쌍용교육센터관리시스템

프로젝트 최종 요약 **|** Project Overview  
김대환 ・ 유동현 ・ 이동재 ・ 이채린 ・ 조연우

**프로젝트 개요**

주제 **|** 쌍용교육센터 관리시스템 데이터베이스 제작

목적 **|** 관리시스템 내 저장된 데이터 최적화, 데이터베이스 자체 기능 구현 및 최적화

개발 기간 **|** 2023년 3월 27일 ~ 2023년 4월 7일

개발 환경 **|** Windows10/Windows11, Oracle Database 11g, SQL Developer

eXERD, Draw.io, Google Drive

사용 기술 **|** ANSI-SQL, PL/SQL

**기능 요약**

관리자 **|** 기초 정보 관리 기능, 교사 계정 관리, 개설 과정 관리, 개설 과목 관리, 교육생 관리,

시험 관리 및 성적 조회, 출결 관리 및 출결 조회, 교육생 면접 및 선발, 상담 관리,

과목별 교재 관리, 수료생 취업 활동 관리, 협력 기업 관리, 교사 추천 도서 관리,

만남의 광장 관리, 달란트 시장 관리

교사 **|** 강의 스케줄 조회, 배점 입출력, 성적 입출력, 출결 조회, 상담일지 작성 및 조회,

교육생 사후처리 조회, 교사 추천 도서 등록 및 조회, 만남의 광장, 퀴즈 관리

교육생 **|** 성적 조회, 출결 관리 및 출결 조회, 사후처리 입력, 교사 추천 도서 조회,

만남의 광장, 달란트 시장

밑줄: 기본 요구사항 외 추가 기능

**협업 방식**

• 기능 고안, 데이터베이스 구조 등 프로젝트 전체를 관통하는 기본 골자는 원칙적으로 커뮤니케이션을 통해 의사 결정 및 공동 요약, 공동 요약문 작업을 통해 전원 숙지

• 개인에게 특정 기능을 전면 위임하지 않고, 다른 계정(관리자/교사/교육생)간 공유되는 기능 및 유사 기능을 두 명 이상이 담당함으로써 문제 발생 시 함께 고민하고 수정할 수 있는 팀 작업 구조 구축

• 구글 드라이브(Google Drive)상 코드 정리 문서 공유를 통해 담당 파트 동시 작업, 작업 완료 후 구글 미트(Google Meet) 화면 공유를 통한 코드 공동 점검 및 리뷰 → 오류 발견 시 해결법 공동 탐색

**데이터 구조**

테이블 **|** 39개 테이블 155개 컬럼

테스트 데이터 **|** 관리자 데이터 5개, 교사 데이터 10개, 학생(수강생+수료생) 데이터 500개

지원생(수강 예정+탈락생) 데이터1,000개, 과목 데이터 41개,

과정 데이터(진행 중 + 진행 완료 및 예정) 21개 등 약 15만 개

A picture containing text, indoor, screenshot

Description automatically generated

물리 ERD

**구현 예시**

쿼리문 **|** 재취업 지원을 위한 6개월 이내 미취업 수료생 열람(관리자)

SELECT

s.student\_seq AS "학생 번호", m.name AS "이름",

c.curs\_name AS "과정명", f.field\_name AS "분야",

jc.company AS "회사명", jc.hire\_date AS "입사일"

FROM job\_current jc

LEFT OUTER JOIN student s

ON jc.student\_seq = s.student\_seq

LEFT OUTER JOIN Member m

ON m.member\_seq = s.member\_seq

LEFT OUTER JOIN curriculum\_list c

ON s.curriculum\_list\_seq = c.curriculum\_list\_seq

LEFT OUTER JOIN job\_field f

ON jc.job\_field\_seq = f.job\_field\_seq

WHERE jc.company IS NULL

AND MONTHS\_BETWEEN(SYSDATE, s.drop\_out\_date) <= 6;

프로시저 **|** 개설 과목 수정(관리자)

CREATE OR REPLACE PROCEDURE curriculum\_subject\_update (

curriculum\_list\_seq in number,

subject\_seq in number

)

IS

BEGIN

UPDATE curriculum\_subject

SET

subject\_seq = subject\_seq

WHERE curriculum\_list\_seq = curriculum\_list\_seq AND subject\_seq = subject\_seq;

EXCEPTION WHEN OTHERS THEN

DBMS\_OUTPUT.put\_line('수정에 실패했습니다.');

ROLLBACK;

END curriculum\_subject\_update;

트리거 **|** 미인정 지각/결석 시 벌점 부여 및 벌점 15점 이상 시 중도탈락 처리

CREATE OR REPLACE TRIGGER attendance\_penalty\_trigger

AFTER INSERT OR UPDATE ON attendance\_detail

FOR EACH ROW

DECLARE

v\_student\_seq student.student\_seq%TYPE;

v\_attendance\_seq attendance\_detail.attendance\_seq%TYPE;

v\_check\_in attendance\_detail.check\_in%TYPE;

v\_check\_out attendance\_detail.check\_out%TYPE;

v\_penalty student.penalty%TYPE;

v\_total\_penalty NUMBER;

BEGIN

v\_attendance\_seq := :NEW.attendance\_seq;

SELECT student\_seq INTO v\_student\_seq FROM student WHERE student\_seq = (SELECT student\_seq FROM attendance WHERE attendance\_seq = v\_attendance\_seq);

IF v\_student\_seq IS NULL THEN

RETURN;

END IF;

v\_check\_in := :NEW.check\_in;

v\_check\_out := :NEW.check\_out;

IF v\_check\_in IS NULL AND v\_check\_out IS NULL THEN

v\_penalty := 3;

ELSIF v\_check\_in IS NOT NULL AND to\_date(v\_check\_in, 'HH24:MI:SS') > to\_date('09:10:00', 'HH24:MI:SS') THEN

v\_penalty := 1;

ELSIF v\_check\_out IS NOT NULL AND to\_date(v\_check\_out, 'HH24:MI:SS') < to\_date('17:50:00', 'HH24:MI:SS') THEN

v\_penalty := 1;

ELSE

v\_penalty := 0;

END IF;

SELECT penalty INTO v\_total\_penalty FROM student WHERE student\_seq = v\_student\_seq;

IF v\_total\_penalty IS NULL THEN

v\_total\_penalty := 0;

END IF;

v\_total\_penalty := v\_total\_penalty + v\_penalty;

IF v\_total\_penalty >= 15 THEN

UPDATE student SET drop\_out = '중도탈락' WHERE student\_seq = v\_student\_seq;

END IF;

UPDATE student SET penalty = v\_total\_penalty WHERE student\_seq = v\_student\_seq;

END;

**제작 후기**

• 업무 SQL 작성 과정

ERD 설계를 바탕으로 업무 기능들을 구현하는 과정에서 DB에 대한 이해도가 높아졌다. 어려운 업무 구현을 다같이 토론하는 과정에서 많은 걸 배울 수 있었다.

• ERD 설계 과정

요구사항을 해석해서 ERD를 설계하고 정규화하는 과정이 가장 어려웠다. ERD를 잘 정리했다면 쿼리문 작성이 더 수월했을 것 같다는 아쉬움이 들었지만, 팀원들과 함께 부족한 부분을 보완하는 과정을 통해 많이 성장할 수 있었던 것 같다.

• ERD 보완점

- 테이블을 정규화하여 많이 만들었지만, 불필요한 테이블들이 존재하는 것 같아 아쉽다.

- 처음 생각했던 것과 다른 방향의 컬럼들이 생성되어, 다소 지저분한 결과가 도출됐다.

- 각 테이블에 개설과목번호와 개설과정번호 중 하나만 있으면 될 것 같다.

- attendance\_detail 테이블에 student\_seq를 FK로 하나 더 추가하면 더 유용할 것 같다.

- 상품구매내역을 별도 테이블로 분리했지만, 통합하여 불필요한 테이블을 감소시키면 조금 더 가독성이 좋을 것 같다.

• 협업툴 사용

구글 드라이브를 활용하여 실시간으로 작업하여 상호작용이 가능했다. 다같이 작업하여 의견을 표현하기 유연했고, 업무의 효율성이 높았다.

• 협업 및 프로젝트 진행 관련

팀원 모두가 적극적으로 참여하여 기한 내에 여유롭게 작업할 수 있었다. 매일 목표를 정하여 자율적으로 추가업무를 맡아서 하여 좋은 결과를 얻을 수 있었다. 프로젝트 협업 시에 소통하는 방법에 대해 배울 수 있었다.