**DB Object 스크립트(PL-SQL)**

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-01 | 작성일 | 2022-11-02 |
| 요구사항명 | 관리자 계정 | 작성자 | 최유찬 |
| DML | | | |
| **-- 계정으로 로그인 후 관리자 기본정보 간단하게 출력**  create or replace procedure procAdmin (  pjumin in number,  pseq out number,  pname out varchar2  )  is  begin  select seq, name into pseq, pname from tblAdmin where jumin = pjumin;  exception  when others then  dbms\_output.put\_line('예외 처리');  end procAdmin;  declare  vseq tblAdmin.seq%type;  vname tblAdmin.name%type;    begin    procAdmin('**<주민번호 뒷자리>**', vseq, vname);  dbms\_output.put\_line(vseq || ' ' || vname);    end;  create or replace procedure procAdmin (  pjumin in varchar2,  pcursor out sys\_refcursor  )  is  begin  open pcursor for select \* from tblAdmin where jumin = pjumin;  end procAdmin;  declare  vcursor sys\_refcursor;  vrow tblAdmin%rowtype;  begin  procAdmin('**<주민번호 뒷자리>**', vcursor);    loop    fetch vcursor into vrow;  exit when vcursor%notfound;    dbms\_output.put\_line(vrow.seq);  dbms\_output.put\_line(vrow.name);    end loop;  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-02 | 작성일 | 2022-11-02 |
| 요구사항명 | 기초정보 관리기능 | 작성자 | 최유찬 |
| DML | | | |
| **-- 과정명 추가**  create or replace procedure procClass (  pcourseseq in number,  pname in varchar2,  presult out number  )  is  begin    insert into tblClass values (pcourseseq, pname);    presult := 1;    exception  when others then  presult := 0;    end procClass;  declare  vresult number;    begin  procClass(**<과정번호>**, '**<과정명>**', vresult);  dbms\_output.put\_line(vresult);    end;  **-- 과정명 수정**  create or replace procedure procClass (  pcourseseq in number,  pname in varchar2,  presult out number  )  is  begin  update tblClass set name = pname where courseseq = pcourseseq;    presult := 1;  exception  when others then  presult := 0;    end procClass;  declare  vresult number;  begin  procClass(**<과정번호>**, '**<수정할 과정이름>**', vresult);  dbms\_output.put\_line(vresult);  end;  **-- 과정명 삭제**  create or replace procedure procClass (  pcourseseq number,  presult out number  )  is  begin  delete from tblClass where courseseq = pcourseseq;  presult := 1;  exception  when others then  presult := 0;    end procClass;  declare  vresult number;    begin  procClass(**<과정번호>**, vresult);  dbms\_output.put\_line(vresult);    end;  **-- 교재명 입력**  create or replace procedure procBook (  pbookseq in number,  pbookname in varchar2,  presult out number  )  is  begin    insert into tblBook values (pbookseq, pbookname);    presult := 1;    exception  when others then  presult := 0;    end procBook;  declare  vresult number;    begin  procBook(**<교재번호>**, '**<교재명>**', vresult);  dbms\_output.put\_line(vresult);    end;  **-- 교재명 수정**  create or replace procedure procBook (  pbookseq in number,  pbookname in varchar2,  presult out number  )  is  begin  update tblBook set bookname = pbookname where bookseq = pbookseq;    presult := 1;  exception  when others then  presult := 0;    end procBook;  declare  vresult number;  begin  procBook(**<교재번호>**, '**<수정할 교재이름>**', vresult);  dbms\_output.put\_line(vresult);  end;  **-- 교재명 삭제**  create or replace procedure procBook (  pbookseq number,  presult out number  )  is  begin  delete from tblClass where bookseq = pbookseq;  presult := 1;  exception  when others then  presult := 0;    end procBook;  declare  vresult number;    begin  procBook(**<교재번호>**, vresult);  dbms\_output.put\_line(vresult);    end;  **-- 과목명 입력**  create or replace procedure procSubject (  psubseq in number,  pname in varchar2,  pbookseq in number  presult out number  )  is  begin    insert into tblClass values (pcourseseq, pname, pbookseq);    presult := 1;    exception  when others then  presult := 0;    end procSubject;  declare  vresult number;    begin  procClass(**<과목번호>**, '**<과목명>**', **<교재번호>**, vresult);  dbms\_output.put\_line(vresult);    end;  **-- 과목명 수정**  create or replace procedure procSubject (  psubseq in number,  pname in varchar2,  pbookseq in number,  presult out number  )  is  begin  update tblClass set name = pname, bookseq = pbookseq where subseq = psubseq;    presult := 1;  exception  when others then  presult := 0;    end procSubject;  declare  vresult number;  begin  procSubject(**<과목번호>**, '**<수정할 과목이름>**', **<수정할 교재번호>**, vresult);  dbms\_output.put\_line(vresult);  end;  **-- 과목명 삭제**  create or replace procedure procSubject (  psubseq number,  presult out number  )  is  begin  delete from tblSubject where subseq = psubseq;  presult := 1;  exception  when others then  presult := 0;    end procSubject;  declare  vresult number;    begin  procSubject(**<과목번호>**, vresult);  dbms\_output.put\_line(vresult);    end;  **-- 강의실명 입력**  create or replace procedure procClassRoom (  pclassroomseq in number,  pname in varchar2,  pstulimit in number,  presult out number  )  is  begin    insert into tblClassRoom values (pclassroomseq, pname, pstulimit);    presult := 1;    exception  when others then  presult := 0;    end procClassRoom;  declare  vresult number;    begin  procClassRoom(**<강의실번호>**, '**<강의실명>**', **<학생정원>**, vresult);  dbms\_output.put\_line(vresult);    end;  **-- 강의실 수정**  create or replace procedure procClassRoom (  pclassroomseq in number,  pname in varchar2,  pstulimit number,  presult out number  )  is  begin  update tblClassRoom set name = pname, stulimit = pstulimit where classroomseq = pclassroomseq;    presult := 1;  exception  when others then  presult := 0;    end procClassRoom;  declare  vresult number;  begin  procClassRoom(**<강의실번호>**, '**<수정할 강의실이름>**', **<수정할 학생정원>**, vresult);  dbms\_output.put\_line(vresult);  end;  **-- 강의실 삭제**  create or replace procedure procClassRoom (  pclassroomseq number,  presult out number  )  is  begin  delete from tblClassRoom where classroomseq = pclassroomseq;  presult := 1;  exception  when others then  presult := 0;    end procClassRoom;  declare  vresult number;    begin  procClassRoom(**<강의실번호>**, vresult);  dbms\_output.put\_line(vresult);    end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-03 | 작성일 | 2022-11-02 |
| 요구사항명 | 교사계정 관리기능 | 작성자 | 최유찬 |
| DML | | | |
| **-- 교사정보 등록**  create or replace procedure procTeachers (  pteacherseq number,  pname varchar2,  pjumin varchar2,  ptel varchar2,  pschedule varchar2,  presult out number  )  is  begin    insert into tblTeachers values (pteacherseq, pname, pjumin, ptel, pschedule);    presult := 1;    exception  when others then  presult := 0;    end procTeachers;  declare  vresult number;    begin  procTeachers(**<교사번호>**, '**<교사명>**', '**<교사주민번호>**', '**<교사전화번호>**', '**<교사상태>**', vresult);  dbms\_output.put\_line(vresult);    end;  **-- 교사정보 수정**  create or replace procedure procTeachers (  pteacherseq number,  pname varchar2,  pjumin varchar2,  ptel varchar2,  pschedule varchar2,  presult out number  )  is  begin  update tblTeachers set name = pname, jumin = pjumin, tel = ptel, schedule = pschedule where classroomseq = pclassroomseq;    presult := 1;  exception  when others then  presult := 0;    end procTeachers;  declare  vresult number;  begin  procTeachers(**<교사번호>**, '**<수정할 교사명>**', '**<수정할 주민번호>'**, '**<수정할 전화번호>**', '**<수정할 상태>**', vresult);  dbms\_output.put\_line(vresult);  end;  **-- 교사정보 삭제**  create or replace procedure procTeachers (  pteacherseq number,  presult out number  )  is  begin  delete from tblTeachers where teacherseq = pteacherseq;  presult := 1;  exception  when others then  presult := 0;    end procTeachers;  declare  vresult number;    begin  procTeachers(**<교사번호>**, vresult);  dbms\_output.put\_line(vresult);    end;  **-- 특정 교사의 강의과목에 대한 정보 조회**  create or replace procedure procTeachers (  pname varchar2,  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  t.name as "이름",  s.name as "개설과목",  os.startdate as "과목시작",  os.endDate as "과목끝",  c.name as "과정명",  oc.startdate as "과정시작",  oc.enddate as "과정끝",  b.bookname as "교재명",  cr.name as "강의실"    from tblTeachers t  inner join tblPossible p  on t.teacherSeq = p.teacherSeq  inner join tblSubject s  on p.subSeq = s.subSeq  inner join tblOpenSub os  on t.teacherSeq = os.teacherSeq  inner join tblClassSub cs  on s.subSeq = cs.subSeq  inner join tblClass c  on cs.courseSeq = c.courseSeq  inner join tblOpenClass oc  on oc.courseSeq = c.courseSeq  inner join tblClassRoom cr  on cr.classRoomSeq = oc.classRoomSeq  inner join tblBook b  on b.bookSeq = s.bookSeq  where t.name = '**<교사명>**';    end procTeachers;  declare  vcursor SYS\_REFCURSOR;  vTname tblTeachers.name%type;  vSname tblSubject.name%type;  vOSstartdate tblOpenSub.startdate%type;  vOsenddate tblOpenSub.enddate%type;  vCname tblClass.name%type;  vOCstartdate tblOpenClass.startdate%type;  vOCenddate tblOpenClass.enddate%type;  vbookname tblBook.bookname%type;  vCRname tblClassRoom.name%type;    begin  procTeachers('**<교사명>**', vcursor);    loop  fetch vcursor into vTname, vSname, vOSstartdate, vOsenddate, vCname, vOCstartdate, vOCenddate, vbookname, vCRname;  exit when vcursor%notfound;    if vOCstartdate > sysdate    dbms\_output.put\_line('교사명 : ' || vTname || ' | ' || '과목명 : ' || vSname || ' | ' || '과목기간 : ' || vOSstartdate || ' - ' || vOSenddate || ' | ' || '과정명 : ' || vCname || ' | '  || ' | ' || '과정기간 : ' || vOCstartdate || ' - ' || vOCenddate || ' | ' || '교재명 : ' || vbookname || ' | ' || '강의실 : ' || vCRname || ' | ' || '강의 예정');    elsif vOCenddate < sysdate    dbms\_output.put\_line('교사명 : ' || vTname || ' | ' || '과목명 : ' || vSname || ' | ' || '과목기간 : ' || vOSstartdate || ' - ' || vOSenddate || ' | ' || '과정명 : ' || vCname || ' | '  || ' | ' || '과정기간 : ' || vOCstartdate || ' - ' || vOCenddate || ' | ' || '교재명 : ' || vbookname || ' | ' || '강의실 : ' || vCRname || ' | ' || '강의 종료');  else  dbms\_output.put\_line('교사명 : ' || vTname || ' | ' || '과목명 : ' || vSname || ' | ' || '과목기간 : ' || vOSstartdate || ' - ' || vOSenddate || ' | ' || '과정명 : ' || vCname || ' | '  || ' | ' || '과정기간 : ' || vOCstartdate || ' - ' || vOCenddate || ' | ' || '교재명 : ' || vbookname || ' | ' || '강의실 : ' || vCRname || ' | ' || '강의중');    end if;  end loop;  close vcursor;  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-04 | 작성일 | 2022-11-02 |
| 요구사항명 | 개설과정관리기능 | 작성자 | 최유찬 |
| DML | | | |
| **-- 개설 과정 정보 출력시 개설 과정명, 개설 과정기간, 강의실명, 개설 과목 등록 여부, 교육생 등록 인원 출력**  create or replace procedure procOpenClass (  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  c.name as "개설과정명",  oc.startdate as "과정시작",  oc.enddate as "과정끝",  cr.name as "강의실명",  p.등록인원  from tblOpenClass oc  inner join tblclass c  on oc.courseSeq = c.courseSeq  inner join tblClassroom cr  on oc.classroomSeq = cr.classroomSeq  inner join (select  openSeq,  count(stuSeq) as "등록인원"  from tblApply  group by openSeq) p  on oc.openSeq = p.openSeq  order by oc.openSeq;  end procOpenClass;  declare  vcursor SYS\_REFCURSOR;  vCname tblClass.name%type;  vOCstartdate tblOpenClass.startdate%type;  vOCenddate tblOpenClass.enddate%type;  vCrname tblClassRoom.name%type;  vcount number;    begin  procOpenClass(vcursor);    loop  fetch vcursor into vCname, vOCstartdate, vOCenddate, vCrname, vcount;  exit when vcursor%notfound;    dbms\_output.put\_line('과정명: ' || vCname || ' | ' || '과정기간: ' || vOCstartdate || '-' || vOCenddate || ' | ' || '강의실: ' || vCrname || ' | ' || '등록인원: ' || vcount );    end loop;  close vcursor;  end;  **-- 특정 개설 과정 선택 시 개설과목정보(과목명, 과목기간, 교재명, 교사명)과 교육생 정보(이름, 주민번호 뒷자리, 전화번호, 등록일, 수료 및 중도탈락)을 출력한다.**  create or replace procedure procOpenClass (  pseq number,  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  s.name as 과목명,  os.startdate as 과목시작,  os.enddate as 과목끝,  b.bookname as 교재명,  t.name as 교사명,  st.name as 학생명,  st.jumin as 학생주민번호,  st.tel as 학생전화번호,  st.regdate as 등록일,  st.fail as 상태    from tblOpenClass oc  inner join tblClassOpen co  on oc.openSeq = co.openSeq  inner join tblOpenSub os  on os.opensubSeq = co.opensubSeq  inner join tblSubject s  on s.subSeq = os.subSeq  inner join tblBook b  on b.bookSeq = s.bookSeq  inner join tblTeachers t  on t.teacherSeq = os.teacherSeq  inner join tblApply a  on oc.openSeq = a.openSeq  inner join tblStudents st  on st.stuSeq = a.stuSeq  where oc.openSeq = **<개설과정번호>**;  end procOpenClass;  declare  vcursor sys\_refcursor;  vSname tblSubject.name%type;  vOSstartdate tblOpenSub.startdate%type;  vOSenddate tblOpenSub.enddate%type;  vBookname tblBook.bookname%type;  vTname tblTeachers.name%type;  vStname tblStudents.name%type;  vStjumin tblStudents.jumin%type;  vSttel tblStudents.tel%type;  vStregdate tblStudents.regdate%type;  vStfail tblStudents.fail%type;    begin  procOpenClass(**<개설과정번호>**, vcursor);    loop  fetch vcursor into vSname, vOSstartdate, vOSenddate, vBookname, vTname, vStname, vStjumin, vSttel, vStregdate, vStfail;  exit when vcursor%notfound;  if vStfail is not null then    dbms\_output.put\_line('과목명: ' || vSname || ' | ' || '과목기간 : ' || vOSstartdate || ' - ' || vOSenddate || ' | ' || '교재명: ' || vBookname || ' | '  || '교사명: ' || vTname || ' | ' || '교육생명: ' || vStname || ' | ' || '교육생주민번호: ' || vStjumin || ' | ' || '교육생 전화번호: ' || vSttel || ' | ' || '등록일: ' || vStregdate  || ' | ' || '상태: ' || '중도탈락');    else    dbms\_output.put\_line('과목명: ' || vSname || ' | ' || '과목기간 : ' || vOSstartdate || ' - ' || vOSenddate || ' | ' || '교재명: ' || vBookname || ' | '  || '교사명: ' || vTname || ' | ' || '교육생명: ' || vStname || ' | ' || '교육생주민번호: ' || vStjumin || ' | ' || '교육생 전화번호: ' || vSttel || ' | ' || '등록일: ' || vStregdate  || ' | ' || '상태: ' || '수료');    end if;    end loop;  close vcursor;  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-05 | 작성일 | 2022-11-02 |
| 요구사항명 | 개설 과목 관리 기능 | 작성자 | 최유찬 |
| DML | | | |
| **-- 개설 과목 출력 시 개설 과정 정보, 과목명, 과목기간, 교재명, 교사명을 출력한다**  create or replace procedure procSubject (  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  os.openSubSeq as 개설과정번호,  s.name as 과목명,  os.startdate as 과목시작,  os.enddate as 과목끝,  b.bookName as 교재명,  t.name as 교사명  from tblOpenSub os  inner join tblSubject s  on os.subSeq = s.subSeq  inner join tblBook b  on b.bookSeq = s.bookSeq  inner join tblTeachers t  on t.teacherSeq = os.teacherSeq;  end procSubject;  declare  vcursor sys\_refcursor;  vOSopensubseq tblOpensub.opensubseq%type;  vSname tblSubject.name%type;  vOSstartdate tblOpensub.startdate%type;  vOSenddate tblOpensub.enddate%type;  vBbookname tblBook.bookname%type;  vTname tblTeachers.name%type;    begin  procSubject(vcursor);    loop  fetch vcursor into vOSopensubseq, vSname, vOSstartdate, vOSenddate, vBbookname, vTname;  exit when vcursor%notfound;    dbms\_output.put\_line('개설과목번호: ' || vOSopensubseq || ' | ' || '과목이름: ' || vSname || ' | ' || '과목기간: ' || vOSstartdate || '-' || vOSenddate || ' | ' || '교재명: ' || vBbookname || ' | ' || '교사명: ' || vTname);    end loop;  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-06 | 작성일 | 2022-11-02 |
| 요구사항명 | 교육생 관리 기능 | 작성자 | 최유찬 |
| DML | | | |
| **-- 특정 교육생 선택 시 수강신청 또는 수강중인, 수강했던 과정 정보(과정명, 과정기간, 강의실, 수료 및 중도탈락 여부, 수료 및 중도탈락 날짜)를 출력한다.**  create or replace procedure procStudent (  pname varchar2,  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  s.name as 학생명,  c.name as 과정명,  oc.startdate as 과정시작,  oc.enddate as 과정끝,  cr.name as 강의실,  s.fail as 중도탈락여부,  s.enddate as 수료날짜  from tblStudents s  inner join tblApply a  on s.stuSeq = a.stuSeq  inner join tblOpenClass oc  on oc.openSeq = a.openSeq  inner join tblClass c  on c.courseSeq = oc.courseSeq  inner join tblClassroom cr  on cr.classroomSeq = oc.classroomSeq  where s.name = '**<교육생명>**';  end procStudent;  declare  vcursor sys\_refcursor;  vSname tblStudents.name%type;  vCname tblClass.name%type;  vOCstartdate tblOpenClass.startdate%type;  vOCenddate tblOpenClass.enddate%type;  vCrname tblClassRoom.name%type;  vSfail tblStudents.fail%type;  vSenddate tblStudents.enddate%type;    begin  procStudent('**<교육생명>**', vcursor);    loop  fetch vcursor into vSname, vCname, vOCstartdate, vOCenddate, vCrname, vSfail, vSenddate;  exit when vcursor%notfound;    dbms\_output.put\_line('교육생명: ' || vSname || ' | ' || '과정기간: ' || vOCstartdate || '-' || vOCenddate || ' | ' || '강의실명 : ' || vCrname || ' | ' || '중도탈락여부: ' || vSfail  || ' | ' || '수료날짜: ' || vSenddate );    end loop;  close vcursor;  end;  **-- 강의 종료된 과정 목록에서 선택한 과정의 교육생 정보를 확인할 수 있다. (졸업생 정보)**  create or replace procedure procStudent (  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  s.stuSeq as 학생번호,  s.name as 학생이름,  s.jumin as 주민번호,  s.tel as 전화번호,  s.regdate as 등록일,  s.enddate as 수료일  from tblStudents s  inner join tblApply a  on s.stuSeq = a.stuSeq  inner join tblOpenClass oc  on oc.openSeq = a.openSeq  where oc.enddate < sysdate and s.enddate is not null;  end procStudent;  declare  vcursor sys\_refcursor;  vSstuseq tblStudents.stuseq%type;  vSname tblStudents.name%type;  vSjumin tblStudents.jumin%type;  vStel tblStudents.tel%type;  vSregdate tblStudents.regdate%type;  vSenddate tblStudents.enddate%type;    begin  procStudent(vcursor);    loop  fetch vcursor into vSstuseq, vSname, vSjumin, vStel, vSregdate, vSenddate;  exit when vcursor%notfound;    dbms\_output.put\_line('교육생번호: ' || vSstuseq || ' | ' || '교육생명: ' || vSname || ' | ' || '교육생주민번호: ' || vSjumin || ' | ' || '교육생전화번호: ' || vStel || ' | '  || '등록일: ' || vSregdate || ' | ' || '수료날짜: ' || vSenddate);    end loop;  close vcursor;  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-07 | 작성일 | 2022-11-02 |
| 요구사항명 | 시험 관리 및 성적 조회 | 작성자 | 김하늘 |
| DML | | | |
| **-- 1. 특정 개설 과정의 개설 과목 정보 조회**  **-- (개설과목명, 과목시작일, 과목종료일, 교재명, 교사명)**  create or replace procedure procSubInfo (  pseq in tblOpenClass.openSeq%type  )  is  vname tblSubject.name%type;  vstartDate tblOpenSub.startDate%type;  vendDate tblOpenSub.endDate%type;  vbookname tblBook.bookName%type;  vteachername tblTeachers.name%type;    cursor vcursor  is  select  j.name,  o.startDate,  o.endDate,  b.bookname,  t.name  from tblOpenSub o  inner join tblClassOpen c  on o.openSubSeq = c.openSubSeq  inner join tblOpenClass p  on p.openSeq = c.openSeq  inner join tblSubject j  on j.subSeq = o.subSeq  inner join tblBook b  on b.bookSeq = j.bookSeq  inner join tblTeachers t  on t.teacherSeq = o.teacherSeq  where c.openSeq = pseq  order by c.openSeq asc;  begin  dbms\_output.put\_line('[과목명] [과목시작일][과목종료일] [교재명] [교사명]');  open vcursor;    loop  fetch vcursor into vname, vstartDate, vendDate, vbookname, vteachername;  exit when vcursor%notfound;        dbms\_output.put\_line(rpad(vname, 13, ' ') || ' ' || rpad(vstartDate || ' ~ ' || vendDate, 23, ' ') || ' ' || rpad(vbookname, 50, ' ') || ' ' || vteachername);  end loop;  close vcursor;    end;  /  **-- (개설과정명, 강의실명)**  create or replace procedure procSubOpen (  pseq in tblOpenClass.openSeq%type  )  is  vsubname tblClass.name%type;  vroomname tblClassRoom.name%type;    cursor vcursor  is  select  c.name,  r.name  from tblClass c  inner join tblOpenClass o  on c.courseSeq = o.courseSeq  inner join tblClassRoom r  on r.classRoomSeq = o.classRoomSeq  where o.courseSeq = pseq;  begin  dbms\_output.put\_line('———————————————————————————————————————————');  dbms\_output.put\_line(' 특정 개설 과정의 개설 과목 정보');  dbms\_output.put\_line('———————————————————————————————————————————' || CHR(10));  open vcursor;    loop  fetch vcursor into vsubname, vroomname;  exit when vcursor%notfound;        dbms\_output.put\_line('[과정명]: ' || vsubname);  dbms\_output.put\_line('[강의실명]: ' || vroomname || CHR(10));    end loop;  close vcursor;    end;  /  begin  procSubOpen(**<개설과정번호>**);  procSubInfo(**<개설과정번호>**);  end;  /  **-- 2. 개설 과목 별로 성적 등록 여부, 시험 문제 파일 등록**  create or replace procedure procCheck (  pseq in tblOpenSub.openSubSeq%type  )  is  vjname tblSubject.name%type;  vresult1 varchar(5);  vresult2 varchar(5);  vresult3 varchar(5);    cursor vcursor  is  select  distinct  j.name,  case  when g.stuSeq = s.stuSeq then 'Y'  else 'N'  end,  case  when w.testSeq is not null then 'Y'  else 'N'  end,  case  when p1.testSeq is not null then 'Y'  else 'N'  end  from tblOpenSub o  inner join tblGrade g  on o.openSubSeq = g.openSubSeq  inner join tblStudents s  on s.stuSeq = g.stuSeq  inner join tblClassOpen p  on p.openSubSeq = o.openSubSeq  inner join tblWTest w  on o.openSubSeq = w.openSubSeq  inner join tblPTest p1  on p1.openSubSeq = o.openSubSeq  inner join tblSubject j  on j.subSeq = o.subSeq  where o.openSubSeq = pseq;  begin  dbms\_output.put\_line('[과목명] [성적 등록 여부] [전체필기시험등록여부] [전체실기시험등록여부] ');    open vcursor;    loop  fetch vcursor into vjname, vresult1, vresult2, vresult3;  exit when vcursor%notfound;      dbms\_output.put\_line(vjname || ' ' || lpad(vresult1, 5, ' ') || ' ' || lpad(vresult2, 13, ' ') || ' ' || lpad(vresult3, 15, ' '));    end loop;  close vcursor;    end;  /  begin  procCheck(**<개설과목번호>**);  end;  /  **-- 3. 과목별 성적 출력[과정명, 과정기간, 강의실명, 과목명, 교사명, 교재명, 교육생 이름, 주민번호 뒷자리, 출결, 필기, 실기]**  **-- (과정명, 과정기간, 강의실명, 과목명, 교사명, 교재명)**  create or replace procedure procSubject (  pseq in tblOpenSub.openSubSeq%type  )  is  vjname tblClass.name%type;  vstartDate tblOpenClass.startDate%type;  vendDate tblOpenClass.endDate%type;  vroomname tblClassRoom.name%type;  vsname tblSubject.name%type;  vtname tblTeachers.name%type;  vbookname tblBook.bookName%type;    cursor vcursor  is  select  c2.name,  c1.startDate,  c1.endDate,  r.name,  s.name,  t.name,  b.bookName  from tblOpenSub o  inner join tblSubject s  on s.subSeq = o.subSeq  inner join tblClassOpen c  on c.openSubSeq = o.openSubSeq  inner join tblOpenClass c1  on c1.openSeq = c.openSeq  inner join tblClassRoom r  on r.classRoomSeq = c1.classRoomSeq  inner join tblClass c2  on c2.courseSeq = c1.courseSeq  inner join tblBook b  on b.bookSeq = s.bookSeq  inner join tblTeachers t  on t.teacherSeq = o.teacherSeq  where o.openSubSeq = pseq;  begin  dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 과목별 성적 출력');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));    open vcursor;    loop  fetch vcursor into vjname, vstartDate, vendDate, vroomname, vsname, vtname, vbookname;  exit when vcursor%notfound;      dbms\_output.put\_line('[과정명]: ' || vjname );  dbms\_output.put\_line('[과정기간]: ' || vstartDate || ' ~ ' || vendDate);  dbms\_output.put\_line('[강의실명]: ' || vroomname );  dbms\_output.put\_line('[과목명]: ' || vsname );  dbms\_output.put\_line('[교사명]: ' || vtname );  dbms\_output.put\_line('[교재명]: ' || vbookname || CHR(10));    end loop;  close vcursor;    end;  /  **-- (교육생 이름, 주민번호 뒷자리, 출결, 필기, 실기)**  create or replace procedure procScore (  pseq in tblOpenSub.openSubSeq%type  )  is  vstuname tblStudents.name%type;  vjumin tblStudents.jumin%type;  vattScore tblGrade.attScore%type;  vwriteScore tblGrade.writeScore%type;  vpracScore tblGrade.pracScore%type;    cursor vcursor  is  select  t.name,  t.jumin,  g.attScore,  g.writeScore,  g.pracScore  from tblOpenSub o  inner join tblGrade g  on g.openSubSeq = o.openSubSeq  inner join tblStudents t  on t.stuSeq = g.stuSeq  where o.openSubSeq = pseq  order by t.stuSeq asc;  begin    dbms\_output.put\_line('[교육생 이름] [주민번호 뒷자리] [출결] [필기] [실기]');    open vcursor;    loop  fetch vcursor into vstuname, vjumin, vattScore, vwriteScore, vpracScore;  exit when vcursor%notfound;      dbms\_output.put\_line(lpad(vstuname, 9, ' ') || ' ' || lpad(vjumin, 13, ' ') || ' ' || lpad(vattScore, 6, ' ') || ' ' || lpad(vwriteScore, 4, ' ') || ' ' || vpracScore);      end loop;  close vcursor;    end;  /  begin  procSubject(**<개설과목번호>**);  procScore(**<개설과목번호>**);  end;  /  **-- 4. 교육생 개인별 성적 출력[이름, 주민번호 뒷자리, 과정명, 과정기간, 강의실명, 과목명, 과목 기간, 교사명, 필기, 실기]**  **-- (이름, 주민번호 뒷자리, 과정명, 과정기간, 강의실명)**  create or replace procedure procStudent (  pseq in tblStudents.stuSeq%type  )  is  vstuname tblStudents.name%type;  vjumin tblStudents.jumin%type;  vcname tblClass.name%type;  vstartDate tblOpenClass.startDate%type;  vendDate tblOpenClass.endDate%type;  vroomname tblClassRoom.name%type;  cursor vcursor  is  select  distinct  t.name,  t.jumin,  s.name,  p.startDate,  p.endDate,  r.name  from tblStudents t  inner join tblGrade g  on t.stuSeq = g.stuSeq  inner join tblOpenSub o  on o.openSubSeq = g.openSubSeq  inner join tblClassOpen c  on c.openSubSeq = o.openSubSeq  inner join tblOpenClass p  on p.openSeq = c.openSeq  inner join tblClass s  on s.courseSeq = p.courseSeq  inner join tblClassRoom r  on r.classRoomSeq = p.classRoomSeq  where t.stuSeq = pseq;  begin  dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 교육생 개인별 성적 출력');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));    open vcursor;    loop  fetch vcursor into vstuname, vjumin, vcname, vstartDate, vendDate, vroomname;  exit when vcursor%notfound;      dbms\_output.put\_line('[교육생 이름]: ' || vstuname );  dbms\_output.put\_line('[주민번호 뒷자리]: ' || vjumin);  dbms\_output.put\_line('[과정명]: ' || vcname );  dbms\_output.put\_line('[과정기간]: ' || vstartDate || ' ~ ' || vendDate );  dbms\_output.put\_line('[강의실명]: ' || vroomname || CHR(10));    end loop;  close vcursor;    end;  /  **-- (교육생 이름, 주민번호 뒷자리, 출결, 필기, 실기)**  create or replace procedure procStuScore (  pseq in tblStudents.stuSeq%type  )  is  vjname tblSubject.name%type;  vstartDate tblOpenSub.startDate%type;  vendDate tblOpenSub.endDate%type;  vtname tblTeachers.name%type;  vattScore tblGrade.attScore%type;  vwriteScore tblGrade.writeScore%type;  vpracScore tblGrade.pracScore%type;    cursor vcursor  is  select  distinct  j.name,  o.startDate,  o.endDate,  t.name,  g.attScore,  g.writeScore,  g.pracScore  from tblOpenSub o  inner join tblSubject j  on j.subSeq = o.subSeq  inner join tblGrade g  on o.openSubSeq = g.openSubSeq  inner join tblTeachers t  on t.teacherSeq = o.teacherSeq  inner join tblStudents d  on d.stuSeq = g.stuSeq  where d.stuSeq = pseq  order by d.stuSeq asc;  begin    dbms\_output.put\_line('[과목명] [과목기간] [교사명] [출결] [필기] [실기]');    open vcursor;    loop  fetch vcursor into vjname, vstartDate, vendDate, vtname, vattScore, vwriteScore, vpracScore;  exit when vcursor%notfound;      dbms\_output.put\_line(rpad(vjname, 9, ' ') || ' ' || rpad(vstartDate || '~' || vendDate, 20 , ' ') || ' ' || vtname || ' ' || vattScore || ' ' || vwriteScore || ' ' || vpracScore);      end loop;  close vcursor;    end;  /  begin  procStudent(**<교육생번호>**);  procStuScore(**<교육생번호>**);  end;  /  **-- 5. 강의실별 재시험 교육생 출력[재시험 날짜, 과목명, 교육생 이름]**  create or replace procedure procReTest (  pseq in tblClassRoom.classRoomSeq%type  )  is  vreTest tblReTest.reTestDate%type;  vjname tblSubject.name%type;  vstuname tblStudents.name%type;    cursor vcursor  is  select  t.reTestDate,  j.name,  s.name  from tblOpenSub o  inner join tblSubject j  on o.subSeq = j.subSeq  inner join tblGrade g  on g.openSubSeq = o.openSubSeq  inner join tblStudents s  on s.stuSeq = g.stuSeq  inner join tblReTest t  on t.gradeSeq = g.gradeSeq  inner join tblClassOpen c  on c.openSubSeq = o.openSubSeq  inner join tblOpenClass c1  on c1.openSeq = c.openSeq  inner join tblClassRoom r  on r.classRoomSeq = c1.classRoomSeq  where r.classRoomSeq = pseq;  begin    dbms\_output.put\_line('[재시험날짜] [과목명] [교육생이름]');    open vcursor;    loop  fetch vcursor into vreTest, vjname, vstuname;  exit when vcursor%notfound;      dbms\_output.put\_line(rpad(vreTest, 15, ' ') || ' ' || rpad(vjname, 13, ' ') || ' ' || vstuname);    end loop;  close vcursor;    end;  /  begin  procReTest(**<강의실번호>**);  end;  / | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-08 | 작성일 | 2022-11-02 |
| 요구사항명 | 출결 관리 및 출결 조회 | 작성자 | 김하늘 |
| DML | | | |
| **-- 1. 기간별 출결 조회(교육생 이름, 날짜, 출결)**  create or replace procedure procAttendance(  pstart varchar2,  pend varchar2  )  is  vsname tblStudents.name%type;  vattdate varchar2(30);  vattendance tblAtt.attendance%type;  cursor vcursor  is  select  s.name,  to\_char(attdate, 'yyyy-mm-dd'),  a.attendance  from tblStudents s  inner join tblAtt a  on s.stuSeq = a.stuSeq  full outer join tblHoliday h  on h.regDate = a.attDate  where to\_char(a.attdate, 'd') <> '1' and to\_char(a.attdate, 'd') <> '7'  and h.name is null  and attdate between to\_date(pstart, 'yyyy-mm-dd') and to\_date(pend, 'yyyy-mm-dd')  order by to\_char(attdate, 'yyyy-mm-dd') asc;  begin    dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 기간별 출결 조회');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));    dbms\_output.put\_line('[교육생 이름] [날짜] [출결]');      open vcursor;    loop  fetch vcursor into vsname, vattdate, vattendance;  exit when vcursor%notfound;    dbms\_output.put\_line(lpad(vsname, 9, ' ') || ' ' || lpad(vattdate, 13, ' ') || ' ' || vattendance);    end loop;  close vcursor;  exception  when others then  dbms\_output.put\_line('예외 처리');    end;  /  begin  procAttendance('**<조회하는 시작년월일>**', '**<조회하는 끝년월일>**');  end;  /  **-- 2. 월별 출결 (정상, 지각, 조퇴, 외출, 병가, 기타)**  create or replace procedure procMonthAtt(  pstart varchar2,  pend varchar2,  pseq number  )  is  vattdate varchar2(30);  vresult1 number;  vresult2 number;  vresult3 number;  vresult4 number;  vresult5 number;  vresult6 number;  vresult7 number;  cursor vcursor  is  select  to\_char(attdate, 'mm'),  count(a.attendance),  count(case  when a.attendance = '정상' then '정상'  end),  count(case  when a.attendance = '지각' then '지각'  end),  count(case  when a.attendance = '조퇴' then '조퇴'  end),  count(case  when a.attendance = '외출' then '외출'  end),  count(case  when a.attendance = '병가' then '병가'  end),  count(case  when a.attendance = '기타' then '기타'  end)  from tblStudents s  inner join tblAtt a  on s.stuSeq = a.stuSeq  full outer join tblHoliday h  on h.regDate = a.attDate  where to\_char(a.attdate, 'd') <> '1' and to\_char(a.attdate, 'd') <> '7'  and attdate between to\_date(pstart, 'mm') and last\_day(to\_date(pend, 'mm')) and s.stuSeq = pseq  and h.name is null  group by to\_char(attdate, 'mm')  order by to\_char(attdate, 'mm') asc;  begin    dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 기간별 출결 조회');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));    dbms\_output.put\_line('[교육생 월별 출결] [출결] [정상] [지각] [조퇴] [외출] [병가] [결석]');      open vcursor;    loop  fetch vcursor into vattdate, vresult1, vresult2, vresult3, vresult4, vresult5, vresult6, vresult7;  exit when vcursor%notfound;    dbms\_output.put\_line(lpad(vattdate, 9, ' ') || ' ' || lpad(vresult1, 9, ' ') || ' ' || lpad(vresult2, 4, ' ') || ' ' || lpad(vresult3, 4, ' ') || ' ' || lpad(vresult4, 3, ' ') || ' ' || lpad(vresult5, 4, ' ') || ' ' || lpad(vresult6, 4, ' ') || ' ' || lpad(vresult7, 3, ' '));    end loop;  close vcursor;  exception  when others then  dbms\_output.put\_line('예외 처리');    end;  /  begin  procMonthAtt('**<조회하는 시작월>**', '**<조회하는 끝월>**', 1);  end;  /  **-- 3. 특정 인원 출결 현황을 조회**  create or replace procedure procStuAtt(  pseq tblStudents.stuSeq%type  )  is  vstuname tblStudents.name%type;  vattDate tblAtt.attDate%type;  vattendance tblAtt.attendance%type;  cursor vcursor  is  select  s.name,  a.attDate,  a.attendance  from tblAtt a  inner join tblStudents s  on a.stuSeq = s.stuSeq  full outer join tblHoliday h  on h.regDate = a.attDate  where to\_char(a.attdate, 'd') <> '1' and to\_char(a.attdate, 'd') <> '7'  and s.stuSeq = pseq  and h.name is null  order by a.attDate asc;  begin    dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 특정인원 출결 조회');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));    dbms\_output.put\_line('[교육생 이름] [날짜] [출결]');      open vcursor;    loop  fetch vcursor into vstuname, vattdate, vattendance;  exit when vcursor%notfound;      dbms\_output.put\_line(lpad(vstuname, 9, ' ') || ' ' || lpad(vattdate, 10, ' ') || ' ' || vattendance);    end loop;  close vcursor;  exception  when others then  dbms\_output.put\_line('예외 처리');    end;  /  begin  procStuAtt(**<교육생번호>**);  end;  /  **-- 4. 특정 개설과정 선택 시 모든 교육생의 출결을 조회**  create or replace procedure procAllStu(  pseq tblOpenClass.openSeq%type  )  is  vstuseq tblStudents.stuSeq%type;  vcoseq tblClass.courseSeq%type;  vstartDate tblOpenClass.startDate%type;  vendDate tblOpenClass.endDate%type;  vstuli tblOpenClass.stuLimit%type;  vstuname tblStudents.name%type;  vattDate tblAtt.attDate%type;  vattendance tblAtt.attendance%type;  cursor vcursor  is  select  distinct  t.stuSeq,  s.courseSeq,  p.startDate,  p.endDate,  p.stuLimit,  t.name,  a.attDate,  a.attendance  from tblAtt a  inner join tblStudents t  on t.stuSeq = a.stuSeq  inner join tblGrade g  on g.stuSeq = t.stuSeq  inner join tblOpenSUb o  on o.openSubSeq = g.openSubSeq  inner join tblClassOpen c  on c.openSubSeq = o.openSubSeq  inner join tblOpenClass p  on p.openSeq = c.openSeq  inner join tblClass s  on s.courseSeq = p.courseSeq  full outer join tblHoliday h  on h.regDate = a.attDate  where p.openSeq = pseq  and to\_char(a.attdate, 'd') <> '1' and to\_char(a.attdate, 'd') <> '7'  and h.name is null  order by attDate asc;  begin    dbms\_output.put\_line('———————————————————————————————————————————————————');  dbms\_output.put\_line(' 특정 개설과정의 모든 교육생의 출결');  dbms\_output.put\_line('———————————————————————————————————————————————————' || CHR(10));    dbms\_output.put\_line('[학생번호] [과정번호] [과정기간] [수강정원] [교육생이름] [날짜] [근태기록]');      open vcursor;    loop  fetch vcursor into vstuseq, vcoseq, vstartDate, vendDate, vstuli, vstuname, vattDate, vattendance;  exit when vcursor%notfound;    dbms\_output.put\_line(lpad(vstuseq, 7, ' ') || ' ' || lpad(vcoseq, 6, ' ') || ' ' || lpad(vstartDate || '~' || vendDate, 20 , ' ') || ' ' || lpad(vstuli, 5, ' ') || ' ' || lpad(vstuname, 11, ' ') || ' ' || lpad(vattDate, 11, ' ') || ' ' || vattendance);    end loop;  close vcursor;  exception  when others then  dbms\_output.put\_line('예외 처리');    end;  /  begin  procAllStu(**<개설과정번호>**);  end;  /  **--병가**  create or replace procedure ProcSickLeave(  psickLeave number  )  is  today date;  vatt varchar(30);  begin  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');      update tblAtt set attendance = '병가' where stuSeq = psickLeave and attdate = today;    end ProcSickLeave;  begin  ProcSickLeave(**<교육생번호>**);  end;  **--기타**  create or replace procedure ProcOthers(  pothers number  )  is  today date;  vatt varchar(30);  begin  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');      update tblAtt set attendance = '기타' where stuSeq = pothers and attdate = today;    end ProcOthers;  begin  ProcOthers(**<교육생번호>**);  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-09 | 작성일 | 2022-11-02 |
| 요구사항명 | 취업현황 | 작성자 | 김하늘 |
| DML | | | |
| **-- 1. 교육생별(교육생이름, 취업현황, 취업날짜, 회사명, 과정명)**  create or replace procedure procStuEmp (  pseq in tblStudentEmp.employmentSeq%type  )  is  vstuname tblStudents.name%type;  vemNow tblStudentEmp.employNow%type;  vemDate tblStudentEmp.employDate%type;  vcompany tblStudentEmp.company%type;  vcname tblClass.name%type;  cursor vcursor  is  select  s.name,  e.employNow,  e.employDate,  e.company,  c1.name  from tblStudents s  inner join tblStudentEmp e  on s.stuSeq = e.stuSeq  inner join tblOpenClass c  on c.openSeq = e.openSeq  inner join tblClass c1  on c1.courseSeq = c.courseSeq  where e.employmentSeq = pseq;  begin    dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 교육생 개인별 취업현황');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));      open vcursor;    loop  fetch vcursor into vstuname, vemNow, vemDate, vcompany, vcname;  exit when vcursor%notfound;      dbms\_output.put\_line('[교육생이름]: ' || vstuname);  dbms\_output.put\_line('[취업현황]: ' || vemNow);  dbms\_output.put\_line('[취업날짜]: ' || vemDate);  dbms\_output.put\_line('[회사명]: ' || vcompany);  dbms\_output.put\_line('[과정명]: ' || vcname);    end loop;  close vcursor;    end;  /  begin  procStuEmp(**<교육생취업번호>**);  end;  /  **-- 교육생 취업정보 수정하기**  create or replace procedure procEditEmp (  pseq tblStudents.stuSeq%type,  pnow tblStudentEmp.employNow%type,  paemdate tblStudentEmp.employDate%type,  pcompany tblStudentEmp.company%type,  pemseq tblStudentEmp.company%type  )  is  begin  update tblStudentEmp set employNow = pnow, employDate = paemDate, company = pcompany  where stuSeq = pseq and employmentSeq = pemseq;    dbms\_output.put\_line('수정을 완료했습니다.');    exception  when others then  dbms\_output.put\_line('예외처리');  end;  /  begin  procEditEmp(**<교육생 번호>**, **<수정할 취업현황>**, **<수정할 취업날짜>**, **<수정할 회사명>, <개설과정번호>**);  end;  /  **-- 2. 교사별(과정명, 과정을 들은 교육생 수, 취업을 한 교육생 수, 가르친 교육생의 취업률(%))**  create or replace procedure procTeacherEmp (  pseq in tblTeachers.teacherSeq%type  )  is  vtseq tblTeachers.teacherSeq%type;  vpseq tblOpenClass.openSeq%type;  vcname tblClass.name%type;  vclassCount number;  vemNowCount number;  vemNowPcount number;    cursor vcursor  is  select  distinct  ts.teacherSeq,  p.openSeq,  c.name,  count(\*),  count(case  when e.employNow = '취업' then '취업'  end),  round(count(case  when e.employNow = '취업' then '취업'  end) / count(\*) \* 100, 1)  from tblOpenClass p  inner join tblClass c  on p.courseSeq = c.courseSeq  inner join tblApply a  on a.openSeq = p.openSeq  inner join tblStudentEmp e  on e.openSeq = p.openSeq  inner join tblStudents t  on t.stuSeq = a.stuSeq  inner join tblClassOpen co  on co.openSeq = p.openSeq  inner join tblOpenSub o  on o.openSubSeq = co.openSubSeq  inner join tblTeachers ts  on ts.teacherSeq = o.teacherSeq  where ts.teacherSeq = pseq and endDate < sysdate  group by ts.teacherSeq, p.openSeq, c.name, t.stuSeq  order by ts.teacherSeq, p.openSeq asc;  begin    dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 교사별 교육생 취업현황');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));    dbms\_output.put\_line('[교사번호] [개설과정번호] [과정명] [과정을 들은 교육생 수] [취업을 한 교육생 수] [교육생의취업률(%)]');      open vcursor;    loop  fetch vcursor into vtseq, vpseq, vcname, vclassCount, vemNowCount, vemNowPcount;  exit when vcursor%notfound;      dbms\_output.put\_line(lpad(vtseq, 5, ' ') || ' ' || lpad(vpseq, 14, ' ') || ' ' || rpad(vcname, 75, ' ') || ' ' || rpad(vclassCount, 18, ' ') || ' ' || rpad(vemNowCount, 13, ' ') || ' ' || vemNowPcount);    end loop;  close vcursor;    end;  /  begin  procTeacherEmp(**<교사번호>**);  end;  /  **-- 3. 기간별 조회(년도, 취업을 한 교육생 수)**  create or replace procedure procEmp  is  vemDate varchar2(30);  vemNowCount varchar2(15);    cursor vcursor  is  select  to\_char(employDate, 'yyyy'),  count(\*)  from tblStudentEmp  group by to\_char(employDate, 'yyyy')  having to\_char(employDate, 'yyyy') is not null;  begin    dbms\_output.put\_line('——————————————————————————————');  dbms\_output.put\_line(' 년도별 취업현황');  dbms\_output.put\_line('——————————————————————————————' || CHR(10));    dbms\_output.put\_line('[취업년도] [취업한 교육생 수]');      open vcursor;    loop  fetch vcursor into vemDate, vemNowCount;  exit when vcursor%notfound;      dbms\_output.put\_line(lpad(vemDate, 7, ' ') || ' ' || lpad(vemNowCount, 13, ' '));    end loop;  close vcursor;    end;  /  begin  procEmp();  end;  / | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-10 | 작성일 | 2022-11-02 |
| 요구사항명 | 게시판 | 작성자 | 김하늘 |
| DML | | | |
| **-- 교육생, 교사는 게시판에 게시글을 등록할수 있고, 댓글을 작성할수있다.**  **create or replace procedure procTboard(**  pteacherseq number, -- 게시글을 올릴 선생님번호  pcontent varchar2 -- 게시내용  )  as  vseq number;  **begin(프로시저)**  select nvl(max(boardseq),0) +1 into vseq from tblboard ;  INSERT INTO tblBoard(boardSeq,content,postDate,teacherSeq,stuSeq) VALUES (vseq, pcontent, sysdate , pteacherseq ,null);  exception  when others then dbms\_output.put\_line('예외처리');  **end procTboard;**  **-- 교육생, 교사는 게시판에 게시글을 등록할수 있고, 댓글을 작성할수있다.(호출)**  **begin(호출)**  procTboard(**<게시할 교사의번호>**, ‘**<내용>**’);  **end;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 댓글등록(교사)**  **create or replace procedure procTcomment(**  pteacherseq number, - 댓글을 입력할 교사번호(교육생번호)  pboardseq number, - 댓글입력할 게시물번호  pcontent varchar2 - 댓글내용  )  as  vseq number;  **begin**  select nvl(max(commentseq),0) +1 into vseq from tblcomment;  INSERT INTO tblComment(commentSeq, content, boardSeq, teacherSeq, stuSeq) VALUES (vseq,pcontent,pboardseq,pteacherseq,null);  **end procTcomment;**  exec procTcomment(**<선생님번호>(교육생번호)**,**<게시판번호>**,**<게시내용>**);  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  --게시판 삭제  **create or replace procedure procBoardDelete(**  pteacherseq number,  pboardseq number  )  as  begin  delete from tblboard where teacherSeq = **<삭제할 게시물 교사의 번호>**  **end procBoardDelete;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 게시판 내용수정(프로시저)**  **create or replace procedure procmodiBoard(**  pteacherseq number,  pboardseq number,  pcontent varchar2  )  as  **begin**  update tblboard set content = pcontent where teacherseq = pteacherseq and boardseq = pboardseq;  **end procmodiBoard;**  **--게시내용수정(호출)**  **exec procmodiBoard(<작성자 번호>,<게시판 번호>,<수정할 게시판>);**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 게시판 조회(프로시저)**  declare  cursor vcursor  is  select \* from tblboard;  cursor vcursor2(pboardseq number)  is  select \* from tblcomment where boardseq = pboardseq;  i number;  begin  for vrow in vcursor loop  dbms\_output.put\_line(' ');  dbms\_output.put\_line('--------------------------------------------------------------------------------');  dbms\_output.put\_line('['||vrow.boardseq || '번 게시물]'||' [등록일:'||vrow.postdate||'] '||'[작성자번호(교사/학생): '|| vrow.teacherseq||'/'||vrow.stuseq||']');  dbms\_output.put\_line(vrow.content);  dbms\_output.put\_line(' ');  dbms\_output.put\_line(vrow.boardseq || '번 게시물의 댓글');  i := 1;  for vrow2 in vcursor2(vrow.boardseq) loop  if vrow2.content is not null then  dbms\_output.put\_line('['||i||']'||vrow2.content||'['||'작성자번호(교사/교육생): '||vrow2.teacherseq||'/'||vrow2.stuseq ||']');  else dbms\_output.put\_line('[게시물 댓글없음]');  end if;  i := i+1;  end loop;  end loop;    end;  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  --특정 게시판 보기  **create or replace procedure procSeeBoard(**  pboardseq number  )  as  cursor vcursor  is  select \* from tblcomment  where boardseq = pboardseq;  vrow tblboard%rowtype;  vrow2 tblcomment%rowtype;  i number;  begin  i := 1;  select \* into vrow from tblboard where boardseq = pboardseq;  dbms\_output.put\_line('['||vrow.boardseq||'번 게시물 내용]');    dbms\_output.put\_line(vrow.content);  dbms\_output.put\_line(' ');  dbms\_output.put\_line('['||pboardseq|| '번 게시물 댓글]');  open vcursor ;  loop  fetch vcursor into vrow2;  exit when vcursor%notfound;    dbms\_output.put\_line('['|| i ||']'||vrow2.content);  dbms\_output.put\_line(' ');  i := i + 1;  end loop;  close vcursor;  exception  when others then dbms\_output.put\_line('예외처리');  end procSeeBoard;  exec procSeeBoard(5);  -------------------------------------------------------------------------------------------------------------------------------------------------------------------------- | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | B-11 | 작성일 | 2022-11-02 |
| 요구사항명 | 강의 평가 | 작성자 | 김하늘 |
| DML | | | |
| **create or replace procedure procSeeEval(**  popenseq number -- 조회할 과정번호  )  as  cursor vcursor  is  select cl.name,st.name,ev.content from tblstudents st  inner join tblevaluation ev  on st.stuseq = ev.stuseq  inner join tblapply ap  on ap.stuseq = st.stuseq  inner join tblopenclass oc  on oc.openseq = ap.openseq  inner join tblclass cl  on cl.courseseq = oc.courseseq  where oc.openseq = popenseq;  vSubname tblclass.name%type;  vStuname tblstudents.name%type;  vcontent tblevaluation.content%type;  **begin**    open vcursor;  loop  fetch vcursor into vSubname, vStuname, vcontent;  exit when vcursor%notfound;  dbms\_output.put\_line('과정명: '||vSubname||', '||'수료한 학생이름: '||vStuname||', '||'과정평: '||vcontent);  end loop;  close vcursor;  **end procSeeEval;** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-02 | 작성일 | 2022-11-02 |
| 요구사항명 | 강의 스케줄 조회 기능 | 작성자 | 이한솔 |
| DML | | | |
| **강의 스케쥴 조회**  **과목번호, 과정명, 과정기간(시작 년월일 / 끝 년월일), 강의실, 과목명, 과목기간, 교재명, 교육생 등록 인원**  **- 교육생 정보 (교육생 이름, 전화번호, 등록일, 수료 또는 중도탈락)**  **- 강의 진행 상태 (날짜 기준)**   * **강의 목록**   create or replace procedure procSchedule (  vSeq in number,  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  s.subSeq,  c.name,  oc.startdate,  oc.enddate,  cr.name,  s.name,  os.startdate,  os.enddate,  b.bookName,  count(\*)  from tblteachers t  left outer join tblOpenSub os  on t.teacherSeq = os.teacherseq  inner join tblSubject s  on os.subSeq = s.subSeq  inner join tblClassSub cs  on s.subSeq = cs.subSeq  inner join tblClass c  on cs.courseSeq = c.courseSeq  inner join tblopenclass oc  on c.courseSeq = oc.courseSeq  inner join tblClassRoom cr  on oc.classroomSeq = cr.classroomseq  left outer join tblApply a  on oc.openSeq = a.openSeq  inner join tblBook b  on s.bookSeq = b.bookSeq  where t.teacherSeq = vSeq  group by a.openSeq, c.name, s.subSeq, cr.name, s.name, b.Bookname, oc.startdate, oc.enddate, os.startdate, os.enddate  order by s.subSeq;  end procSchedule;   * 프로시저 호출   declare  vcursor SYS\_REFCURSOR;  vSubSeq tblSubject.subSeq%type;  vClassName tblClass.name%type;  vClassStart tblOpenClass.startdate%type;  vClassEnd tblOpenClass.enddate%type;  vRoom tblClassRoom.name%type;  vSubName tblSubject.name%type;  vSubStart tblOpenSub.startdate%type;  vSubEnd tblOpenSub.enddate%type;  vBook tblBook.bookname%type;  vCount number;  begin  procSchedule(**교사번호**, vcursor);  loop  fetch vcursor into vsubSeq, vClassName, vClassStart, vClassEnd, vRoom, vSubName, vSubStart, vSubEnd, vBook, vCount;  exit when vcursor%notfound;    dbms\_output.put\_line('과목번호 : ' || vsubSeq || ' | ' || '과정명 : ' || vClassName || ' | ' || '과정기간 : ' || vClassStart || ' - ' || vClassEnd || ' | ' || '강의실 : ' || vRoom || ' ' ||  '과목명 : ' || vSubName || ' | ' || '과목기간 : ' || vSubStart || ' - ' || vSubEnd || ' | ' || '교재명 : ' || vBook || ' | ' || '교육생 등록 인원 : ' || vCount);  end loop;  close vcursor;  end;   * **교육생 정보 (교육생 이름, 전화번호, 등록일, 수료 또는 중도탈락)**   create or replace procedure procStu (  vSeq in number,  pcursor out sys\_refcursor  )  is begin  open pcursor  for  select  st.name,  st.tel,  st.regdate,  st.enddate,  st.fail  from tblteachers t inner join tblopensub os on os.teacherSeq = t.teacherseq  inner join tblclassopen co  on co.opensubseq = os.opensubseq  inner join tblopenclass oc  on oc.openseq = co.openseq  inner join tblapply a  on a.openseq = oc.openseq  inner join tblstudents st  on st.stuseq = a.stuseq  where t.teacherseq = vseq  group by st.name, st.tel, st.regdate, st.enddate, st.fail;  end procStu;   * **프로시저 호출**   declare  vcursor SYS\_REFCURSOR;  vName tblStudents.name%type;  vtel tblStudents.tel%type;  vsign date;  vend date;  vfail tblStudents.fail%type;  begin  procStu(**교사번호**, vcursor);    loop  fetch vcursor into vName, vtel, vsign, vend, vfail;  exit when vcursor%notfound;    if vfail is null then    if vend is null then  dbms\_output.put\_line('교육생 이름 : ' || vName || ' | ' ||  '전화번호 : ' || vtel || ' | ' ||  '등록일 : ' || vsign || ' | ' ||  '수료일 : ' || 'null' );  else  dbms\_output.put\_line('교육생 이름 : ' || vName || ' | ' ||  '전화번호 : ' || vtel || ' | ' ||  '등록일 : ' || vsign || ' | ' ||  '수료일 : ' || vend );  end if;    else  dbms\_output.put\_line('교육생 이름 : ' || vName || ' | ' ||  '전화번호 : ' || vtel || ' | ' ||  '등록일 : ' || vsign || ' | ' ||  '중도탈락여부 : ' || vfail);  end if;  end loop;  close vcursor;  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-03 | 작성일 | 2022-11-02 |
| 요구사항명 | 배점 입출력 기능 | 작성자 | 이한솔 |
| DML | | | |
| **배점 입출력**  **1. 강의를 마친 과목의 목록 > 특정과목 > 배점 정보( 출결, 필기, 실기로 구분)**  **시험날짜, 시험문제 추가**  **2. 과목 목록 출력**  **- 과목번호, 과정명, 과정기간, 강의실, 과목명, 과목기간, 교재명, 출결, 필기, 실기 배점**  **3. 특정 과목번호 선택시**  **출결 배점, 필기 배점, 실기 배점, 시험날짜, 시험문제**  **입력 화면**  **-- 배점 입력 (출결, 필기, 실기)**  **-- 배점 등록 안된 과목은 null로 출력**  create or replace procedure procInsPoint (  vnum number,  vatt tblpoint.attendance%type,  vwrite tblpoint.write%type,  vpract tblpoint.practice%type  )  is  begin  update tblpoint set attendance = vatt, write = vwrite, practice = vpract where pointseq = vnum;  end procInsPoint;   * **프로시저 호출**   begin  procInsPoint(**배점번호, 출결배점, 필기배점, 실기배점**);  end;   * **시험날짜 (필기)**   create or replace procedure procWExam (  vos number,  vtdate date  )  is  vts number;  begin  select w.testseq into vts from tblopensub os right outer join tblwtest w on os.opensubseq = w.opensubseq where os.opensubseq = vos;    update tblwtest set testdate = vtdate where testseq = vts;  end procWExam;   * **프로시저 호출**   begin  procWExam(**과목번호**, **시험날짜**);  end;  select \* from tblwtest where opensubseq = 1;   * **시험문제 (필기)**   create or replace procedure procWQuestion (  vos number,  vqnum number,  vquestion tblwriteQ.question%type  )  is  vqs number;  vts number;  begin    select w.testseq into vts from tblopensub os right outer join tblwtest w on os.opensubseq = w.opensubseq where os.opensubseq = vos;    vqs := writeSeq.nextVal;  insert into tblwriteq values (vqs, vqnum, vquestion, vts);  end procWQuestion;   * **프로시저 호출**   begin  procWQuestion(**과목번호, 문제번호, 문제내용**);  end;  select \* from tblwriteq where testseq = 1;   * **시험날짜 (실기)**   create or replace procedure procPExam (  vos number,  vtdate date  )  is  vts number;  begin  select p.testseq into vts from tblopensub os right outer join tblptest p on os.opensubseq = p.opensubseq where os.opensubseq = vos;    update tblptest set testdate = vtdate where testseq = vts;  end procPExam;   * **프로시저 호출**   begin  procPExam(**과목번호, 시험날짜**);  end;  select \* from tblptest where opensubseq = 1;   * **시험문제 (실기)**   create or replace procedure procPQuestion (  vos number,  vqnum number,  vquestion tblpracticeQ.question%type  )  is  vqs number;  vts number;  begin    select p.testseq into vts from tblopensub os right outer join tblptest p on os.opensubseq = p.opensubseq where os.opensubseq = vos;    vqs := PractSeq.nextVal;  insert into tblpracticeq values (vqs, vqnum, vquestion, vts);  end procPQuestion;   * **프로시저 호출**   begin  procPQuestion(**과목번호, 문제번호, 문제내용**);  end;  select \* from tblpracticeq where testseq = 1; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | C-04 | 작성일 | 2022-11-02 |
| 요구사항명 | 성적 입출력 기능 | 작성자 | 이한솔 |
| DML | | | |
| **성적 입출력**  **1. 강의를 마친 과목의 목록 > 특정 과목 선택 > 교육생 정보**  **> 교육생 정보 선택 > 해당 교육생의 시험 점수 입력**  **출결, 필기, 실기 점수 구분 입력**  **2. 과목 목록 출력**  **- 과목번호 | 과정명 | 과정기간 | 강의실 | 과목명 | 과목기간 | 교재명 | 출결 / 필기 / 실기 배점**  **3. 특정 과목 선택**  **- 교육생 정보**  **- 이름 | 전화번호 | 수료 / 중도탈락 | 출결 / 필기 / 실기 점수 | 과락 여부**   * **강의 마친 과목 목록**   create or replace procedure procEndSub (  vSeq in number,  pcursor out sys\_refcursor  )  is  begin  open pcursor  for  select  s.subSeq,  c.name,  oc.startdate,  oc.enddate,  cr.name,  s.name,  os.startdate,  os.enddate,  b.bookName,  count(\*),  p.attendance,  p.write,  p.practice  from tblteachers t  inner join tblOpenSub os  on t.teacherSeq = os.teacherseq  inner join tblSubject s  on os.subSeq = s.subSeq  inner join tblClassSub cs  on s.subSeq = cs.subSeq  inner join tblClass c  on cs.courseSeq = c.courseSeq  inner join tblopenclass oc  on c.courseSeq = oc.courseSeq  inner join tblClassRoom cr  on oc.classroomSeq = cr.classroomseq  inner join tblApply a  on oc.openSeq = a.openSeq  inner join tblBook b  on s.bookSeq = b.bookSeq  inner join tblpoint p  on p.pointseq = os.pointseq  where t.teacherSeq = vSeq and os.enddate < sysdate  group by a.openSeq, c.name, s.subSeq, cr.name, s.name, b.Bookname, oc.startdate, oc.enddate, os.startdate, os.enddate, p.attendance, p.write, p.practice  order by s.subSeq;  end procEndSub;   * **프로시저 호출**   declare  vcursor SYS\_REFCURSOR;  vSubSeq tblSubject.subSeq%type;  vClassName tblClass.name%type;  vClassStart tblOpenClass.startdate%type;  vClassEnd tblOpenClass.enddate%type;  vRoom tblClassRoom.name%type;  vSubName tblSubject.name%type;  vSubStart tblOpenSub.startdate%type;  vSubEnd tblOpenSub.enddate%type;  vBook tblBook.bookname%type;  vCount number;  vAtt number;  vWrite number;  vPract number;  begin  procEndSub(**교사번호**, vcursor);  loop  fetch vcursor into vsubSeq, vClassName, vClassStart, vClassEnd, vRoom, vSubName, vSubStart, vSubEnd, vBook, vCount, vAtt, vWrite, vPract;  exit when vcursor%notfound;    dbms\_output.put\_line('과목번호 : ' || vsubSeq || ' | ' || '과정명 : ' || vClassName || ' | ' || '과정기간 : ' || vClassStart || ' - ' || vClassEnd || ' | ' || '강의실 : ' || vRoom || ' ' ||  '과목명 : ' || vSubName || ' | ' || '과목기간 : ' || vSubStart || ' - ' || vSubEnd || ' | ' || '교재명 : ' || vBook || ' | ' || '교육생 등록 인원 : ' || vCount || ' | ' ||  '출결 : ' || vAtt || ' | ' || '필기 : ' || vWrite || ' | ' || '실기 : ' || vPract );  end loop;  close vcursor;  end;  **-- 특정 과목 선택시 교육생 정보 출력**  **-- 이름 | 전화번호 | 수료 / 중도탈락 | 출결 / 필기 / 실기 점수 | 과락 여부**  create or replace procedure procScore (  vSeq in number,  vos number,  pcursor out sys\_refcursor  )  is begin  open pcursor  for  select  st.name,  st.tel,  st.enddate,  st.fail,  g.attscore,  g.writescore,  g.pracscore,  g.retest,  f.faildate  from tblteachers t  inner join tblOpenSub os  on t.teacherSeq = os.teacherseq  inner join tblSubject s  on os.subSeq = s.subSeq  inner join tblClassSub cs  on s.subSeq = cs.subSeq  inner join tblClass c  on cs.courseSeq = c.courseSeq  inner join tblopenclass oc  on c.courseSeq = oc.courseSeq  inner join tblClassRoom cr  on oc.classroomSeq = cr.classroomseq  inner join tblApply a  on oc.openSeq = a.openSeq  inner join tblStudents st  on st.stuSeq = a.stuSeq  right outer join tblgrade g  on g.stuseq = st.stuseq  left outer join tblfail f  on st.stuseq = f.stuseq  where t.teacherSeq = vSeq and os.opensubseq = vos;  end procScore;   * **프로시저 호출**   declare  vcursor SYS\_REFCURSOR;  vName tblStudents.name%type;  vtel tblStudents.tel%type;  vend date;  vfail tblStudents.fail%type;  vatt number;  vwrite number;  vpract number;  vretest tblgrade.retest%type;  vfdate tblfail.faildate%type;  begin  procScore(**교사번호, 과목번호**, vcursor);    loop  fetch vcursor into vName, vtel, vend, vfail, vatt, vwrite, vpract, vretest, vfdate;  exit when vcursor%notfound;    if vfail is null then    if vend is null then  dbms\_output.put\_line('교육생 이름 : ' || vName || ' | ' ||  '전화번호 : ' || vtel || ' | ' ||  '중도탈락여부 : ' || 'null' || ' | ' ||  '수료일 : ' || 'null' || ' | '||  '출결 : ' || vatt || ' | ' ||  '필기 : ' || vwrite || ' | ' ||  '실기 : ' || vpract || ' | ' ||  '과락여부 : ' || vretest || ' | ' );  else  dbms\_output.put\_line('교육생 이름 : ' || vName || ' | ' ||  '전화번호 : ' || vtel || ' | ' ||  '중도탈락여부 : ' || 'null' || ' | ' ||  '수료일 : ' || vend || ' | '||  '출결 : ' || vatt || ' | ' ||  '필기 : ' || vwrite || ' | ' ||  '실기 : ' || vpract || ' | ' ||  '과락여부 : ' || vretest || ' | ' );  end if;    else  dbms\_output.put\_line('교육생 이름 : ' || vName || ' | ' ||  '전화번호 : ' || vtel || ' | ' ||  '중도탈락여부 : ' || vfail || ' | ' ||  '중도탈락일 : ' || vfdate || ' | ' ||  '수료일 : ' || 'null' || ' | '||  '출결 : ' || 'null' || ' | ' ||  '필기 : ' || 'null' || ' | ' ||  '실기 : ' || 'null' || ' | ' ||  '과락여부 : ' || 'null' || ' | ' );  end if;  end loop;  close vcursor;  end;   * **성적 입력**   create or replace procedure procInsScore (  vos number,  vss number,  vatt number,  vwrite number,  vprac number  )  is  begin    if vwrite < 60 or vprac < 60 then  insert into tblgrade (gradeseq, attscore, writescore, pracscore, retest, opensubseq, stuseq) values (gradeseq.nextVal, vatt, vwrite, vprac, '불합격', vos, vss);  else  insert into tblgrade (gradeseq, attscore, writescore, pracscore, retest, opensubseq, stuseq) values (gradeseq.nextVal, vatt, vwrite, vprac, '통과', vos, vss);  end if;    end procInsScore;   * **프로시저 호출**   begin  procInsScore(과목번호, 학생번호, 출결점수, 필기점수, 실기점수);  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | **C-05** | 작성일 | 2022-11-01 |
| 요구사항명 | 출결 관리 및 출결 조회 | 작성자 | 유동환 |
| DML | | | |
| **-- 선생님이 담당한 학생들 출결출력**  **create or replace procedure procTeacherAtt (**  pteacherseq in number,  pcursor out sys\_refcursor  )  is  **begin**  open pcursor  for  select tech.name, st.name ,at.attdate ,at.attendance  from tblstudents st  inner join tblatt at  on st.stuseq = at.stuseq  inner join tblgrade gr  on st.stuseq = gr.stuseq  inner join tblopensub os  on gr.opensubseq = os.opensubseq  inner join tblteachers tech  on os.teacherseq = tech.teacherseq  where tech.teacherseq = pteacherseq  order by st.stuseq  ;  **end procTeacherAtt;**  **-- 선생님이 담당한 학생들 출결출력(호출)**  declare  vcursor sys\_refcursor;  vTname tblteachers.name%type;  vStame tblstudents.name%type;  vAttdate tblatt.attdate%type;  vAttendance tblatt.attendance%type;  **begin**  procTeacherAtt(1,vcursor);  loop  fetch vcursor into vTname, vStame,vAttdate,vAttendance;  exit when vcursor%notfound;  dbms\_output.put\_line('교사이름: '||vTname|| ', ' ||'학생이름:'||vStame|| ', ' ||'출결날짜:'||vAttdate|| ', ' ||'근태: '||vAttendance);  end loop;  **end;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 기간별 출별현황 출력(프로시저)**  **create or replace procedure procSinceAtt(**  pbegin date, **-- 시작날짜**  pend date **-- 끝날짜**  )  as  cursor vcursor  is  select st.stuseq,at.attdate,at.attendance from tblatt at inner join tblstudents st on at.stuseq = at.stuseq where at.attdate between pbegin and pend;  --커서선언  vStuseq number;  vAttdate date;  vAttendance varchar2(20);  **begin**  open vcursor;  loop  fetch vcursor into vStuseq, vAttdate, vAttendance;  exit when vcursor%notfound;  dbms\_output.put\_line( '학생번호: '||vStuseq|| ' ' || '출결날짜: '|| vattdate || '근태: ' || vattendance);  end loop;  close vcursor;  exception  when STORAGE\_ERROR then  dbms\_output.put\_line('메모리 초과');  when others then  dbms\_output.put\_line('예외 처리');    **end procSinceAtt;**  **-- 기간별 출별현황 출력(호출부)**  **begin**  procSinceAtt(**<시작날짜>,<끝날짜>**);  **end;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **--특정인원 출결확인(프로시저)**  **create or replace procedure procSinceStu**(  pstuseq number, --특정학생번호  pstuname varchar2 --특정학생 이름  )  as  vname tblstudents.name%type;  vattdate tblatt.attdate%type;  vattendance tblatt.attendance%type;  cursor vcursor  is  select st.name, at.attdate, at.attendance  from tblstudents st  inner join tblatt at  on st.stuseq = at.stuseq  where st.stuseq = pstuseq and st.name = pstuname  order by st.stuseq, at.attdate;  **begin(프로시저)**    open vcursor;  loop  fetch vcursor into vname, vattdate, vattendance;  exit when vcursor%notfound;  dbms\_output.put\_line('교육생이름: '||vname || ' '|| '출결날짜: ' || vattdate || ' 근태: '||vattendance );  end loop;  close vcursor;    exception when others  then  dbms\_output.put\_line('예외처리');  **end procSinceStu;**  **begin(호출)**  procSinceStu(**<특정학생번호>**, **<특정학생이름>**);  **end;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **--특정과정 출결조회(프로시저)**  **create or replace procedure procSinceClass(**  pOpenseq number, --특정과정 번호  presult out sys\_refcursor  )  as  **begin(프로시저)**  open presult  for  select oc.openseq, st.stuseq, st.name, at.attdate, at.attendance from tblstudents st  inner join tblatt at  on st.stuseq = at.stuseq  inner join tblapply ap  on st.stuseq = ap.stuseq  inner join tblopenclass oc  on oc.openseq = ap.openseq  inner join tblclass cl  on cl.courseseq = oc.courseseq  where oc.openseq = <조회할 과정번호>  order by st.stuseq, at.attdate;  exception  when others then  dbms\_output.put\_line('예외처리');  **end procSinceClass;**  **--특정과정 출결조회(호출)**  declare  vresult sys\_refcursor; **-- 커서**  vOpenseq tblopenclass.openseq%type;  **-- 과정번호**  vClassname tblclass.name%type;  **-- 과정명**  vStuseq tblstudents.stuseq%type; **-- 학생번호**  vStuname tblstudents.name%type;  **-- 학생이름**  vAttdate tblatt.attdate%type;  **-- 출결날짜**  vAttendance tblatt.attendance%type; **--근태**  **begin(호출)**  procSinceClass(**<조회하고싶은 과정번호>**,vresult);  loop  fetch vresult into vOpenseq, vStuseq, vStuname, vAttdate, vAttendance;  exit when vresult%notfound;  dbms\_output.put\_line('과정번호: '||vOpenseq ||' '||'학생번호: '|| vStuseq || ' 학생이름: '||vStuname || ' 출결날짜: '|| vAttdate ||'근태: '|| vAttendance);  end loop;  **end;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 근태 상황을 구분(프로시저)**  **create or replace procedure procAttCnt(**  pstuseq number  )  as  cursor vresult  is  select at.attendance, count(\*) from tblatt at where stuseq = **<조회할 학생의 번호>** group by at.attendance having at.attendance in ('병가','정상','기타','외출','지각','주말');    vAttendance tblatt.attendance%type;  vcnt number;  vname varchar2(20);  **begin(프로시저)**  select name into vname from tblstudents where stuseq = pstuseq;  dbms\_output.put\_line(' '||vname||'의 근태' );  dbms\_output.put\_line('[근태]'||' '||'[횟수]');  dbms\_output.put\_line('------------------------------');  open vresult;  loop  fetch vresult into vAttendance, vcnt;  dbms\_output.put\_line(vAttendance||' '||vcnt||'번');  exit when vresult%notfound;      end loop;  close vresult;  exception when others then dbms\_output.put\_line('예외처리');  **end procAttCnt;**  --**근태 상황을 구분(호출)**  **begin(호출)**  procAttCnt(**<조회하고 싶은 학생번호>**);  **end;** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | **C-06** | 작성일 | 2022-11-01 |
| 요구사항명 | 게시판 | 작성자 | 유동환 |
| DML | | | |
| **-- 교육생, 교사는 게시판에 게시글을 등록할수 있고, 댓글을 작성할수있다.**  **create or replace procedure procTboard(**  pteacherseq number, -- 게시글을 올릴 선생님번호  pcontent varchar2 -- 게시내용  )  as  vseq number;  **begin(프로시저)**  select nvl(max(boardseq),0) +1 into vseq from tblboard ;  INSERT INTO tblBoard(boardSeq,content,postDate,teacherSeq,stuSeq) VALUES (vseq, pcontent, sysdate , pteacherseq ,null);  exception  when others then dbms\_output.put\_line('예외처리');  **end procTboard;**  **-- 교육생, 교사는 게시판에 게시글을 등록할수 있고, 댓글을 작성할수있다.(호출)**  **begin(호출)**  procTboard(**<게시할 교사의번호>**, ‘**<내용>**’);  **end;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 댓글등록(교사)**  **create or replace procedure procTcomment(**  pteacherseq number, - 댓글을 입력할 교사번호(교육생번호)  pboardseq number, - 댓글입력할 게시물번호  pcontent varchar2 - 댓글내용  )  as  vseq number;  **begin**  select nvl(max(commentseq),0) +1 into vseq from tblcomment;  INSERT INTO tblComment(commentSeq, content, boardSeq, teacherSeq, stuSeq) VALUES (vseq,pcontent,pboardseq,pteacherseq,null);  **end procTcomment;**  exec procTcomment(**<선생님번호>(교육생번호)**,**<게시판번호>**,**<게시내용>**);  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  --게시판 삭제  **create or replace procedure procBoardDelete(**  pteacherseq number,  pboardseq number  )  as  begin  delete from tblboard where teacherSeq = **<삭제할 게시물 교사의 번호>**  **end procBoardDelete;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 게시판 내용수정(프로시저)**  **create or replace procedure procmodiBoard(**  pteacherseq number,  pboardseq number,  pcontent varchar2  )  as  **begin**  update tblboard set content = pcontent where teacherseq = pteacherseq and boardseq = pboardseq;  **end procmodiBoard;**  **--게시내용수정(호출)**  **exec procmodiBoard(<작성자 번호>,<게시판 번호>,<수정할 게시판>);**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 게시판 조회(프로시저)**  declare  cursor vcursor  is  select \* from tblboard;  cursor vcursor2(pboardseq number)  is  select \* from tblcomment where boardseq = pboardseq;  i number;  begin  for vrow in vcursor loop  dbms\_output.put\_line(' ');  dbms\_output.put\_line('--------------------------------------------------------------------------------');  dbms\_output.put\_line('['||vrow.boardseq || '번 게시물]'||' [등록일:'||vrow.postdate||'] '||'[작성자번호(교사/학생): '|| vrow.teacherseq||'/'||vrow.stuseq||']');  dbms\_output.put\_line(vrow.content);  dbms\_output.put\_line(' ');  dbms\_output.put\_line(vrow.boardseq || '번 게시물의 댓글');  i := 1;  for vrow2 in vcursor2(vrow.boardseq) loop  if vrow2.content is not null then  dbms\_output.put\_line('['||i||']'||vrow2.content||'['||'작성자번호(교사/교육생): '||vrow2.teacherseq||'/'||vrow2.stuseq ||']');  else dbms\_output.put\_line('[게시물 댓글없음]');  end if;  i := i+1;  end loop;  end loop;    end;  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  --특정 게시판 보기  **create or replace procedure procSeeBoard(**  pboardseq number  )  as  cursor vcursor  is  select \* from tblcomment  where boardseq = pboardseq;  vrow tblboard%rowtype;  vrow2 tblcomment%rowtype;  i number;  begin  i := 1;  select \* into vrow from tblboard where boardseq = pboardseq;  dbms\_output.put\_line('['||vrow.boardseq||'번 게시물 내용]');    dbms\_output.put\_line(vrow.content);  dbms\_output.put\_line(' ');  dbms\_output.put\_line('['||pboardseq|| '번 게시물 댓글]');  open vcursor ;  loop  fetch vcursor into vrow2;  exit when vcursor%notfound;    dbms\_output.put\_line('['|| i ||']'||vrow2.content);  dbms\_output.put\_line(' ');  i := i + 1;  end loop;  close vcursor;  exception  when others then dbms\_output.put\_line('예외처리');  end procSeeBoard;  exec procSeeBoard(5);  -------------------------------------------------------------------------------------------------------------------------------------------------------------------------- | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | **C-07** | 작성일 | 2022-11-01 |
| 요구사항명 | 강의 평가 | 작성자 | 유동환 |
| DML | | | |
| **create or replace procedure procSeeEval(**  popenseq number -- 조회할 과정번호  )  as  cursor vcursor  is  select cl.name,st.name,ev.content from tblstudents st  inner join tblevaluation ev  on st.stuseq = ev.stuseq  inner join tblapply ap  on ap.stuseq = st.stuseq  inner join tblopenclass oc  on oc.openseq = ap.openseq  inner join tblclass cl  on cl.courseseq = oc.courseseq  where oc.openseq = popenseq;  vSubname tblclass.name%type;  vStuname tblstudents.name%type;  vcontent tblevaluation.content%type;  **begin**    open vcursor;  loop  fetch vcursor into vSubname, vStuname, vcontent;  exit when vcursor%notfound;  dbms\_output.put\_line('과정명: '||vSubname||', '||'수료한 학생이름: '||vStuname||', '||'과정평: '||vcontent);  end loop;  close vcursor;  **end procSeeEval;**  **--과정평 조회**  exec procSeeEval(<조회할 과정번호>);  -------------------------------------------------------------------------------------------------------------------------------------------------------------------------- | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | **D-01** | 작성일 | 2022-11-01 |
| 요구사항명 | 성적조회 | 작성자 | 장진광 |
| DML | | | |
| **--교육생 개인의 정보와 수강 정보를 출력**  **--성적 정보는 과목별 목록형태로 출력된다.**  **--성적조회**  create or replace procedure pstuSco(  ppw varchar2  )  is  vsNum tblStudents.stuSeq%type;  vsjNum tblSubject.subSeq%type;  vjumin tblStudents.jumin%type;  vsjName tblSubject.name%type;  vstart tblOpenSub.startDate%type;  vend tblOpenSub.endDate%type;  vbookName tblBook.bookName%type;  vtName tblTeachers.name%type;  vattPoint tblPoint.attendance%type;  vwritePoint tblPoint.write%type;  vpracticePoint tblPoint.practice%type;  vaScore tblGrade.attScore%type;  vwScore tblGrade.writeScore%type;  vpScore tblGrade.pracScore%type;    cursor vcursor is  select  s.stuSeq,  sj.subSeq ,  s.jumin,  sj.name,  os.startDate,  os.endDate,  b.bookname,  t.name,  p.attendance,  p.write,  p.practice,  g.attScore,  g.writeScore,  g.pracScore  from tblStudents s  inner join tblGrade g  on s.stuSeq = g.stuSeq  inner join tblOpenSub os  on os.openSubSeq = g.openSubSeq  inner join tblTeachers t  on t.teacherSeq = os.teacherSeq  inner join tblPoint p  on p.pointSeq = os.pointSeq  inner join tblSubject sj  on sj.subSeq = os.subSeq  inner join tblWTest wt  on os.openSubSeq = wt.openSubSeq  inner join tblPTest pt  on os.openSubSeq = pt.openSubSeq  inner join tblBook b  on b.bookSeq = sj.bookSeq  where s.jumin = ppw;    begin  open vcursor;  loop  fetch vcursor into vsNum, vsjNum , vjumin, vsjName, vstart , vend, vbookName , vtName, vattPoint, vwritePoint, vpracticePoint , vaScore, vwScore, vpScore;  exit when vcursor%notfound;  dbms\_output.put\_line('-------------------------------------------------------------------------------------------------------------------------------------------------------------');  dbms\_output.put\_line('교육생번호 : ' || vsNum || ' 과목 이름 : ' || vsjName || ' 과목기간 : ' ||vstart|| '~' ||vend|| ' 교재이름: ' || vbookName || ' 교사명: ' || vtName );  dbms\_output.put\_line('출석배점 : ' || vattPoint|| '필기배점 : ' ||vwritePoint|| '실기배점 : ' ||vpracticePoint );  dbms\_output.put\_line('출석점수 : ' || vaScore|| '필기점수 : ' || vwScore|| '실기점수 : ' ||vpScore );  dbms\_output.put\_line('--------------------------------------------------------------------------------------------------------------------------------------------------------------');  dbms\_output.put\_line('');  end loop;  close vcursor;      exception  when others then  dbms\_output.put\_line('존재하지않는 계정입니다.');  end; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | **D-02** | 작성일 | 2022-11-01 |
| 요구사항명 | 출결관리 | 작성자 | 장진광 |
| DML | | | |
| **-- 입실**  create or replace procedure ProcCheckin(  pcheckin number  )  is  today date;  begi  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');  if ( TO\_CHAR(sysdate, 'HH24:MI' )  BETWEEN '09:00' AND '09:10' ) then  update tblAtt set attendance = '정상' where stuSeq = pcheckin and attdate = today;  elsif ( TO\_CHAR(sysdate, 'HH24:MI' )  BETWEEN '09:11' AND '17:50' ) then  update tblAtt set attendance = '지각' where stuSeq = pcheckin and attdate = today;  else  update tblAtt set attendance = '결석' where stuSeq = pcheckin and attdate = today;  end if;  end ProcCheckin;  **--외출**  create or replace procedure ProcGoOut(  pgoOut number  )  is  today date;  vatt varchar(30);  begin  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');  select attendance into vatt from tblAtt where stuSeq = pgoOut and attdate = today;  if vatt like '정상' or vatt like ‘지각’ then  update tblAtt set attendance = '외출' where stuSeq = pgoOut and attdate = today;  else  null;  end if;  exception  when others then  dbms\_output.put\_line('외출이 불가능합니다.');  end ProcGoOut;  **--조퇴**  create or replace procedure ProcEarlyLeave(  pearlyLeave number  )  is  today date;  vatt varchar(30);  begin  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');  select attendance into vatt from tblAtt where stuSeq = pearlyLeave and attdate = today;  if vatt like '정상' or vatt like ‘지각’ then  update tblAtt set attendance = '조퇴' where stuSeq = pearlyLeave and attdate = today;  else  null;  end if;  exception  when others then  dbms\_output.put\_line('조퇴가 불가능합니다.');  end ProcEarlyLeave;  **--병가**  create or replace procedure ProcSickLeave(  psickLeave number  )  is  today date;  vatt varchar(30);  begin  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');  update tblAtt set attendance = '병가' where stuSeq = psickLeave and attdate = today;  end ProcSickLeave;  **--기타..**  create or replace procedure ProcOthers(  pothers number  )  is  today date;  vatt varchar(30);  begin  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');  update tblAtt set attendance = '기타' where stuSeq = pothers and attdate = today;  end ProcOthers;  **-- 퇴실**  create or replace procedure ProcCheckOut(  pcheckout number  )  is  today date;  begin  today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');    if ( TO\_CHAR(sysdate, 'HH24:MI' )  BETWEEN '17:50' AND '18:00' ) then    update tblAtt set attendance = '정상' where stuSeq = pcheckout and attdate = today and attendance in ('정상','외출');    elsif ( TO\_CHAR(sysdate, 'HH24:MI' )  BETWEEN '18:01' AND '23:59' ) then    update tblAtt set attendance = '지각' where stuSeq = pcheckout and attdate = today and attendance in('정상', '외출', '지각');      end if;    end ProcCheckOut; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | **D-03** | 작성일 | 2022-11-01 |
| 요구사항명 | 강의평가 | 작성자 | 장진광 |
| DML | | | |
| **- 과정을 마친 교육생의 경우에만 과정에 대한 평가를 입력할 수 있다.**  **- 중도 탈락된 교육생의 경우 과정에 대한 평가를 입력할 수 없다.**  **--강의평가 등록**  create or replace procedure ProcEva(  peva number,  pcontent VARCHAR2  )  is  maxSeq number;  evaDay date;  today date;  failcheck varchar2(50);  begin    select fail into failcheck from tblStudents s where s.stuSeq = peva;    select nvl(max(evaluationSEQ),0) +1 into maxSeq from tblEvaluation;    select o.endDate into evaDay  from tblOpenClass o  inner join tblApply a  on o.openSeq = a.openSeq  inner join tblStudents s  on s.stuSeq = a.stuSeq  where s.stuSeq = peva;    today := TO\_CHAR(SYSDATE, 'YYYY/MM/DD');  if failcheck is null then  if TO\_CHAR(sysdate, 'yyyy/mm/dd') >= to\_char(evaDay, 'yyyy/mm/dd') then    insert into tblEvaluation values (maxSeq, pcontent, peva) ;    else  null;  end if;  end if;    exception  when others then  dbms\_output.put\_line('아직 강의평가 남기기가 불가능합니다.');    end ProcEva;    **--강의평가 삭제**  create or replace procedure ProcEvaD(  peva number  )  is  maxSeq number;  begin      delete from tblEvaluation where stuSeq = peva;      exception  when others then  dbms\_output.put\_line('강의평가 삭제가 불가능합니다.');    end ProcEvaD; | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 요구사항번호 | **D-04** | 작성일 | 2022-11-01 |
| 요구사항명 | 게시판 | 작성자 | 유동환 |
| DML | | | |
| **-- 교육생, 교사는 게시판에 게시글을 등록할수 있고, 댓글을 작성할수있다.**  **create or replace procedure procTboardStu(**  pstuseq number,  pcontent varchar2 -- 게시내용  )  as  vseq number;  **begin(프로시저)**  select nvl(max(boardseq),0) +1 into vseq from tblboard ;  INSERT INTO tblBoard(boardSeq,content,postDate,teacherSeq,stuSeq) VALUES (vseq, pcontent, sysdate , null, pstuseq);  exception  when others then dbms\_output.put\_line('예외처리');  **end procTboardStu;**  **-- 교육생, 교사는 게시판에 게시글을 등록할수 있고, 댓글을 작성할수있다.(호출)**  **begin(호출)**  procTboardStu(**<게시할 교육생의 번호>**, ‘**<내용>**’);  **end;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 댓글등록**  **create or replace procedure procTcommentStu(**  pstuseq number, - 댓글을 입력할 교육생번호  pboardseq number, - 댓글입력할 게시물번호  pcontent varchar2 - 댓글내용  )  as  vseq number;  **begin**  select nvl(max(commentseq),0) +1 into vseq from tblcomment;  INSERT INTO tblComment(commentSeq, content, boardSeq, teacherSeq, stuSeq) VALUES (vseq,pcontent,pboardseq,null,pstuseq);  **end procTcommentStu;**  exec procTcommentStu(**<교육생번호>)**,**<게시판번호>**,**<게시내용>**);  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  --게시판 삭제  **create or replace procedure procBoardDeleteStu(**  pstuseq number,  pboardseq number  )  as  begin  delete from tblboard where stuSeq = **<삭제할 게시물 교육생의 번호>**  **end procBoardDeleteStu;**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 게시판 내용수정(프로시저)**  **create or replace procedure procmodiBoardStu(**  pstuseq number,  pboardseq number,  pcontent varchar2  )  as  **begin**  update tblboard set content = pcontent where stuseq = pstuseq and boardseq = pboardseq;  **end procmodiBoardStu;**  **--게시내용수정(호출)**  **exec procmodiBoardStu(<작성자 번호>,<게시판 번호>,<내용>);**  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  **-- 게시판 조회(프로시저)**  declare  cursor vcursor  is  select \* from tblboard;  cursor vcursor2(pboardseq number)  is  select \* from tblcomment where boardseq = pboardseq;  i number;  begin  for vrow in vcursor loop  dbms\_output.put\_line(' ');  dbms\_output.put\_line('--------------------------------------------------------------------------------');  dbms\_output.put\_line('['||vrow.boardseq || '번 게시물]'||' [등록일:'||vrow.postdate||'] '||'[작성자번호(교사/학생): '|| vrow.teacherseq||'/'||vrow.stuseq||']');  dbms\_output.put\_line(vrow.content);  dbms\_output.put\_line(' ');  dbms\_output.put\_line(vrow.boardseq || '번 게시물의 댓글');  i := 1;  for vrow2 in vcursor2(vrow.boardseq) loop  if vrow2.content is not null then  dbms\_output.put\_line('['||i||']'||vrow2.content||'['||'작성자번호(교사/교육생): '||vrow2.teacherseq||'/'||vrow2.stuseq ||']');  else dbms\_output.put\_line('[게시물 댓글없음]');  end if;  i := i+1;  end loop;  end loop;    end;  --------------------------------------------------------------------------------------------------------------------------------------------------------------------------  --특정 게시판 보기  **create or replace procedure procSeeBoard(**  pboardseq number  )  as  cursor vcursor  is  select \* from tblcomment  where boardseq = pboardseq;  vrow tblboard%rowtype;  vrow2 tblcomment%rowtype;  i number;  begin  i := 1;  select \* into vrow from tblboard where boardseq = pboardseq;  dbms\_output.put\_line('['||vrow.boardseq||'번 게시물 내용]');    dbms\_output.put\_line(vrow.content);  dbms\_output.put\_line(' ');  dbms\_output.put\_line('['||pboardseq|| '번 게시물 댓글]');  open vcursor ;  loop  fetch vcursor into vrow2;  exit when vcursor%notfound;    dbms\_output.put\_line('['|| i ||']'||vrow2.content);  dbms\_output.put\_line(' ');  i := i + 1;  end loop;  close vcursor;  exception  when others then dbms\_output.put\_line('예외처리');  end procSeeBoard;  exec procSeeBoard(5);  -------------------------------------------------------------------------------------------------------------------------------------------------------------------------- | | | |