한 aiNum를

// 각각 aiVO의 aiNum에 받아옴

"DELETE FROM AIRINFO WHERE AIRINFO\_NUM='aiNum';"

// ㄴ> 쿼리문 실행

row 확인

JDBCUtil.disconnect() 호출

true로 반환

// 예외 시 false로 반환

————————————————————————————————————————————

public ArrayList<AirDetVO> selectAll(AirDetVO adVO)

- 항공권 목록 출력, 인기순 출력, 시간순 출력, 시간 필터 검색

JDBCUtil.connect() 호출

// 사용자가 입력한 airInfoStart를

// 각각 aiVO에 받아옴

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO;"

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO ORDER BY AIRINFO\_START ASC;

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO WHERE AIRINFO\_START < airInfoStart;

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO WHERE AIRINFO\_START > airInfoStart;

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO 예매한 항공정보 개수 순;

// ㄴ> 쿼리문 실행

while(rs.next())로 임시 객체에 저장

JDBCUtil.disconnect() 호출

임시객체로 반환

// 예외 시 null로 반환

————————————————————————————————————————————

public AirDetVO selectOne(AirDetVO adVO)

- 항공권 선택 : 선택한 항공권을 가져옴

JDBCUtil.connect() 호출

// 사용자가 입력한 aiNum를

// 각각 aiVO의 aiNum에 받아옴

"SELECT AIRINFO\_AIRLINE, AIRINFO\_ROUTE, ... FROM AIRINFO WHERE AIRINFO\_NUM='aiNum';"

// ㄴ> 쿼리문 실행

if(rs.next())로 임시 객체에 저장

JDBCUtil.disconnect() 호출

임시객체로 반환

// 예외 시 null로 반환