**교육 시설 운영 데이터 관리**

**주제 : Oracle 기반 DB 프로젝트**

**1. 개발환경**

|  |  |
| --- | --- |
| Oracle version | Oracle Database 11g |
| Development tool | SQL Developer |
| ERD tool | eXERD |

**2. 개요 및 목적**

|  |  |
| --- | --- |
| 개 요 | 교육 센터 운영에 필요한 데이터를 관리할 수 있다. |
| 목 적 | 1. 체계적인 센터 운영을 위해 다양한 프로세스를 구성할 수 있어야 한다  2. 교육생을 중심으로 한 다양한 데이터를 조회 및 수정할 수 있어야 한다. |

**3. 핵심 업무**

|  |  |
| --- | --- |
| **목차** | |
| 업무 | 상세내용 |
| **[공통]** | |
| 교육기관에서 과정을 구성하고 강의 일정을 운영하는 것을 목표 | |
| 교육기관에서 과정을 수강한 교육생들에 대한 전반적인 관리를 목표 | |
| 운영을 위해 관리자, 교사, 교육생으로 사용자를 구분 | |
| **[관리자]** | |
| 기초 정보 관리 | 교사 계정 관리 및 개설 과정, 개설 과목에 사용하게 될 기초 정보를 등록 및 관리 |
| 교사/교육생 관리 | 여러 명의 교사와 교육생 정보를 등록 및 관리 |
| 출결 관리 및 조회 | 특정 개설 과정을 선택하는 경우 다양한 방식으로 조회 및 시스템 관리 |
| 개설 과정/과목 관리 | 여러 개의 개설 과정과 과목을 등록 및 관리 |
| 훈련비 지급 관리 | 훈련비 측정 시스템을 관리 |
| 전체 상담 관리 | 전반적인 상담시스템을 등록 및 관리 |
| **[교사]** | |
| 개인 정보 조회 | 자신의 강의 스케줄을 확인 |
| 배점 입출력 | 강의를 마친 과목에 대한 성적 처리를 위해서 배점 입출력 |
| 성적 정보 관리 | 강의를 마친 과목에 대한 성적 처리를 위해서 성적 입출력 |
| 교육생 출결 관리 | 강의한 과정에 한해 선택하는 경우 모든 교육생의 출결을 조회 |
| 교육생 상담 관리 | 교육생의 상담 내용 관리 |
| **[교육생]** | |
| 훈련비 조회 | 본인의 월 별 훈련비 조회 가능 |
| 상담 내역 관리 | 진행 과목에 대한 상담 조회 및 관리 |
| 출결 관리 및 조회 | 매일 근태 기록 및 현황 조회 |

**4. 담당 업무**

|  |  |
| --- | --- |
| 고수경  **[관리자]** | 문서: 요구분석서, 순서도, 쿼리문 정의서  구현:  [관리자] 로그인 기능, 기초 정보 관리, 교사 계정 관리, 개설 과정 관리, 개설 과목 관리 |
| 김영석  **[관리자]** | 문서: 요구분석서, 순서도, 쿼리문 정의서  구현:  [관리자] 시험 관리 및 성적 조회(개설과정별, 과목별, 교육생 개인별, 교육생 출결 현황 조회), 교육생 관리 기능(등록, 수정, 삭제, 조회) |
| 김은지  **[ERD, 총괄, 관리자, 교사, 교육생]** | 문서: 요구분석서, 순서도, DDL, DML(더미), PPT, 발표, 요약본, 전체 문서 수정 및 관리, 전체 설계  구현:  ERD  [관리자] 과목 상담 조회 및 관리  [교사] 과목상담 답변  [교육생] 과목상담 작성 및 조회 오류수정 |
| 백재민  **[관리자, 교사, 교육생]** | 문서: 요구분석서, 순서도, 쿼리문 정의서  구현:  [관리자] 기초정보 관리 - 과목, 강의실, 교재, 교사, 개설과정 정보 수정/삭제하기, 교사 추천 도서 관리  [교사] 강의 스케줄 조회, 성적 입출력, 출결 관리 및 출결 조회, 추천도서 관리 기능 전체  [교육생] 성적 조회, 교사 추천도서 조회 |
| 박채은  **[관리자, 교사, 교육생]** | 문서: 요구분석서, 순서도, 쿼리문 정의서  구현:  [관리자] 수료 교육생 취업활동 관리, 협력기업관리, 개설 과정관리 수료날짜 지정, 출결 관리 및 출결 조회  [교사] 배점 입출력  [교육생] 수료 교육생 취업활동 조회, 협력기업 조회 |
| 황혜연  **[교육생]** | 문서: 요구분석서, 순서도, 쿼리문 정의서  구현:  [교육생] 출결 관리 및 출결 조회, 훈련비 조회, 과목상담 작성 및 조회 |

**5. 데이터 구조(ERD)**

|  |
| --- |
| **Step 1. 과정 생성** |
| **텍스트이(가) 표시된 사진  자동 생성된 설명** |
| **Step 2. 과정 시행 확정** |
|  |
| **Step3. 교육생 등록** |
|  |
| **Step4. 개강 및 운영** |
|  |
| **Step5. 종강 및 사후 관리** |
|  |

**6. 순서도**

|  |
| --- |
| **관리자** |
|  |
| **교사** |
|  |
| **교육생** |
|  |

**7. DML(업무 SQL)**

|  |  |
| --- | --- |
| **ANSI - SQL** | **설명** |
|  | B-09 관리자 추천도서 관리, C-06 교사 추천도서 관리, D-04 학생 추천도서 조회  사용한 기술: 관리자와 교사는 해당 내용을 insert문을 통해 해당 내용을 추가하거나 delete문을 이용해 삭제 update문을 이용해 수정이 가능하도록 하였으며 테이블 간의 inner join을 이용해 현재 추천도서 목록을 조회하도록 하였으며 학생은 이 추천도서 조회목록을 전체or선생님or과목 별로 조회가 가능하도록 where문을 이용해 작성하였다. |
|  | B-11관리자 수료 교육생 취업활동관리, D-07 수료 교육생 취업활동 조회  사용한 기술: 조인을 통한 select문으로 출력. D-07의 경우도 고용보험등록과 협력기업조회일을 제외하면 같은 방법으로 조인된다 |
|  | B-12 관리자 협력 기업 관리  사용한 기술: 협력 기업 테이블에서 기업명, 구인인원, 구인형태, 업종, 모집부분을 select로 출력한다. |
|  | C-07 교사 과목상담답변(관련: 관리자 과목상담관리, 교육생 상담관리)  사용한 기술: 수강한 과목, 상담번호, 교육생이름, 선생님이름, 과목기간 등을 join을 통해 select로 출력한다. |

**8. PL – SQL**

|  |  |
| --- | --- |
| **PL - SQL** | **설명** |
|  | B-10 관리자 과목 상담관리, 교육생 과목 상담관리, 교사 과목 상담관리  관리자는 프로시저를 이용하여 과목명을 입력 받으면 학생과 교사의 상담내역을 조회할 수 있다. |
|  | B-11 관리자 수료 교육생 취업활동관리  관리자는 프로시저를 통해 교육생 번호 혹은 개설과정명, 이름을 입력 받으면 관련된 교육생의 취업활동정보를 조회할 수 있다. |
|  | B-12 협력기업관리, D-08 협력기업조회  관리자, 수료 교육생은 프로시저를 통해 협력기업을 원하는 검색어(구인형태, 업종, 모집분야)로 조회할 수 있다. 수료 교육생은 협력 기업 조회 가능일이 지나면 ‘더 이상 기업을 조회할 수 없습니다’만 출력된다. |
|  | C-06 교사추천도서 등록  교사는 본인의 이름, 비밀번호, 책제목, 연관 과목, 출판사명을 순서대로 입력하면 아래 사진과 같이 추천도서 등록이 완료되었다는 문구와 함께 정상적으로 등록이 이루어진다. |
|  | D-04 학생 교사추천도서 조회하기  학생은 교사가 추천하는 등록된 도서를 전체로 조회하거나 등록한 선생님의 이름으로 검색, 조회를 원하는 과목명의 형태로 교사가 추천하는 도서를 조회할 수 있다. |

**9. 후기**

|  |  |
| --- | --- |
| 잘한 점 | 팀원들과의 원활한 소통과 협력 덕분에 모두의 의견이 전부 반영된 프로젝트로 완성이 되었다고 생각한다. 또한 ERD를 설계할 때와 순서도를 작성할 때 팀원들과 꾸준히 의견을 주고받으며 세세하게 짬으로 인해 ANSI-SQL과 PL/SQL을 작성할 때에 큰 어려움이 없었다고 생각한다. ERD작성에 시간을 투자하여 데이터의 구조는 문제가 없었다. 순서도와 ERD를 팀원 모두의 의견을 통해 만드는 과정 중에 수많은 수정을 거쳐야 했지만 그 의견을 주고받음으로써 만족스러운 ERD와 순서도를 만들어 그에 따른 쿼리문과 PL/SQL문을 잘 만들어 낼 수 있어서 |
| 문제점 및 보완점 | 1. 문제점 - 컴파일은 성공했으나 Trigger 동작 문제 발생  수료 및 중도탈락 처리를 Trigger 처리를 통해 작성하려 했으나 Trigger를 기준으로 사용한 테이블을 Begin에서 사용하여 무한루프에 빠지는 문제가 발생했습니다.  2. 보완점  해당 문제를 보완하기 위해 한 테이블내 트리거를 생성하는 것이 아닌 새로운 테이블과 생성된 테이블을 기준으로 Trigger문을 작성해야 한다는 것을 알게 되었습니다. |
| 후 기 | 설계를 아무리 꼼꼼하게 짰다고 해도 SQL구문을 단계로 넘어갔을 때 무언가의 오류 혹은 의문을 가지게 되는 것을 겪어보고서, ERD 작성의 중요성을 느끼게 되었다.  테이블을 조인시키는 것으로 SELECT문이나 프로시저 등을 만들고 실행시키는 것 자체는 어렵지 않았지만 논리적인 오류나 예상치 못한 결과를 나내는 것이 가장 힘들었다. 하지만 비슷한 구조를 제작하고 데이터의 관계를 이해하며, 팀원들과 소통을 통해 원하는 결과를 만들어가는 과정으로 해결할 수 있었다.  쿼리문을 만드는 시간보다 문서작업이 왜 더 중요한지를 다시 한번 깨달을 수 있는 프로젝트였고 많은 의견이 오고 갔지만 결국 그 의견들이 하나로 만들어져서 우리 수준보다 높은 순서도와 ERD를 만들어 낸 것 같아서 뜻 깊은 프로젝트였던 것 같다.  많은 관계를 형성되었다. 그래서 SQL을 시작단계에서 어떤 관계를 맺고 있는지 모르는 부분도 있었지만, 마무리 단계에서는 대부분의 ERD의 관계를 이해하고 사용할 수 있게 되었다 |