**쿼리문 정의서  
ANSI-SQL SCRIPT**

|  |  |
| --- | --- |
| 작성일자 | 2023-04-03 ~ 2023-04-04 |
| 쌍용교육센터 오라클 프로젝트 1조 | |
| 김대환, 유동현, 이동재, 이채린, 조연우 | |

|  |
| --- |
| **관리자 계정 쿼리문** |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-01 | 작성일 | 2023-04-03 |
| 요구사항명 | 기초 정보 관리 기능 | 작성자 | 이채린 |
| DML | | | |
| --1.과정명 관리(과정명 등록, 수정, 삭제, 조회)  --등록  insert into curriculum\_list values (curriculum\_seq.nextVal, '프론트엔드 개발자 과정');  --수정  update curriculum\_list set curs\_name = '백엔드 개발자 과정' where curriculum\_list\_seq = 22;  --삭제  delete from curriculum\_list where curriculum\_list\_seq = 22;  --조회  select \* from curriculum\_list;  –2. 과목명 관리(과목명을 등록, 수정, 삭제, 조회)  --등록  insert into Subject values (seqSub.nextVal, 'C#');  --수정  update Subject set Subject\_name = 'C##' where subject\_seq = 42;  --삭제  delete from Subject where Subject\_seq = 42;  --조회  select \* from Subject;  -–3. 강의실명 관리(강의실명을 등록, 수정, 삭제, 조회)  --등록  insert into room values (7, '강의실8', 30);  --수정  update room set room\_name = '강의실7' where room\_seq = 7;  --삭제  delete from room where room\_seq = 7;  --조회  select \* from room; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-02 | 작성일 | 2023-04-03 |
| 요구사항명 | 교사 계정 관리 기능 | 작성자 | 이채린 |
| DML | | | |
| --1. 교사 정보 등록(이름, 주민번호 뒷자리, 전화번호, 강의 가능 과목을 입력하여 교사정보를 등록)  insert into teacher values (11, '유재석', '2888888', '010-2222-4243');  --2. 교사 정보 수정  update teacher set name = '유지석' where teacher\_seq = 11;  --3. 교사 정보 삭제  delete from teacher where teacher\_seq = 11;  --4. 교사 정보 조회(이름, 주민번호 뒷자리, 전화번호, 강의 가능 과목)을 출력한다.  select  t.name as "이름",  t.ssn as "주민번호뒷자리",  t.tel as "전화번호",  s.subject\_name as "강의가능과목"  from teacher t  inner join teaching\_subject ts  on t.teacher\_seq = ts.teacher\_seq  inner join subject s  on ts.subject\_seq = s.subject\_seq; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-03 | 작성일 | 2023-04-03 |
| 요구사항명 | 개설 과정 관리 | 작성자 | 이채린 |
| DML | | | |
| --1. 과정등록  -- 과정번호, 과정명, 과정기간, 강의실명, 상세과정  insert into open\_curs values (7, 7, 6, 7, '프론트엔드 기초 과정', '2023/04/03','2023/8/17',26);  --2. 과정 정보 수정  update open\_curs set curs\_name = '백엔드 기초 과정' where open\_curs\_seq = 7;  --3. 과정 삭제  delete from open\_curs where open\_curs\_seq = 7;  --4. 과정 조회  --4-1 개설 과정번호, 과정명, 과정기간, 강의실명, 개설 과목 등록 여부, 교육생 수용 인원을 출력  select distinct oc.open\_curs\_seq as "개설 과정번호", oc.curs\_name as "과정명",  CASE  WHEN ROUND(MONTHS\_BETWEEN(oc.end\_date, oc.begin\_date),1) - FLOOR(MONTHS\_BETWEEN(oc.end\_date, oc.begin\_date)) >= 0.5  THEN CEIL(MONTHS\_BETWEEN(oc.end\_date, oc.begin\_date)) - 0.5  ELSE FLOOR(MONTHS\_BETWEEN(oc.end\_date, oc.begin\_date))  END || '개월' as "과정기간",  r.room\_name as "강의실명",  CASE WHEN cs.subject\_seq is not null THEN '과목 등록 완료' ELSE '과목 등록 미완료' END as "개설 과목 등록 여부",  oc.student\_limit as "교육생 수용 인원"  from open\_curs oc  inner join room r on oc.room\_seq = r.room\_seq  inner join curriculum\_list cl on oc.curriculum\_list\_seq = cl.curriculum\_list\_seq  inner join curriculum\_subject cs on cl.curriculum\_list\_seq = cs.curriculum\_list\_seq;  --4-2 개설 과정 선택시, 개설과목정보(과목명, 과목기간, 교재명, 교사명)을 출력  select sj.subject\_name as "과목명", os.begin\_date as "과목기간(시작)", os.end\_date as "과목기간(끝)", b.subject as "교재명", t.name as "교사명"  from open\_curs oc  inner join curriculum\_subject cs on oc.curriculum\_list\_seq = cs.curriculum\_list\_seq  inner join subject sj on cs.subject\_seq = sj.subject\_seq  inner join open\_subject os on sj.subject\_seq = os.subject\_seq and oc.open\_curs\_seq = os.open\_curs\_seq  inner join subject\_book sb on sj.subject\_seq = sb.subject\_seq  inner join book b on sb.book\_seq = b.book\_seq  inner join teacher t on oc.teacher\_seq = t.teacher\_seq  where oc.open\_curs\_seq = 선택할개설과정번호;  --4-3 개설 과정 선택시, 교육생 정보(이름, 주민번호 뒷자리, 전화번호, 등록일, 수료여부(수료,중도탈락))을 출력  SELECT  m.name AS 이름,  m.ssn AS "주민번호 뒷자리",  m.tel AS 전화번호,  s.enroll\_date AS 등록일,  s.drop\_out AS "수료 여부"  FROM student s  INNER JOIN open\_curs oc  ON s.curriculum\_list\_seq = oc.curriculum\_list\_seq  INNER JOIN member m  ON m.member\_seq = s.member\_seq  WHERE open\_curs\_seq = <조회할 개설 과정 번호>;  –5 과정 수료 처리  update student set drop\_out = '수료완료', drop\_out\_date = sysdate where student\_seq = <교육생번호>; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-04 | 작성일 | 2023-04-03 |
| 요구사항명 | 개설 과목 관리 | 작성자 | 이채린 |
| DML | | | |
| --5. 특정 개설 과정 선택 시 개설 과목 신규 등록을 할 수 있다.  -- 개설 과목 정보에 대한 입력, 수정, 삭제, 조회 기능  --입력  insert into curriculum\_subject values (21, 6);  --수정  update curriculum\_subject set subject\_seq = 7 where curriculum\_list\_seq = 21 and subject\_seq = 6;  --삭제  delete from curriculum\_subject where curriculum\_list\_seq = 21 and subject\_seq = 7;  --조회  select \* from curriculum\_subject; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-05 | 작성일 | 2023-04-03 |
| 요구사항명 | 교육생 관리 | 작성자 | 이채린 |
| DML | | | |
| --입력  insert into member values ('1001', '김도영', '1911111', '010-9635-2576');  --출력(교육생 이름, 주민번호 뒷자리, 전화번호, 등록일, 수강신청 횟수)  select mb.name as "교육생 이름", mb.ssn as "주민번호 뒷자리", mb.tel as "전화번호", st.enroll\_date as "등록일", count(en.enrollment\_seq) as "수강신청 횟수"  from member mb  inner join student st on mb.member\_seq = st.member\_seq  inner join enrollment en on mb.member\_seq = en.member\_seq  group by mb.name, mb.ssn, mb.tel, st.enroll\_date;  --특정 교육생 선택 출력(교육생이 수강 신청한 또는 수강중인, 수강했던 개설 과정 정보(과정명, 과정기간(시작 년월일, 끝 년월일), 강의실, 수료 및 중도탈락 여부, 수료 및 중도탈락 날짜)  select oc.curs\_name as "과정명", oc.begin\_date as "과정기간(시작)", oc.end\_date as "과정기간(끝)", r.room\_name as "강의실", st.drop\_out as "수료 및 중도탈락 여부", st.drop\_out\_date as "수료 및 중도탈락 날짜"  from member mb  inner join student st on mb.member\_seq = st.member\_seq  inner join enrollment en on mb.member\_seq = en.member\_seq  inner join open\_curs oc on en.open\_curs\_seq = oc.open\_curs\_seq  inner join room r on oc.room\_seq = r.room\_seq  where st.student\_seq = <교육생번호>;  --수료 및 중도 탈락 처리(날짜 입력)  update student set drop\_out\_date = sysdate where student\_seq = 31;  -- 강의 예정인 과정, 강의 중인 과정, 강의 종료된 과정 중에서 선택한 과정을 신청한 교육생 정보를 확인할 수 있다.  select mb.name as "교육생 이름", mb.ssn as "주민번호 뒷자리", mb.tel as "전화번호", st.drop\_out as "교육이수여부"  from curriculum\_list cl  inner join student st on cl.curriculum\_list\_seq = st.curriculum\_list\_seq  inner join member mb on st.member\_seq = mb.member\_seq  where cl.curriculum\_list\_seq = <과정번호>;  -- 교육생 검색을 할 수 있다.  select \*  from student s  inner join member m on s.member\_seq = m.member\_seq  where s.student\_seq = <교육생번호>; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-06 | 작성일 | 2023-04-04 |
| 요구사항명 | 시험 관리 및 성적 조회 | 작성자 | 이채린 |
| DML | | | |
| 1. 시험 관리  - 특정 개설 과정 선택 시 등록된 개설 과목 정보를 출력하고, 과목별 성적 등록 여부를 확인할 수 있다.  select distinct sj.subject\_name as "과목명", os.begin\_date as "과목기간(시작)", os.end\_date as "과목기간(끝)",  CASE WHEN ss.attend\_score is not null and ss.test\_writescore is not null and ss.test\_realscore is not null  THEN '성적 등록 완료'  ELSE '성적 등록 미완료'  END as "과목별 성적 등록 여부"  from curriculum\_subject cs  inner join subject sj on cs.subject\_seq = sj.subject\_seq  inner join subject\_score ss on sj.subject\_seq = ss.subject\_seq  inner join open\_subject os on sj.subject\_seq = os.subject\_seq  where cs.curriculum\_list\_seq = <개설과정번호>;  2. 성적 조회  - 개설 과목별, 교육생 개인별로 성적 정보를 출력할 수 있다.  - 개설 과목별 성적 출력 시 과정명, 과정 기간, 강의실명, 과목명, 교사명, 교재명 등을 출력하고, 해당 과목을 수강한 모든 교육생의 성적 정보(이름, 주민번호 뒷자리, 필기, 실기)를 같이 출력한다.  select distinct cl.curs\_name as "과정명", oc.begin\_date as "과정기간(시작)", oc.end\_date as "과정기간(끝)", sj.subject\_name as "과목명", t.name as "교사명", b.subject as "교재명",  mb.name as "교육생 이름", mb.ssn as "주민번호 뒷자리", ss.test\_writescore as "필기", ss.test\_realscore as "실기"  from open\_subject os  inner join subject sj on sj.subject\_seq = os.subject\_seq  inner join curriculum\_subject cs on sj.subject\_seq = cs.subject\_seq  inner join curriculum\_list cl on cs.curriculum\_list\_seq = cl.curriculum\_list\_seq  inner join open\_curs oc on cl.curriculum\_list\_seq = oc.curriculum\_list\_seq  inner join teacher t on oc.teacher\_seq = t.teacher\_seq  inner join subject\_book sb on sj.subject\_seq = sb.subject\_seq  inner join book b on sb.book\_seq = b.book\_seq  inner join student st on cl.curriculum\_list\_seq = st.curriculum\_list\_seq  inner join member mb on st.member\_seq = mb.member\_seq  inner join subject\_score ss on st.student\_seq = ss.student\_seq;  - 교육생 개인별 성적 출력 시 교육생 정보(이름, 주민번호 뒷자리, 과정명, 과정 기간, 강의실명 등)를 출력하고, 교육생이 수강한 모든 과목에 대한 성적 정보(과목명, 과목 기간, 교사명, 필기, 실기)를 같이 출력한다.  select  mb.name as "이름",  mb.ssn as "주민번호 뒷자리",  mb.tel as "전화번호",  cl.curs\_name as "과정명",  st.enroll\_date as "과정 시작날짜",  st.drop\_out\_date as "과정 종료날짜",  oc.room\_seq as "강의실",  sj.subject\_name as "과목명",  os.begin\_date as "과목 시작날짜",  os.end\_date as "과목 종료날짜",  tc.name as "교사명",  sc.test\_writescore as "필기시험점수",  sc.test\_realscore as "실기시험점수"  from student st  inner join member mb  on st.member\_seq = mb.member\_seq  inner join curriculum\_list cl  on st.curriculum\_list\_seq = cl.curriculum\_list\_seq  left outer join open\_curs oc  on st.curriculum\_list\_seq = oc.curriculum\_list\_seq  left outer join open\_subject os  on oc.open\_curs\_seq = os.open\_curs\_seq and oc.teacher\_seq = os.teacher\_seq  inner join subject sj  on os.subject\_seq = sj.subject\_seq  inner join teacher tc  on os.teacher\_seq = tc.teacher\_seq  inner join subject\_score sc  on st.student\_seq = sc.student\_seq and st.curriculum\_list\_seq = sc.curriculum\_list\_seq and sj.subject\_seq = sc.subject\_seq; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-07 | 작성일 | 2023-04-04 |
| 요구사항명 | 출결 관리 및 출결 조회 | 작성자 | 이채린 |
| DML | | | |
| - 개설 과정 선택 시 모든 교육생의 출결을 조회할 수 있다.  select oc.open\_curs\_seq as "과정번호", st.student\_seq as "교육생번호", mb.name as "교육생이름", at.attendance\_date as "출석날짜"  from open\_curs oc  inner join student st on oc.curriculum\_list\_seq = st.curriculum\_list\_seq  inner join attendance at on st.student\_seq = at.student\_seq  inner join member mb on mb.member\_seq = st.member\_seq  where oc.open\_curs\_seq = <개설과정번호>;  - 출결 현황을 기간별(년, 월, 일)로 조회할 수 있다.  select oc.open\_curs\_seq as "과정번호", st.student\_seq as "교육생번호", mb.name as "교육생이름", at.attendance\_date as "출결날짜"  from open\_curs oc  inner join student st on oc.curriculum\_list\_seq = st.curriculum\_list\_seq  inner join attendance at on st.student\_seq = at.student\_seq  inner join member mb on mb.member\_seq = st.member\_seq  WHERE  oc.open\_curs\_seq = <개설과정번호> and  attendance\_date BETWEEN '2023-03-01' AND '2023-03-31'; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-08 | 작성일 | 2023-04-04 |
| 요구사항명 | 교육생 면접 및 선발 | 작성자 | 이채린 |
| DML | | | |
| 1. 지원생 등록(면접을 지원한 지원생들의 이름, 지원일자, 면접 예정일자, 지원동기, 지원과정 등을 등록한다)  insert into member values ('1001', '지원생', '2123123', '010-9999-9999');  insert into Interview values(interviewSeq.nextVal, 1001, '코딩 공부', '웹서비스를 활용한 딥러닝 개발자 과정');  2. 교육생 선발  insert into student values ('501', '6', '2023-01-30', '교육이수중', null, '1000'); | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-09 | 작성일 | 2023-04-03 |
| 요구사항명 | 상담 관리 | 작성자 | 조연우 |
| DML | | | |
| -- 특정 과정을 선택하고, 선택된 과정에 대해 작성된 상담 일지 전체 목록을 작성일자가 최신인 순으로 출력한다(작성일자, 상담 주제, 학생 이름, 상담 내용).  SELECT  c.counsel\_date AS "상담 날짜",  tp.counsel\_subject AS "상담 주제",  m.name AS "학생 이름",  c.counsel\_text AS "상담 내용"  FROM counsel c  INNER JOIN teacher t  ON c.teacher\_seq = t.teacher\_seq  INNER JOIN open\_curs o  ON o.teacher\_seq = c.teacher\_seq  INNER JOIN counsel\_topic tp  ON c.counsel\_topic\_seq = tp.counsel\_topic\_seq  INNER JOIN student s  ON c.student\_seq = s.student\_seq  INNER JOIN member m  ON s.member\_seq = m.member\_seq  WHERE o.open\_curs\_seq = <과정 번호>  ORDER BY c.counsel\_date DESC;  -- 지정된 과정의 전체 상담 일지를 수강생별로 조회할 수 있다.  SELECT  c.counsel\_date AS "상담 날짜",  tp.counsel\_subject AS "상담 주제",  m.name AS "학생 이름",  c.counsel\_text AS "상담 내용"  FROM counsel c  INNER JOIN teacher t  ON c.teacher\_seq = t.teacher\_seq  INNER JOIN open\_curs o  ON o.teacher\_seq = c.teacher\_seq  INNER JOIN counsel\_topic tp  ON c.counsel\_topic\_seq = tp.counsel\_topic\_seq  INNER JOIN student s  ON c.student\_seq = s.student\_seq  INNER JOIN member m  ON s.member\_seq = m.member\_seq  WHERE o.open\_curs\_seq = <과정 번호> AND m.name = <’학생 이름’>  ORDER BY c.counsel\_date DESC; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-10 | 작성일 | 2023-04-04 |
| 요구사항명 | 과목별 교재 관리 | 작성자 | 조연우 |
| DML | | | |
| 1. 교재 등록  - 기초 정보 교재 목록에 교재를 추가할 수 있다.  INSERT INTO BOOK VALUES (<교재 번호>, <’책 제목’>, <’저자’>, <’출판사’>, <발행연도>);  - 개설 과목에 교재 등록 시 기초정보 교재 목록에서 선택적으로 추가할 수 있다.  INSERT INTO subject\_book VALUES (<과목 번호>, <교재 번호>);  2. 교재 조회  - 개설 과목별 교재 정보 출력 시 과목명, 교재명, 출판사 등을 출력한다.  SELECT  b.book\_seq AS "책 번호",  s.subject\_name AS "과목",  b.subject AS "교재명",  b.author AS "저자",  b.publisher AS "출판사",  b.pub\_year AS "출판연도"  FROM book b  INNER JOIN subject\_book sb  ON b.book\_seq = sb.book\_seq  INNER JOIN subject s  on sb.subject\_seq = s.subject\_seq;  3. 교재 수정  - 기초 정보 교재 목록의 교재 정보를 수정할 수 있다.  UPDATE BOOK SET subject = <’책 제목’> WHERE book\_seq = <책 번호>; – 제목 변경  UPDATE BOOK SET author = <’저자’> WHERE book\_seq = <책 번호>; – 저자 변경  UPDATE BOOK SET publisher = <’출판사’> WHERE book\_seq = <책 번호>; – 출판사 변경  UPDATE BOOK SET pub\_year = <’출판연도’> WHERE book\_seq = <책 번호>; – 출판연도 변경  4. 교재 삭제  - 기초 정보 교재 목록의 교재를 삭제할 수 있다.  DELETE FROM BOOK WHERE book\_seq = <책 번호>; – 책 번호로 삭제  DELETE FROM BOOK WHERE subject = <’책 제목’>; – 책 제목으로 삭제 | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-11 | 작성일 | 2023-04-04 |
| 요구사항명 | 수료생 취업 활동 관리 | 작성자 | 조연우 |
| DML | | | |
| 1. 사후처리 등록  INSERT INTO job\_current VALUES(jfSeq.nextVal, <분야 번호>, <학생 번호>, <회사명>, <입사일>);  2. 사후처리 조회 (교육생번호, 교육생, 교육과정 정보, 수료날짜, 취업현황)  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  INNER JOIN student s  ON jc.student\_seq = s.student\_seq  INNER JOIN Member m  ON m.member\_seq = s.member\_seq  INNER JOIN curriculum\_list c  ON s.curriculum\_list\_seq = c.curriculum\_list\_seq  INNER JOIN job\_field f  ON jc.job\_field\_seq = f.job\_field\_seq  WHERE jc.company IS NOT NULL;  3. 관리자는 취업현황에 대한 재취업 지원 시스템을 제공하기 위해 미취업 교육생 목록을 열람할 수 있다. 단, 재취업 지원은 수료 후 6개월 이내의 교육생들에 한해서 지원한다,  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;  4. 관리자는 취업현황에 대한 수정, 삭제 기능을 사용할 수 있다.  - 수정  UPDATE job\_current SET job\_field\_seq = <직업 분야 번호> WHERE student\_seq = <학생 번호>; – 직업 분야 수정  UPDATE job\_current SET company = <회사명> WHERE student\_seq = <학생 번호>; – 회사명 수정  UPDATE job\_current SET hire\_date = TO\_DATE(<입사일>, 'YYYY-MM-DD') WHERE student\_seq = <학생 번호>; – 입사일 수정  -삭제  DELETE FROM job\_current WHERE student\_seq = <학생 번호>; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-12 | 작성일 | 2023-04-04 |
| 요구사항명 | 협력 기업 관리 | 작성자 | 이채린, 조연우 |
| DML | | | |
| --1. 협력 기업을 등록한다(취업공고번호, 지역번호, 분야번호, 계약형태번호, 회사명, 모집인원, 모집시작일, 모집종료일)  insert into Job\_announce values (jobannounceSeq.nextVal, 1, 1, 1, '신한은행', 3, '2023-04-04', '2023-05-04');  --2. 협력 기업 수정  update Job\_announce set company = '국민은행' where job\_announce\_seq = 41;  --3. 협력 기업 삭제  delete from job\_announce where job\_announce\_seq = 41;  --4. 협력 기업 조회  SELECT  job\_announce\_seq AS 공고번호,  c.city\_name AS 지역,  ct.contract\_type AS 계약형태,  f.field\_name AS 분야,  a.company AS 회사명,  a.begin\_date AS 모집시작일,  a.end\_date AS 모집종료일  FROM job\_announce a  INNER JOIN city c  ON c.city\_seq = a.city\_seq  INNER JOIN job\_field f  ON a.job\_field\_seq = f.job\_field\_seq  INNER JOIN contract\_type ct  ON a.contract\_type\_seq = ct.contract\_type\_seq; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-13 | 작성일 | 2023-04-04 |
| 요구사항명 | 추천 도서 관리 | 작성자 | 이채린 |
| DML | | | |
| --1. 추천 도서 추가(추천도서번호(자동생성), 과목번호, 교사번호, 제목)  insert into Comment\_Book values (cbook\_seq.nextVal, 41, 1, '자바의 모든것');  --2. 추천 도서 수정하기  update comment\_book set book\_name = '자바 개론' where comment\_book\_seq = 121;  --3. 추천 도서 삭제하기  delete from comment\_book where comment\_book\_seq = 121;  --4. 추천 도서 조회하기  select subject\_seq as "과목번호", teacher\_seq as "교사번호", book\_name as "책제목" from comment\_book; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-14 | 작성일 | 2023-04-04 |
| 요구사항명 | 만남의 광장 관리 | 작성자 | 이채린 |
| DML | | | |
| --1. 게시물 조회  select b.board\_seq as "게시물번호", b.student\_seq as "교육생번호", b.board\_title as "게시글 제목", b.board\_content as "게시글 본문", br.reply\_seq as "댓글번호", br.student\_seq as "교육생번호(댓글)", br.reply\_content as "댓글내용"  from board b  inner join board\_reply br on b.board\_seq = br.board\_seq;  --2. 게시물 수정  update board set board\_title = '수정할 제목' where board\_seq = 13;  --3. 게시물 삭제  delete from board where board\_seq = 13; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-15 | 작성일 | 2023-04-04 |
| 요구사항명 | 달란트 시장 관리 | 작성자 | 이채린 |
| DML | | | |
| 1. 퀴즈 관리   --입력  insert into QUIZ values(42, '자바의 이해', '다음중 정답은?'||CHR(13)||CHR(10)||'1. 1번 2. 2번 3. 3번 4. 4번 5. 5번', 1, qanswer\_seq.nextVal);  --수정  update quiz set quiz\_title = 'Java의 이해' where quiz\_seq = 42;  --삭제  delete from quiz where quiz\_seq = 42;  --조회  select quiz\_seq as "퀴즈번호", quiz\_title as "퀴즈제목", quiz\_content as "퀴즈내용", subject\_seq as "과목번호", quiz\_answer as "정답번호" from quiz;  2. 포인트 관리  --입력  insert into point values (pointSeq.nextVal, stdpointSeq.nextVal,1000);  --수정  update point set ppoint = 3000 where student\_seq = 451;  --삭제  delete from point where student\_seq = 451;  --조회  select student\_seq as "학생번호", ppoint as "보유 포인트" from point;  3. 상품 관리  --입력  insert into PRIZE values (prizeSeq.nextVal, '슬리퍼', 3000);  --수정  update prize set prize\_point = 2000 where prize\_seq = 21;  --삭제  delete from prize where prize\_seq = 21;  --조회  select prize\_name as "상품명", prize\_point as "상품가격" from prize; | | | |

|  |
| --- |
| **교사 계정 쿼리문** |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-01 | 작성일 | 2023-04-04 |
| 요구사항명 | 강의 스케줄 조회 | 작성자 | 김대환 |
| DML | | | |
| 1. 강의 스케줄 확인  --1. 강의 예정  select  \*  from Curriculum\_list c  where c.curriculum\_list\_Seq between 7 and 12;  --2. 강의 중  select  \*  from Curriculum\_list c  inner join open\_curs o  on c.curriculum\_list\_seq = o.curriculum\_list\_seq;  --3. 강의 종료  select  c.curriculum\_list\_seq,  c.curs\_name  from Curriculum\_list c  left outer join open\_curs o  on c.curriculum\_list\_seq = o.curriculum\_list\_seq  where open\_curs\_seq is null;  2. 강의 상세 정보 출력  -강의 스케줄 정보 출력 시 과정번호, 과정명, 과목번호, 과목이름, 과정기간(시작 연월일, 끝 연월일), 강의실과 과목명, 과목기간(시작 연월일, 끝 연월일), 교재명, 교육생 등록 인원을 확인할 수 있어야 한다.  select  c.curriculum\_list\_seq as "과정번호",  o.curs\_name as "과정명",  p.subject\_seq as "과목번호",  t.subject\_name as "과목이름",  p.begin\_date as "과정기간(시작)",  p.end\_date as "과정기간(끝)",  o.room\_seq as "강의실",  b.subject as "과목명",  o.student\_limit as "수강정원"  from curriculum\_list c  inner join open\_curs o  on c.curriculum\_list\_seq = o.curriculum\_list\_seq  inner join student s  on s.curriculum\_list\_seq = c.curriculum\_list\_seq  inner join open\_subject p  on o.open\_curs\_seq = p.open\_curs\_seq  inner join subject t  on t.subject\_seq = p.subject\_seq  inner join subject\_book k  on k.subject\_seq = t.subject\_seq  inner join book b  on k.book\_seq = b.book\_seq;  2. 교육중인 과정에 등록된 교육생 정보(교육생 이름, 전화번호, 등록일, 수료 또는 중도탈락)을 확인할 수 있어야 한다.  select  b.name as "교육생 이름",  b.tel as "전화번호",  s.enroll\_date as "등록일",  drop\_out as "현재상태"  from student s  inner join member b  on s.member\_seq = b.member\_seq; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-02 | 작성일 | 2023-04-04 |
| 요구사항명 | 배점 입출력 | 작성자 | 김대환 |
| DML | | | |
| 1. 강의를 마친 과목에 대한 성적 처리를 위해서 배점 입출력  -교사는 자신이 강의를 마친 과목의 목록 중에서 특정 과목을 선택하고 해당 과목의 배점 정보를 출결, 필기, 실기로 구분해 등록할 수 있어야 한다. 시험 날짜, 시험 문제를 추가할 수 있어야 한다.  insert into test values (83, 41, '필기 시험', 'firebase 필기 시험');  -출결, 필기, 실기의 배점 비중은 담당 교사가 과목별로 결정한다. (단, 출결은 최소 20점 이상이어야 하고, 출결, 필기, 실기의 합은 100점이 되도록 한다.)  insert into score values (41, 41,30,30,40);  -배점 등록이 안 된 과목인 경우는 과목 정보가 출력될 때 배점 부분은 null 값으로 출력한다.  select  s.subject\_seq as "강의 번호",  j.subject\_name as "강의 이름",  s.score\_attend as "출석 점수",  s.score\_writeTest as "필기 시험",  s.score\_realTest as "실기 시험"  from score s  inner join subject j  on s.subject\_Seq = j.subject\_seq;  2. 과목 목록 출력  - 과목 목록 출력 시 과목번호, 과정명, 과정기간(시작 연월일, 끝 연월일), 강의실, 과목명, 과목기간(시작 연월일, 끝 연월일), 교재명, 출결, 필기, 실기 배점 등이 출력.  select  o.subject\_seq as "과목번호",  c.curs\_name as "과정명",  c.begin\_date as "과정기간(시작)",  c.end\_date as "과정기간(종료)",  room\_seq as "강의실",  s.subject\_name as "과목이름",  o.begin\_date as "과목시작일",  o.end\_date as "과목종료일",  b.subject as "교재",  e.score\_attend as "출석점수",  e.score\_writeTest as "필기점수",  e.score\_realTest as "실기점수"  from subject s  inner join open\_subject o  on s.subject\_seq = o.subject\_seq  inner join open\_curs c  on c.open\_curs\_seq = o.open\_curs\_seq  inner join subject\_book t  on t.subject\_seq = s.subject\_seq  inner join book b  on t.book\_seq = b.book\_seq  inner join score e  on e.subject\_seq =s.subject\_seq; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-03 | 작성일 | 2023-04-04 |
| 요구사항명 | 성적 입출력 | 작성자 | 김대환 |
| DML | | | |
| -교사는 자신이 강의를 마친 과목의 목록 중에서 특정 과목을 선택하면, 교육생 정보가 출력되고, 특정 교육생 정보를 선택하면, 해당 교육생의 시험 점수를 입력할 수 있어야 한다. (이때, 출결, 필기, 실기 점수를 구분해서 입력할 수 있어야 한다.)  -- 강의를 마친 과정 -> 과목번호  select  subject\_seq  from curriculum\_subject  where curriculum\_list\_seq = ( select  s.curriculum\_list\_seq  from student s  where s.drop\_out\_date < sysdate  group by s.curriculum\_list\_seq  having s.curriculum\_list\_seq = 7/\*강의번호\*/) and subject\_seq = 3/\*과목번호\*/;  — 해당 과목에 해당하는 교육생 번호 and teacher 번호  select  s.subject\_seq,  t.teacher\_seq,  t.name,  d.student\_seq,  m.name  from subject s  inner join teaching\_subject e  on s.subject\_seq = e.subject\_seq  inner join teacher t  on t.teacher\_seq = e.teacher\_seq  inner join subject\_score r  on r.subject\_seq = s.subject\_seq  inner join student d  on d.student\_seq = r.student\_seq  inner join member m  on d.member\_seq = m.member\_seq  where s.subject\_seq = 3/\*과목번호\*/  and t.teacher\_seq = 6/\*선생번호\*/ and d.student\_seq = 5/\*학생번호\*/;  -과목 목록 출력 시 과목번호, 과정명, 과정기간(시작 연월일, 끝 연월일), 강의실, 과목명, 과목기간(시작 연월일, 끝 연월일), 교재명, 출결, 필기, 실기 배점, 성적 등록 여부 등이 출력되고, 특정 과목을 과목번호로 선택 시 교육생 정보(이름, 전화번호, 수료 또는 중도탈락) 및 성적이 출결, 필기, 실기 점수로 구분되어서 출력되어야 한다.  select  o.subject\_seq as "과목번호",  c.curs\_name as "과정명",  c.begin\_date as "과정기간(시작)",  c.end\_date as "과정기간(종료)",  room\_seq as "강의실",  s.subject\_name as "과목이름",  o.begin\_date as "과목시작일",  o.end\_date as "과목종료일",  b.subject as "교재",  e.score\_attend as "출석점수",  e.score\_writeTest as "필기점수",  e.score\_realTest as "실기점수"  from subject s  inner join open\_subject o  on s.subject\_seq = o.subject\_seq  inner join open\_curs c  on c.open\_curs\_seq = o.open\_curs\_seq  inner join subject\_book t  on t.subject\_seq = s.subject\_seq  inner join book b  on t.book\_seq = b.book\_seq  inner join score e  on e.subject\_seq =s.subject\_seq;  -과정을 중도 탈락해서 성적 처리가 제외된 교육생이더라도 교육생 명단에는 출력되어야 한다. 중도 탈락 여부를 확인할 수 있도록 해야 한다.  -중도 탈락인 경우 중도탈락 날짜가 출력되도록 한다.  select  s.student\_seq as "교육생 번호",  m.name as "교육생 이름",  s.drop\_out as "현재 상태",  c.attend\_score as "출석 점수",  c.test\_writescore as "필기 점수",  c.test\_realscore as "실기 점수"  from student s  inner join member m  on s.member\_seq = m.member\_seq  inner join subject\_score c  on s.student\_seq = c.student\_seq  where subject\_seq = 3/\*특정과목번호\*/; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-04 | 작성일 | 2023-04-04 |
| 요구사항명 | 출결 조회 | 작성자 | 김대환 |
| DML | | | |
| 1. 교사가 강의한 과정에 한해 선택하는 경우 모든 교육생의 출결을 조회할 수 있어야 한다.   select  c.curriculum\_list\_seq as "과정번호",  c.subject\_seq as "과목번호",  d.student\_seq as "학생번호",  a.attendance\_date as "출석날짜"  from teaching\_subject s  inner join teacher t  on s.teacher\_seq = t.teacher\_seq  inner join curriculum\_subject c  on c.subject\_seq = s.subject\_seq  inner join subject\_score e  on s.subject\_seq = e.subject\_seq  inner join student d  on e.student\_seq = d.student\_seq  inner join attendance a  on a.student\_seq = d.student\_seq  where t.teacher\_seq = 6;  2. 출결 현황을 기간별(년, 월, 일) 조회할 수 있어야 한다.  select  c.curriculum\_list\_seq as "과정번호",  c.subject\_seq as "과목번호",  d.student\_seq as "학생번호",  a.attendance\_date as "출석날짜"  from teaching\_subject s  inner join teacher t  on s.teacher\_seq = t.teacher\_seq  inner join curriculum\_subject c  on c.subject\_seq = s.subject\_seq  inner join subject\_score e  on s.subject\_seq = e.subject\_seq  inner join student d  on e.student\_seq = d.student\_seq  inner join attendance a  on a.student\_seq = d.student\_seq  where t.teacher\_seq = 6 and a.attendance\_date between '2023-01-01' and '2023-04-07';  3. 특정(특정 과정, 특정 인원) 출결 현황을 조회할 수 있어야 한다.  select  c.curriculum\_list\_seq as "과정번호",  d.student\_seq as "학생번호",  a.attendance\_date as "출석날짜"  from teaching\_subject s  inner join teacher t  on s.teacher\_seq = t.teacher\_seq  inner join curriculum\_subject c  on c.subject\_seq = s.subject\_seq  inner join subject\_score e  on s.subject\_seq = e.subject\_seq  inner join student d  on e.student\_seq = d.student\_seq  inner join attendance a  on a.student\_seq = d.student\_seq  where t.teacher\_seq = 6 and a.attendance\_date between '2023-01-01' and '2023-04-07'  and c.curriculum\_list\_seq =3 /\*특정과정\*/  and d.student\_seq = 1; /\* 특정인원 \*/  4. 모든 출결 조회는 근태 상황을 구분할 수 있어야 한다.(정상, 지각, 조퇴, 외출, 병가, 기타)  select  cl.curriculum\_list\_seq as "과정번호",  st.student\_seq as "교육생번호",  mb.name as "교육생이름",  at.attendance\_date as "출석날짜",  CASE WHEN se.sick\_or\_etc IS NULL THEN '출석'  ELSE se.sick\_or\_etc  END AS "근태상황"  from open\_curs oc  inner join teacher t on t.teacher\_seq = oc.teacher\_seq  inner join curriculum\_list cl on cl.curriculum\_list\_seq = oc.curriculum\_list\_seq  inner join student st on cl.curriculum\_list\_seq = st.curriculum\_list\_seq  inner join member mb on mb.member\_seq = st.member\_seq  inner join attendance at on st.student\_seq = at.student\_seq  left outer join sick\_etc se on at.attendance\_seq = se.attendance\_seq  where t.teacher\_seq = 1; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-05 | 작성일 | 2023-04-04 |
| 요구사항명 | 상담일지 작성 및 조회 | 작성자 | 이채린 |
| DML | | | |
| 1. 상담 일지 작성  - 일지 작성 시 작성일자, 과정명이 출력된다. 작성일자는 작성 시점이 기본 출력되나 과거 날짜로 변경할 수 있고, 과정명은 수정할 수 없다.  - 상담 대상인 수강생명, 일지 제목, 상담 내용을 입력할 수 있다.  insert into counsel values (<상담번호>, <교사번호>, <교육생번호>, <상담주제번호>, <상담일자>, <상담내용>);  2. 상담 일지 조회  select counsel\_seq as "상담번호", teacher\_seq as "교사번호", student\_seq as "교육생번호", counsel\_topic\_seq as "상담주제번호", counsel\_date as "상담일자", counsel\_text as "상담내용" from counsel; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-06 | 작성일 | 2023-04-04 |
| 요구사항명 | 교육생 사후처리 조회 | 작성자 | 조연우, 김대환 |
| DML | | | |
| - 과정을 선택하면 해당 과정을 수강한 전체 교육생의 교육생 번호, 교육생 이름, 취업 분야, 취업 회사, 취업일을 출력한다.  SELECT  j.student\_seq AS "학생 번호",  m.name AS "이름",  m.tel AS "전화번호",  f.field\_name AS "취업 분야",  j.company AS "회사",  j.hire\_date AS "취업일"  FROM job\_current j  LEFT OUTER JOIN student s  ON j.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 j.job\_field\_seq = f.job\_field\_seq  WHERE c.curriculum\_list\_seq = <과정 번호>;  - 학생을 선택하면 해당 학생에 대한 사후 처리를 출력한다.  select  s.student\_seq as "학생번호",  m.name as "이름",  m.tel as "전화번호",  j.job\_field\_seq as "분야 번호",  f.field\_name as "분야",  j.company as "회사",  j.hire\_date as "고용날짜"  from job\_current j  inner join student s  on j.student\_seq = s.student\_seq  inner join member m  on m.member\_seq = s.member\_seq  inner join job\_field f  on f.job\_field\_seq = j.job\_field\_seq  where m.name = '학현정'; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-07 | 작성일 | 2023-04-04 |
| 요구사항명 | 교사 추천 도서 등록 및 조회 | 작성자 | 조연우 |
| DML | | | |
| 1. 추천 도서 등록  INSERT INTO comment\_book VALUES (<책 번호>, <과목 번호>, <교사 번호>, ‘<책 제목>’);  2. 추천 도서 조회  - 전체 조회  SELECT \* FROM comment\_book;  - 자기 추천 조회  SELECT \* FROM comment\_book WHERE teacher\_seq = <교사 번호>;  3. 추천 도서 수정  UPDATE comment\_book SET book\_name = ‘<수정할 책 제목>’ WHERE comment\_book\_seq = <책 번호>; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-08 | 작성일 | 2023-04-04 |
| 요구사항명 | 만남의 광장 | 작성자 | 이채린 |
| DML | | | |
| select b.board\_seq as "게시물번호", b.student\_seq as "교육생번호", b.board\_title as "게시글 제목", b.board\_content as "게시글 본문", br.reply\_seq as "댓글번호", br.student\_seq as "교육생번호(댓글)", br.reply\_content as "댓글내용"  from board b  inner join board\_reply br on b.board\_seq = br.board\_seq; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-09 | 작성일 | 2023-04-04 |
| 요구사항명 | 퀴즈 관리 | 작성자 | 이채린 |
| DML | | | |
| --1. 퀴즈 관리  --입력  insert into QUIZ values(42, '자바의 이해', '다음중 정답은?'||CHR(13)||CHR(10)||'1. 1번 2. 2번 3. 3번 4. 4번 5. 5번', 1, qanswer\_seq.nextVal);  --수정  update quiz set quiz\_title = 'Java의 이해' where quiz\_seq = <퀴즈번호>;  --삭제  delete from quiz where quiz\_seq = <퀴즈번호>;  --조회  select quiz\_seq as "퀴즈번호", quiz\_title as "퀴즈제목", quiz\_content as "퀴즈내용", subject\_seq as "과목번호", quiz\_answer as "정답번호" from quiz; | | | |

|  |
| --- |
| **교육생 계정 쿼리문** |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | D-01 | 작성일 | 2023-04-04 |
| 요구사항명 | 성적 조회 | 작성자 | 이동재,유동현 |
| DML | | | |
| 1. 교육생이 로그인에 성공하면 교육생 개인의 정보와 교육생이 수강한 과정명, 과정 기간(시작 연월일, 끝 연월일), 강의실이 타이틀로 출력된다.  select  mb.name as "이름",  mb.ssn as "주민번호",  mb.tel as "전화번호",  cl.curs\_name as "과정 명",  st.enroll\_date as "과정 시작날짜",  st.drop\_out\_date as "과정 종료날짜",  oc.room\_seq as "강의실"  from student st  inner join member mb  on st.member\_seq = mb.member\_seq  inner join curriculum\_list cl  on st.curriculum\_list\_seq = cl.curriculum\_list\_seq  left outer join open\_curs oc  on st.curriculum\_list\_seq = oc.curriculum\_list\_seq;  2. 성적 정보는 과목별로 목록 형태로 출력된다.  select  mb.name as "이름",  sj.subject\_name as "과목명",  sc.attend\_score as "출결점수",  sc.test\_writescore as "필기시험점수",  sc.test\_realscore as "실기시험점수"  from student st  inner join subject\_score sc  on st.student\_seq = sc.student\_seq  inner join member mb  on st.member\_seq = mb.member\_seq  inner join subject sj  on sc.subject\_seq = sj.subject\_seq;  3. 출력될 정보는 과목번호, 과목명, 과목 기간(시작 연월일, 끝 연월일), 교재명, 교사명, 과목별 배점 정보(출결, 필기, 실기 배점), 과목별 성적 정보(출결, 필기, 실기 점수), 과목별 시험날짜, 시험문제가 출력되어야 한다.  select  distinct s.subject\_seq as “과목번호”,  s.subject\_name as “과목명”,  os.begin\_date as “과목시작일자”,  os.end\_date as “과목종료일자”,  b.subject as “교재명”,  t.name as “교사명”,  sc.score\_attend as “출결배점”,  sc.score\_writetest as “필기시험 배점”,  sc.score\_realtest as “실기시험 배점”,  ss.attend\_score as “출결 점수”,  ss.test\_writescore as “필기 점수”,  ss.test\_realscore as “실기 점수”,  td.test\_date as “과목별 시험날짜”,  q.quiz\_content as “시험문제”  from Subject s  inner join open\_subject os  on s.subject\_seq = os.subject\_seq  inner join subject\_book sb  on sb.subject\_seq = s.subject\_seq  inner join book b  on b.book\_seq = sb.book\_seq  inner join teacher t  on os.teacher\_seq = t.teacher\_seq  inner join score sc  on s.subject\_seq = sc.subject\_seq  inner join subject\_score ss  on s.subject\_seq = ss.subject\_seq  inner join student std  on std.student\_seq = ss.student\_seq  inner join test\_date td  on td.subject\_seq = s.subject\_seq  inner join quiz q  on q.quiz\_seq = td.quiz\_seq  where std.student\_seq = (교육생 번호); | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | D-02 | 작성일 | 2023-04-04 |
| 요구사항명 | 출결 관리 및 출결 조회 | 작성자 | 유동현, 이동재 |
| DML | | | |
| 1. 매일 출석을 기록할 수 있어야 한다.(출근 1회, 퇴근 1회)   -- 3번교육생  --출결  insert into attendance values (attendance\_seq.nextVal, 3, sysdate);  --출근  insert into attendance\_detail values (attendance\_detail\_seq.nextVal, 52024, '09:05:30', null);  --퇴근  update attendance\_detail set check\_out = '17:53:28' where attendance\_detail\_seq = 52024;   1. 본인의 출결 현황을 기간별(전체, 월, 일) 조회할 수 있어야 한다.   -- 3번교육생 전체  select attendance\_date from attendance where student\_seq = 3;  -- 3번교육생 22년 10월  select attendance\_date from attendance where student\_seq = 3 and to\_char(attendance\_date, 'yy') = 22 and to\_char(attendance\_date, 'mm') = '10';  -- 3번교육생 22년 10월 31일  select attendance\_date from attendance where student\_seq = 3 and to\_char(attendance\_date, 'yy') = 22 and to\_char(attendance\_date, 'mm') = '10' and to\_char(attendance\_date, 'dd') = 31; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | D-03 | 작성일 | 2023-04-04 |
| 요구사항명 | 상담일지 조회 | 작성자 | 유동현 |
| DML | | | |
| 1. 교육생은 자신의 상담일지를 조회할 수 있다.  select \* from Counsel where student\_seq = (교육생 번호);  2. 상담 날짜, 상담 교사, 상담 학생, 상담 내용이 출력된다.  select  c.counsel\_date as "상담일자",  t.name as "선생님 이름",  m.name as "교육생 이름",  c.counsel\_text as "상담 내용"  from counsel c  inner join teacher t  on c.teacher\_seq = t.teacher\_seq  inner join student s  on s.student\_seq = c.student\_seq  inner join Counsel\_topic n  on n.counsel\_topic\_seq = c.counsel\_topic\_seq  right outer join member m  on s.member\_seq = m.member\_seq  where s.student\_seq = (교육생 번호); | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | D-04 | 작성일 | 2023-04-04 |
| 요구사항명 | 사후처리 입력 | 작성자 | 유동현 |
| DML | | | |
| 1.취업 활동 등록  insert into job\_current(job\_current\_seq, student\_seq, job\_field\_seq, company,hire\_date) values (jfSeq.nextVal, 31,4,'메타넷디지털','2022/3/25' );  2. 취업 활동 조회  select \* from job\_current where student\_seq = (교육생 번호);  3. 취업 활동 수정  update job\_current set job\_field = 분야번호 , company = (‘회사명’), hire\_date = (‘입사날짜’) where student\_seq = (교육생 번호); | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | D-05 | 작성일 | 2023-04-04 |
| 요구사항명 | 교사 추천 도서 | 작성자 | 유동현 |
| DML | | | |
| -교사가 등록한 추천 도서 정보를 조회할 수 있다.  Select  subject\_seq as "과목번호",  eacher\_seq as "교사번호",  book\_name as "책제목"  from comment\_book; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | D-06 | 작성일 | 2023-04-04 |
| 요구사항명 | 만남의 광장 | 작성자 | 유동현 |
| DML | | | |
| 1. 게시물 등록  insert into Board(Board\_seq, student\_seq, Board\_title, Board\_content) values (12, 11, 'JDBC 보충수업이 하고싶어요.', 'JDBC 공부하는데 추천하는 교재가 있나요?');  2. 게시물 조회  select \* from Board;  3. 게시물 수정  update Board set Board\_content = ‘수료후 학원과 연결된 기업이 있나요?’ where student\_seq = (교육생 번호);  4. 게시물 삭제  delete from Board where student\_seq = 11;  5. 답글 작성  insert into Board\_Reply (reply\_seq, Board\_seq, student\_seq, reply\_content) values (Boardreplyseq.nextVal, (게시물번호), (교육생번호), '학원과 연결된 기업으로는 OO기업, OO기업, OO기업 등 이 있습니다.'); | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | D-07 | 작성일 | 2023-04-04 |
| 요구사항명 | 달란트 시장 | 작성자 | 유동현, 이채린 |
| DML | | | |
| 1. 퀴즈 조회  select quiz\_seq as "퀴즈번호", quiz\_title as "퀴즈제목", quiz\_content as "퀴즈내용", subject\_seq as "과목번호", quiz\_answer as "정답번호" from quiz;  2. 퀴즈 응시  insert into QUIZ\_Correct values (quizcorrectSeq.nextVal, <퀴즈번호>, <답변>, <정답여부>, <교육생번호>);  3. 포인트 조회  select \* from point where student\_seq = (교육생 번호);  4. 상품 조회  select prize\_name as "상품명", prize\_point as "상품가격" from prize; | | | |