**학사관리구현문제**

오늘의 예제]

데이터베이스를 이용하여 다음과 같이 동작하는 프로그램을 구현하시오.

1을 누르면 학생 입력

이름, 전공명, 점수를 입력받아

학번은 시퀀스를 이용하여 "현재년도||학과번호||순차번호"로 입력한다. 데이터베이스의 내용은 다음과 같다

학번(sNO) 이름(sNAME) 학과번호(mNO) 점수(SCORE) 제적여부(sEXPEL)

──────────────────────────────────────────

20201001 정우성 1 90 0

20201002 박세영 1 80 0

20203003 배수지 3 20 0

20204004 홍길동 4 95 0

20205005 송혜교 5 100 0

2를 누르면 원하는 학과이름을 입력받아 학과별 조회후 총점을 추가하여 총점이 높은 순으로 이름(번호)로 출력

등수 이름(nameNO) 학과 점수

───────────────────────────

1등 정우성(20201001) 컴퓨터공학 90

2등 박세영(20201002) 컴퓨터공학 80

3을 누르면 제적당하지 않은 전체 학생을 조회 후 점수가 높은 순으로 아래와 같이 출력한다

등수 이름(nameNO) 학과 점수

───────────────────────────

1등 송혜교(20205005) 연극영화학 100

2등 홍길동(20204004) 정보전자학 95

3등 정우성(20201001) 컴퓨터공학 90

4등 박세영(20201002) 컴퓨터공학 80

5등 배수지(20203003) 산업디자인 20

0. 패키지를 만든다

1. 패키지 안에 erd를 만든다

2. create sql문 실행 - MAJOR테이블에 학과 입력

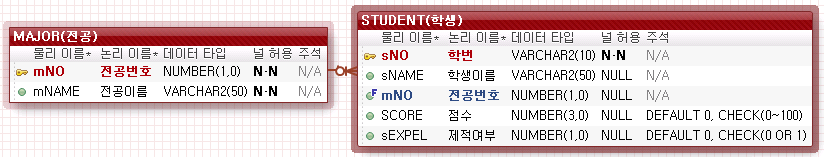
INSERT INTO MAJOR VALUES (1, '컴퓨터공학');

INSERT INTO MAJOR VALUES (2, '경영정보학');

INSERT INTO MAJOR VALUES (3, '산업디자인');

INSERT INTO MAJOR VALUES (4, '정보전자학');

INSERT INTO MAJOR VALUES (5, '연극영화학');



DROP TABLE STUDENT;

DROP TABLE MAJOR;

DROP SEQUENCE STUDENT\_SEQ;

CREATE TABLE MAJOR(

mNO NUMBER(1,0),

mNAME VARCHAR2(50),

PRIMARY KEY(mNO));

CREATE SEQUENCE STUDENT\_SEQ MAXVALUE 999 NOCACHE NOCYCLE;

CREATE TABLE STUDENT(

sNO VARCHAR2(10),

sNAME VARCHAR2(50),

mNO NUMBER(1,0),

SCORE NUMBER(3,0) DEFAULT 0 CHECK(SCORE>=0 AND SCORE<=100),

sEXPEL NUMBER(1,0) DEFAULT 0 CHECK(sEXPEL=0 OR sEXPEL=1),

PRIMARY KEY(sNO),

FOREIGN KEY(mNO) REFERENCES MAJOR(mNO));

INSERT INTO MAJOR VALUES (1, '컴퓨터공학');

INSERT INTO MAJOR VALUES (2, '경영정보학');

INSERT INTO MAJOR VALUES (3, '산업디자인');

INSERT INTO MAJOR VALUES (4, '정보전자학');

INSERT INTO MAJOR VALUES (5, '연극영화학');

COMMIT;

SELECT \* FROM MAJOR;

-- 1번. 이름, 전공명, 점수를 입력받아

-- 학번은 시퀀스를 이용하여 "현재년도||학과번호||순차번호"로 입력한다

SELECT \* FROM STUDENT;

INSERT INTO STUDENT (sNO, sNAME, mNO, SCORE) VALUES

(TO\_CHAR(SYSDATE, 'YYYY')||(SELECT mNO FROM MAJOR WHERE mNAME='컴퓨터공학')

||TRIM(TO\_CHAR(STUDENT\_SEQ.NEXTVAL,'000')),

'박세영',(SELECT mNO FROM MAJOR WHERE mNAME='컴퓨터공학'), 80) ;

SELECT \* FROM STUDENT;

--2를 누르면 원하는 학과이름을 입력받아

-- 학과별 조회후 총점을 추가하여 총점이 높은 순으로 이름(번호)로 출력

SELECT ROWNUM RANK, sNAME||'('||sNO||')' NAME\_NO, mNAME, SCORE

FROM (SELECT \* FROM STUDENT S, MAJOR M WHERE S.mNO=m.mNO AND mNAME='컴퓨터공학'

ORDER BY SCORE DESC);

--3을 누르면

-- 제적당하지 않은 전체 학생을 조회 후 점수가 높은 순으로 아래와 같이 출력한다

--등수 이름(studentID) 학과 점수

--────────────────────────

--1등 송혜교(20205005) 연극영화학 100

SELECT ROWNUM RANK, sNAME||'('||sNO||')' NAME\_NO, mNAME, SCORE

FROM (SELECT \* FROM STUDENT S, MAJOR M WHERE S.mNO=m.mNO AND sEXPEL=0

ORDER BY SCORE DESC);

**package** com.tj.student;

**public** **class** StudentDTO {

**private** **int** rank;

**private** String nameNo;

**private** String mName;

**private** **int** score;

**public** StudentDTO(**int** rank, String nameNo, String mName, **int** score) {

**this**.rank = rank;

**this**.nameNo = nameNo;

**this**.mName = mName;

**this**.score = score;

}

@Override

**public** String toString() {

**return** rank + "등\t" + nameNo + "\t" + mName + "\t" + score;

}

}