화면 및 관련 쿼리

#### Teacher

View

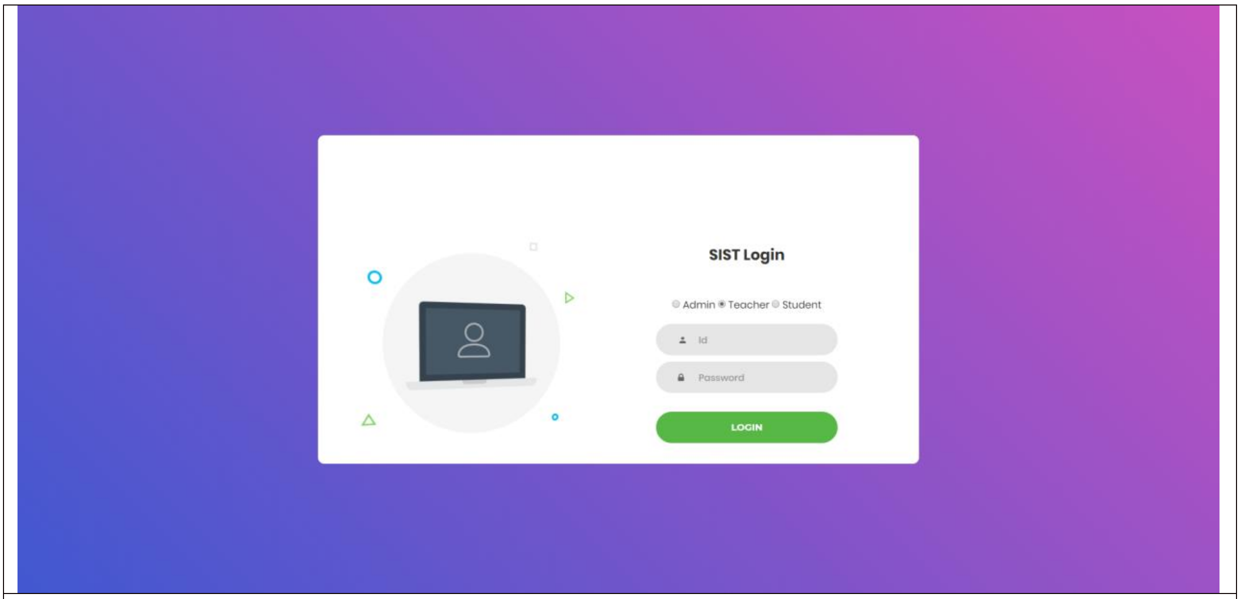
```
#teacher_view1
CREATE OR REPLACE VIEW teacher_view1
AS
SELECT oc.opencourse_id, co.course_name
    , oc.start_opencourse, oc.end_opencourse
    , cr.classroom_name,(CASE WHEN NOW() < start_opencourse  THEN '강의예정'
                            WHEN NOW()> end_opencourse THEN '강의종료'
                            ELSE '강의중'END) AS opencourse
    ,(SELECT count(*)  FROM course_history ch WHERE oc.opencourse_id = opencourse_id ) student_count
    FROM open_course oc, courses co, classrooms cr
    WHERE  oc.course_id = co.course_id
        AND oc.classroom_id = cr.classroom_id;

#teacher_view2
CREATE OR REPLACE VIEW teacher_view2
AS
SELECT os.openSubject_id, s.subject_name, os.start_opensubject
    , os.end_opensubject, book_id, oc.opencourse_id
    FROM open_subject os, subjects s, open_course oc
    WHERE os.subject_id = s.subject_id
    AND os.opencourse_id = oc.opencourse_id;

#teacher_view3
CREATE OR REPLACE VIEW teacher_view3
AS
SELECT os.opensubject_id, s.subject_name, os.start_opensubject
        , os.end_opensubject, os.book_id, oc.opencourse_id
    , oc.start_openCourse, oc.end_openCourse
        FROM open_subject os, subjects s ,open_course oc
        WHERE os.subject_id = s.subject_id
        AND oc.opencourse_id = os.opencourse_id
        ORDER BY opensubject_id;
```

```
#teacher_view3_2
CREATE OR REPLACE VIEW teacher_view3_2
AS
SELECT v.opensubject_id, subject_name, start_opensubject
        , end_opensubject, book_id, opencourse_id, e.exam_id, e.exam_date, e.exam_file_name
    , start_openCourse, end_openCourse
    FROM teacher_view3 v LEFT JOIN exams e
    ON v.opensubject_id = e.opensubject_id
    ORDER BY opensubject_id;

#teacher_view3_3
CREATE OR REPLACE VIEW teacher_view3_3
AS
SELECT v.opensubject_id, subject_name, start_opensubject
        , end_opensubject, book_id, opencourse_id, v.exam_id, exam_date, exam_file_name
    , s.attendance_points, s.written_points, s.practice_points
    , start_openCourse, end_openCourse, s.subjectPoints_id
    FROM teacher_view3_2 v LEFT JOIN subject_score s
    ON v.exam_id = s.exam_id
    ORDER BY opensubject_id;

#teacher_view4_1
CREATE OR REPLACE VIEW teacher_view4_1
AS
 SELECT ch.student_id, ch.opencourse_id, stop_date
    FROM course_history ch left join completion_course cc
    on (ch.student_id=cc.student_id AND ch.opencourse_id=cc.opencourse_id)
    ORDER BY ch.student_id;

#teacher_view4_2
CREATE OR REPLACE VIEW teacher_view4_2
AS
SELECT t.student_id, opencourse_id, stop_date
        , student_name, student_phone, student_regDate
        FROM  teacher_view4_1 t LEFT JOIN students s ON t.student_id=s.student_id ORDER B
Y t.student_id;

#teacher_view4_3
```

```
CREATE OR REPLACE VIEW teacher_view4_3
AS
SELECT t.student_id, t.opencourse_id, stop_date
        , student_name, student_phone, student_regDate
    , (CASE
    WHEN NOW() < end_opencourse AND stop_date IS NULL THEN '진행중'
    WHEN NOW() > end_opencourse AND stop_date IS NULL THEN '수료완료'
    WHEN stop_date IS NOT NULL THEN '중도탈락'
    END)AS completion_status
        , (CASE
    WHEN NOW() < end_opencourse AND stop_date IS NULL THEN null
    WHEN NOW() > end_opencourse AND stop_date IS NULL THEN end_opencourse
    WHEN NOW() IS NOT NULL THEN stop_date
    END)AS completion_date
    FROM teacher_view4_2 t LEFT JOIN open_course oc ON t.opencourse_id=oc.opencourse_id
    ORDER BY t.student_id;

#teacher_view4_4
CREATE OR REPLACE VIEW teacher_view4_4
AS
SELECT t.student_id, t.opencourse_id, stop_date
        , student_name, student_phone, student_regDate
    , completion_status,completion_date, openSubject_id, teacher_id
    FROM teacher_view4_3 t LEFT JOIN open_subject os on t.opencourse_id=os.opencourse_id
    ORDER BY t.student_id;

#teacher_view4_5
CREATE OR REPLACE VIEW teacher_view4_5
AS
SELECT t.student_id, t.opencourse_id, stop_date
        , student_name, student_phone, student_regDate
    , completion_status, completion_date, exam_id, t.openSubject_id, exam_date, exam_file_name
    FROM teacher_view4_4 t LEFT JOIN exams ex ON t.opensubject_id=ex.opensubject_id
    ORDER BY t.student_id;

#teacher_view4_6
CREATE OR REPLACE VIEW teacher_view4_6
AS
```

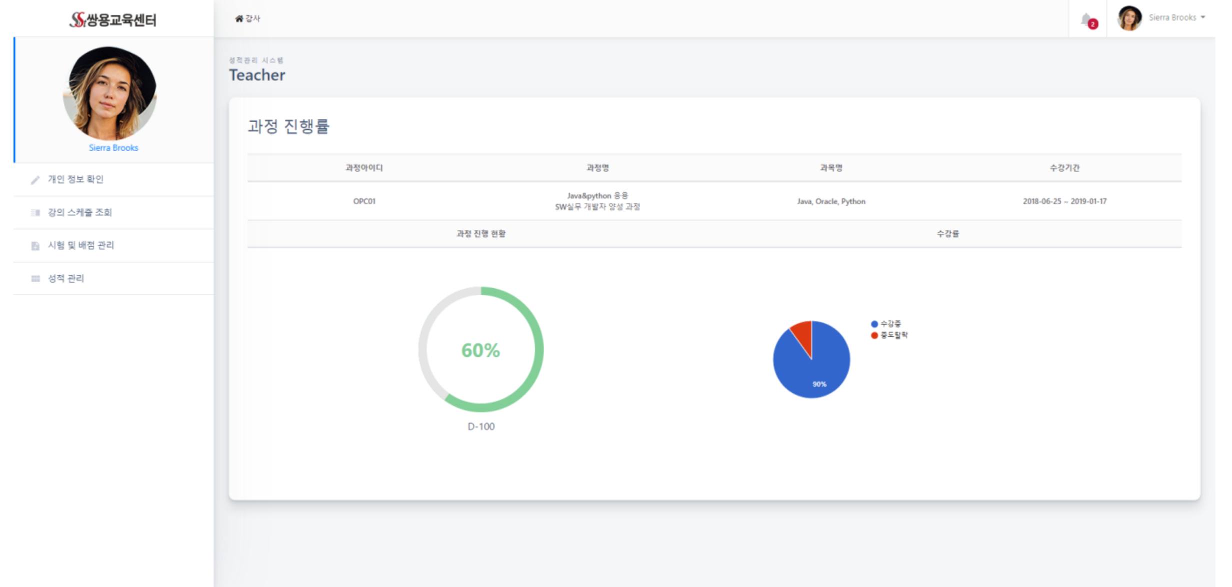
```
SELECT t.student_id, opencourse_id, stop_date, student_name, student_phone, student_regDate
          ,completion_status, completion_date, t.exam_id, opensubject_id, exam_date, exam_file_n
ame
     , attendance_score, written_score, practice_score
     , (attendance_score + written_score + practice_score)AS total_score
FROM teacher_view4_5 t LEFT JOIN student_grade sg ON (t.student_id=sg.student_id AND t.ex
am_id=sg.exam_id);
```

Teacher login



```
#로그인
SELECT teacher_id
     FROM teachers
     WHERE teacher_name = '김선생' AND teacher_pw = '강사pw';
```

Teacher_0



과정 진행률

과정아이디	과정명	과목명	수강기간
OPC01	Java&python 응용 SW실무 개발자 양성 과정	Java, Oracle, Python	2018-06-25 ~ 2019-01-17

과정 진행 현황	수강률

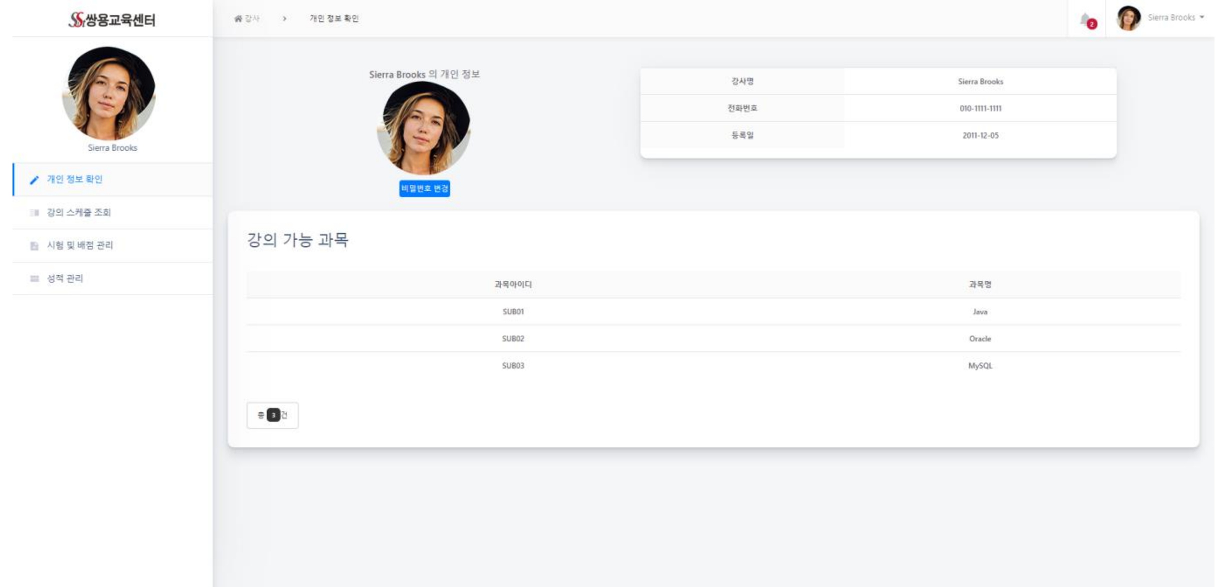
```
#과정 진행률
SELECT v.opencourse_id, course_name
        , group_concat(subject_name SEPARATOR ', ') AS subject_name
    , concat(start_opencourse,'~', end_opencourse) period, end_opencourse
    ,classroom_name, opencourse, student_count
        FROM teacher_view1 v, open_subject os, subjects s
    WHERE v.openCourse_id = os.openCourse_id
        AND s.subject_id = os.subject_id
    AND opencourse = '강의중'
    AND v.opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = '강사아이디');


#수강률 그래프
SELECT COUNT(CASE WHEN NOW() < oc.end_opencourse AND stop_date IS NULL THEN '진행중' END) as 'completion_status1'
        , COUNT(CASE WHEN stop_date IS NOT NULL THEN '중도탈락' END) as 'completion_status2'
    , COUNT(CASE WHEN NOW() > oc.end_opencourse AND stop_date IS NULL THEN '수료완료' END) as 'completion_status3'
FROM students s LEFT JOIN completion_course cc ON s.student_id = cc.student_id
        , open_course oc, course_history ch_, teacher_view1 v
        WHERE  ch_.student_id=s.student_id
    AND v.openCourse_id = oc.openCourse_id
    AND oc.opencourse_id=ch_.opencourse_id
```

```
          AND v.opencourse = '강의중'
          AND ch_.opencourse_id IN (SELECT os.opencourse_id
                                      FROM teachers ins, open_subject os
                                      WHERE ins.teacher_id = os.teacher_id
                                      AND os.teacher_id = '강사아이디');
```

#과정 진행률

```
SELECT (TO_DAYS(now())-TO_DAYS(oc.start_openCourse))/(TO_DAYS(oc.end_openCourse)-TO_DAYS(
oc.start_openCourse))*100 AS course_date
FROM open_course oc, teacher_view1 v
          WHERE v.openCourse_id = oc.openCourse_id
          AND v.opencourse = '강의중'
          AND v.opencourse_id IN (SELECT os.opencourse_id
                                      FROM teachers ins, open_subject os
                                      WHERE ins.teacher_id = os.teacher_id
                                      AND os.teacher_id = '강사아이디');
```

Teacher_1



강사명	Sierra Brooks
전화번호	010-1111-1111
등록일	2011-12-05

강의 가능 과목

과목아이디	과목명
SUB01	Java
SUB02	Oracle
SUB03	MySQL

총 3 건

#개인 정보

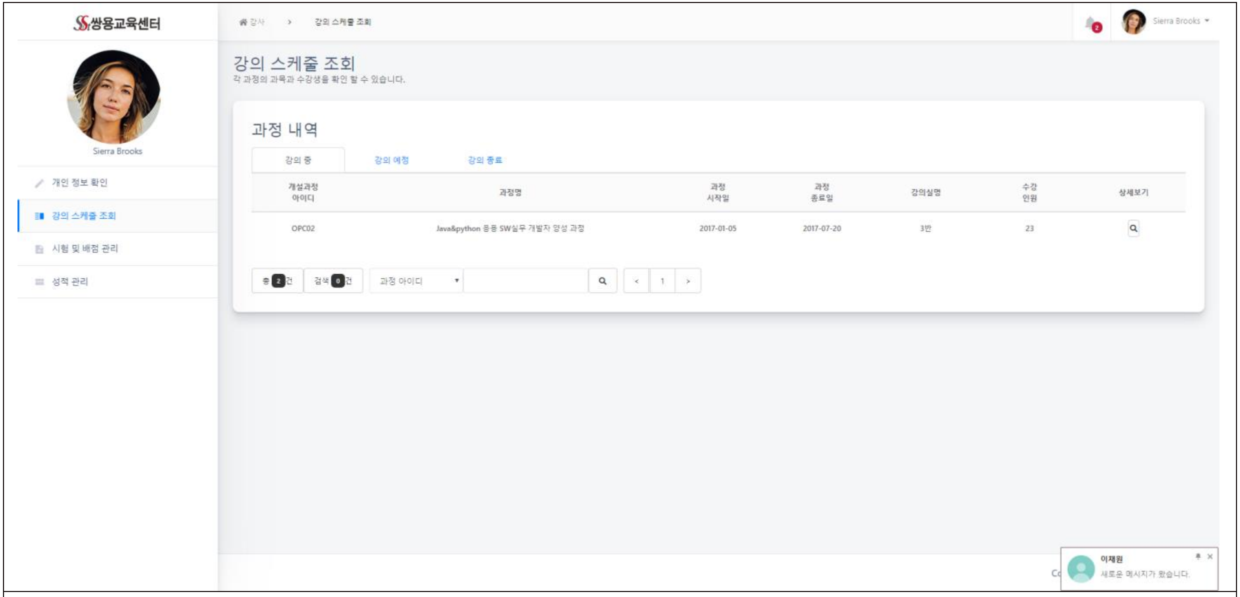
```
SELECT teacher_name, teacher_phone, teacher_regdate
     FROM teachers
     WHERE teacher_id = '강사아이디';
```

#강의 가능 과목

```
SELECT s.subject_id, s.subject_name
```

```
    FROM subjects s, available_subjects as_
    WHERE s.subject_id = as_.subject_id
    AND as_.teacher_id = '강사아이디'
    ORDER BY subject_id;

#비밀번호 변경
UPDATE teachers
        SET teacher_pw = '새로운pw'
    WHERE teacher_id = ''
    AND teacher_pw = '기존pw';
```

Teacher_2_1



강의 스케줄 조회

각 과정의 과목과 수강생을 확인 할 수 있습니다.

과정 내역

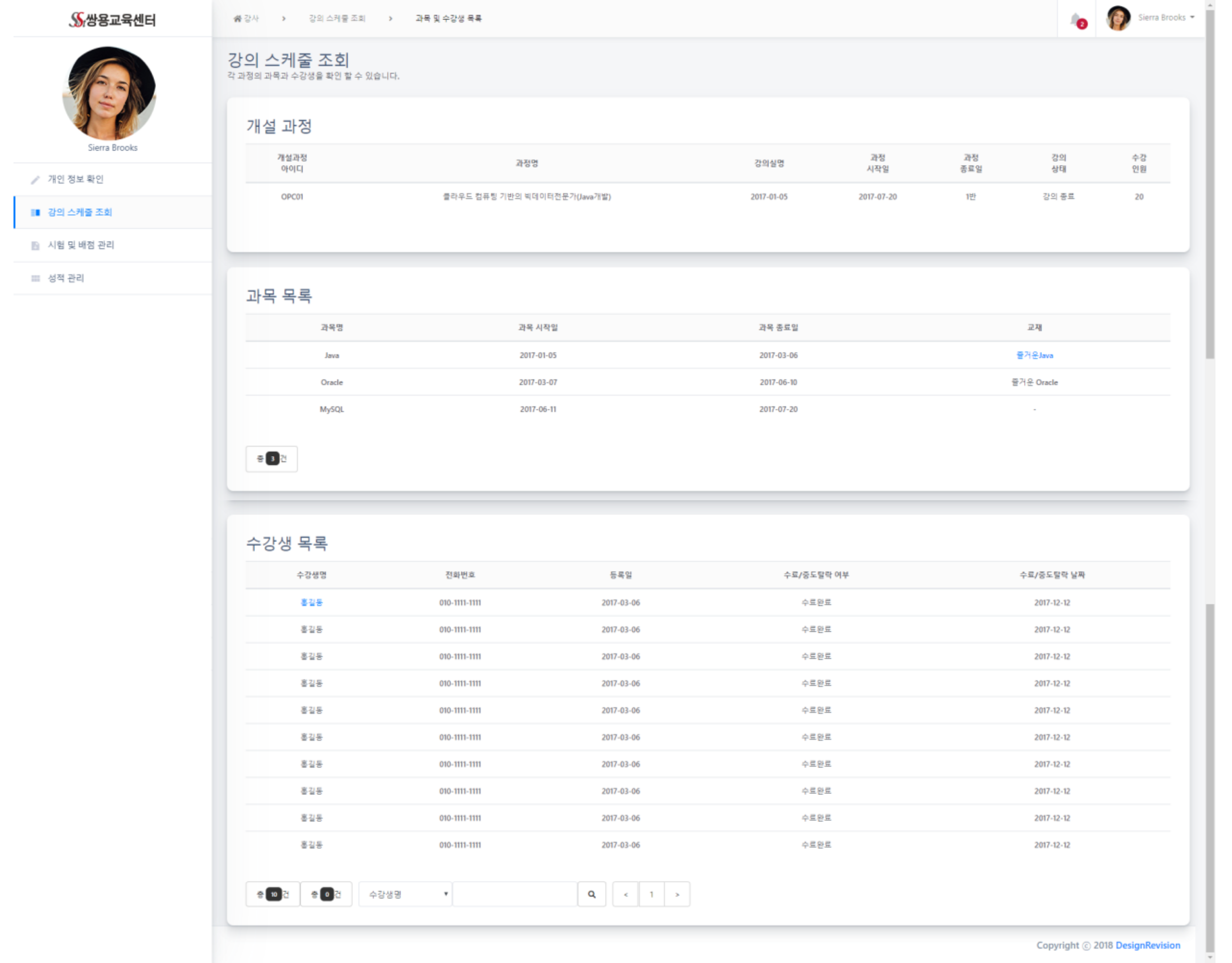
개설과정 아이디	과정명	과정 시작일	과정 종료일	강의실명	수강 인원	상세보기
OPC02	Java&python 응용 SW실무 개발자 양성 과정	2017-01-05	2017-07-20	3반	23	🔍

이재원

새로운 메시지가 왔습니다.

```
#teacher_view1
CREATE OR REPLACE VIEW teacher_view1
AS
SELECT oc.opencourse_id, co.course_name
    , oc.start_opencourse, oc.end_opencourse
    , cr.classroom_name,(CASE WHEN NOW() < start_opencourse  THEN '강의예정'
                              WHEN NOW()> end_opencourse THEN '강의종료'
                              ELSE '강의중'END) AS opencourse
    ,(SELECT count(*)  FROM course_history ch WHERE oc.opencourse_id = opencourse_id ) student_count
    FROM open_course oc, courses co, classrooms cr
    WHERE  oc.course_id = co.course_id
        AND oc.classroom_id = cr.classroom_id;

#과정내역 - 강의중, 강의예정, 강의종료
SELECT opencourse_id, course_name
    , start_opencourse, end_opencourse
    ,classroom_name, opencourse, student_count
                                                  FROM teacher_view1
    WHERE opencourse = '강의중'
    AND opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = 'T01')
        LIMIT 0, 10;
```

Teacher_2_2



강의 스케쥴 조회

각 과정의 과목과 수강생을 확인 할 수 있습니다.

개설 과정

개설과정 아이디	과정명	강의실명	과정 시작일	과정 종료일	강의 상태	수강 인원
OPC01	클라우드 컴퓨팅 기반의 빅데이터전문가(Java개발)	2017-01-05	2017-07-20	1반	강의 종료	20

과목 목록

과목명	과목 시작일	과목 종료일	교재
Java	2017-01-05	2017-03-06	즐거운Java
Oracle	2017-03-07	2017-06-10	즐거운 Oracle
MySQL	2017-06-11	2017-07-20	-

총 3 건

수강생 목록

수강생명	전화번호	등록일	수료/중도탈락 여부	수료/중도탈락 날짜
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12
홍길동	010-1111-1111	2017-03-06	수료완료	2017-12-12

Copyright © 2018 DesignRevision

#개설과정

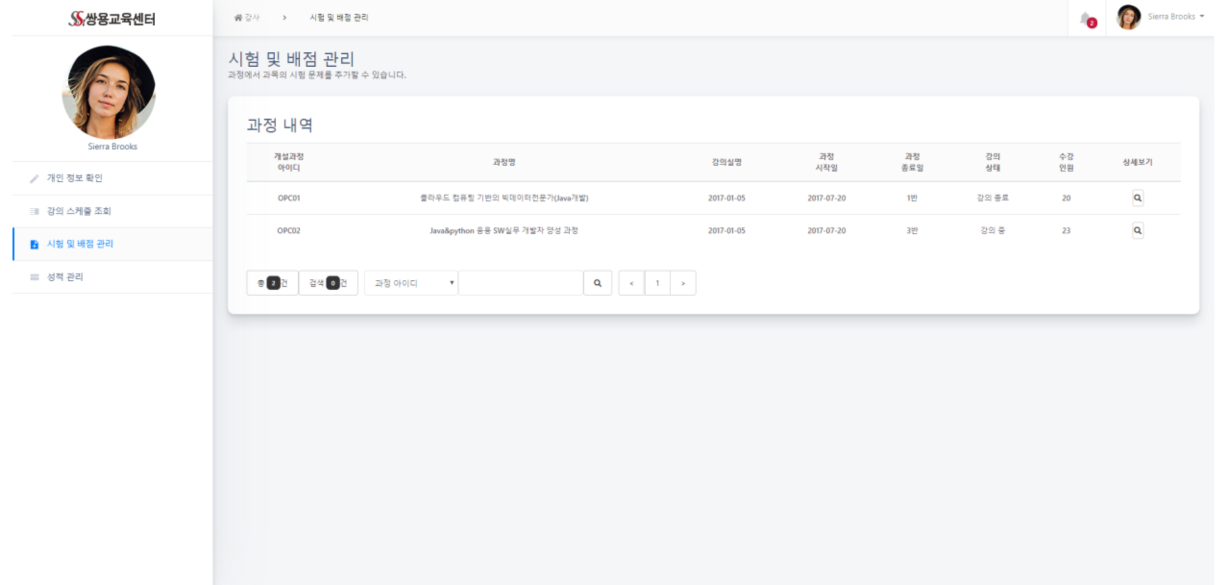
```
SELECT opencourse_id, course_name
    , start_opencourse, end_opencourse
    ,classroom_name, opencourse, student_count
        FROM teacher_view1
    WHERE opencourse_id = 'OPC07'
        AND opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = '강사아이디');
```

#teacher_view2

```
CREATE OR REPLACE VIEW teacher_view2
AS
SELECT os.openSubject_id, s.subject_name, os.start_opensubject
    , os.end_opensubject, book_id, oc.opencourse_id
    FROM open_subject os, subjects s, open_course oc
    WHERE os.subject_id = s.subject_id
    AND os.opencourse_id = oc.opencourse_id;

#과목목록
SELECT openSubject_id, subject_name, start_opensubject
    , end_opensubject, b.book_name, b.isbn
    FROM teacher_view2 v LEFT JOIN books b
    ON v.book_id = b.book_id
    WHERE v.openCourse_id = ‘개설과정아이디'
    AND v.opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = '강사아이디');

#구현시 examsDAO print2()로 3_2페이지꺼랑 같은 쿼리 사용함
SELECT opensubject_id, subject_name, start_opensubject
        , end_opensubject, b.book_name, b.isbn, attendance_points
        , written_points, practice_points, exam_date, exam_file_name, exam_id
        , (SELECT COUNT(*) FROM exams
         WHERE opensubject_id = v.opensubject_id) examCount
FROM teacher_view3_3 v LEFT JOIN books b
   ON v.book_id = b.book_id
   WHERE v.openCourse_id = '개설과정아이디'
   AND v.opencourse_id IN (SELECT os.opencourse_id
                  FROM teachers ins, open_subject os
                  WHERE ins.teacher_id = os.teacher_id
                  AND os.teacher_id = '강사아이디')
                  LIMIT 0, 10;
#수강생 목록
SELECT s.student_id, student_name, student_phone, student_regDate,
    (CASE
    WHEN NOW() < end_opencourse AND stop_date IS NULL THEN '진행중'
    WHEN NOW() > end_opencourse AND stop_date IS NULL THEN '수료완료'
    WHEN stop_date IS NOT NULL THEN '중도탈락'
    END)AS completion_status
  , (CASE
    WHEN NOW() < end_opencourse AND stop_date IS NULL THEN null
    WHEN NOW() > end_opencourse AND stop_date IS NULL THEN end_opencourse
    WHEN NOW() IS NOT NULL THEN stop_date
    END)AS completeion_date, student_picture_name
```

```
FROM students s LEFT JOIN completion_course cc ON s.student_id = cc.student_id
        , open_course oc, course_history ch_, student_pictureList sp
        WHERE  ch_.student_id=s.student_id  AND oc.opencourse_id=ch_.opencourse_id
    AND s.student_id = sp.student_id
        AND ch_.opencourse_id = '개설과정아이디'
        AND ch_.opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = '강사아이디')
        LIMIT 0, 10;
```

Teacher_3_1



시험 및 배점 관리

과정에서 과목의 시험 문제를 추가할 수 있습니다.

과정 내역

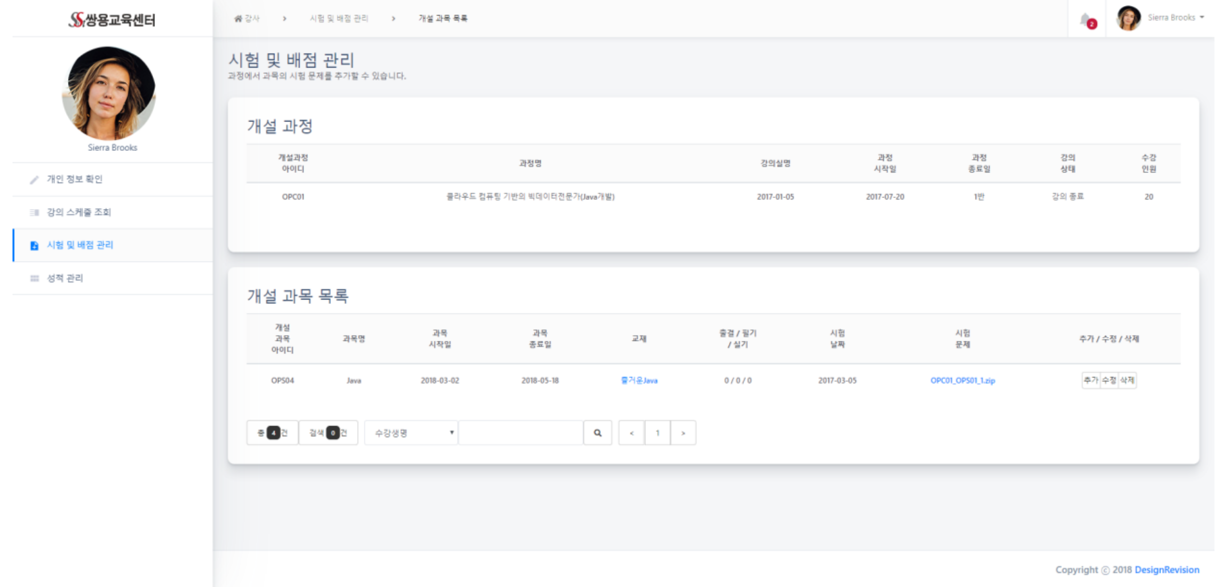
개설과정 아이디	과정명	강의실명	과정 시작일	과정 종료일	강의 상태	수강 인원	상세보기
OPC01	클라우드 컴퓨팅 기반의 빅데이터전문가(Java개발)	2017-01-05	2017-07-20	1반	강의 종료	20	🔍
OPC02	Java&python 응용 SW실무 개발자 양성 과정	2017-01-05	2017-07-20	3반	강의 중	23	🔍

```
# teacher_view1
CREATE OR REPLACE VIEW teacher_view1
AS
SELECT oc.opencourse_id, co.course_name
    , oc.start_opencourse, oc.end_opencourse
    , cr.classroom_name,(CASE WHEN NOW() < start_opencourse  THEN '강의예정'
                        WHEN NOW()> end_opencourse THEN '강의종료'
                        ELSE '강의중'END) AS opencourse
    ,(SELECT count(*)  FROM course_history ch WHERE oc.opencourse_id = opencourse_id ) st
udent
    FROM open_course oc, courses co, classrooms cr
    WHERE  oc.course_id = co.course_id
        AND oc.classroom_id = cr.classroom_id;
```

#과정내역 - 강의중, 강의예정, 강의종료

```
SELECT opencourse_id, course_name
    , start_opencourse, end_opencourse
    ,classroom_name, opencourse, student
        FROM teacher_view1
    WHERE opencourse = '강의종료'
    AND opencourse_id IN (SELECT os.opencourse_id
                            FROM teachers ins, open_subject os
                            WHERE ins.teacher_id = os.teacher_id
                            AND os.teacher_id = '강사아이디')
        LIMIT 0, 10;
```

Teacher_3_2



시험 및 배점 관리

과정에서 과목의 시험 문제를 추가할 수 있습니다.

개설 과정

개설과정 아이디	과정명	강의실명	과정 시작일	과정 종료일	강의 상태	수강 인원
OPC01	클라우드 컴퓨팅 기반의 빅데이터전문가(Java개발)	2017-01-05	2017-07-20	1반	강의 종료	20

개설 과목 목록

개설 과목 아이디	과목명	과목 시작일	과목 종료일	교재	출결 / 필기 / 실기	시험 날짜	시험 문제	추가 / 수정 / 삭제
OPS04	Java	2018-03-02	2018-05-18	즐거운Java	0 / 0 / 0	2017-03-05	OPC01_OPS01_1.zip	추가 수정 삭제

Copyright © 2018 DesignRevision

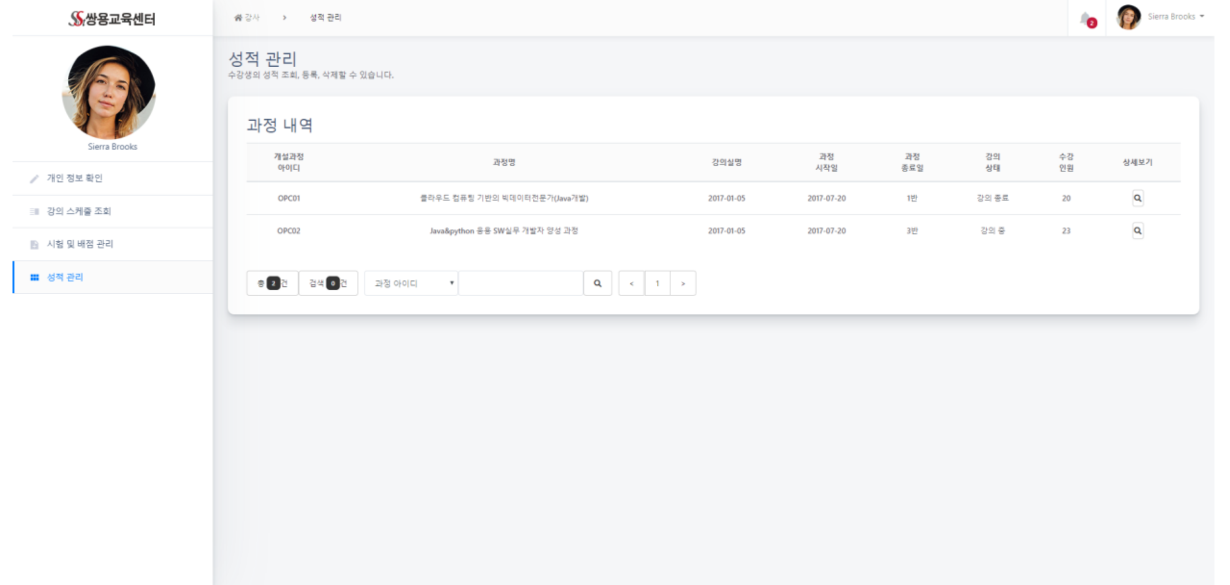
#개설과정

```
SELECT opencourse_id, course_name
    , start_opencourse, end_opencourse
    ,classroom_name, opencourse, student
        FROM teacher_view1
    WHERE opencourse_id = 'OPC01'
    AND opencourse_id IN (SELECT os.opencourse_id
                            FROM teachers ins, open_subject os
                            WHERE ins.teacher_id = os.teacher_id
                            AND os.teacher_id = '강사아이디');
```

```
#teacher_view3
CREATE OR REPLACE VIEW teacher_view3
AS
SELECT os.opensubject_id, s.subject_name, os.start_opensubject
          , os.end_opensubject, os.book_id, oc.opencourse_id
     , oc.start_openCourse, oc.end_openCourse
          FROM open_subject os, subjects s ,open_course oc
          WHERE os.subject_id = s.subject_id
          AND oc.opencourse_id = os.opencourse_id
          ORDER BY opensubject_id;

#teacher_view3_2
CREATE OR REPLACE VIEW teacher_view3_2
AS
SELECT v.opensubject_id, subject_name, start_opensubject
          , end_opensubject, book_id, opencourse_id, e.exam_id, e.exam_date, e.exam_file_name
     , start_openCourse, end_openCourse
     FROM teacher_view3 v LEFT JOIN exams e
     ON v.opensubject_id = e.opensubject_id
     ORDER BY opensubject_id;

#teacher_view3_3
CREATE OR REPLACE VIEW teacher_view3_3
AS
SELECT v.opensubject_id, subject_name, start_opensubject
          , end_opensubject, book_id, opencourse_id, v.exam_id, exam_date, exam_file_name
     , s.attendance_points, s.written_points, s.practice_points
     , start_openCourse, end_openCourse, s.subjectPoints_id
     FROM teacher_view3_2 v LEFT JOIN subject_score s
     ON v.exam_id = s.exam_id
     ORDER BY opensubject_id;

#개설 과목 목록
SELECT opensubject_id, subject_name, start_opensubject
          , end_opensubject, b.book_id, b.book_name, attendance_points
          , written_points, practice_points, exam_date, exam_file_name
          , (SELECT COUNT(*) FROM exams
           WHERE opensubject_id = v.opensubject_id) examCount
FROM teacher_view3_3 v LEFT JOIN books b
```

```
    ON v.book_id = b.book_id
    WHERE v.openCourse_id = 'OPC07'
    AND v.opencourse_id IN (SELECT os.opencourse_id
                       FROM teachers ins, open_subject os
                       WHERE ins.teacher_id = os.teacher_id
                       AND os.teacher_id = '강사아이디')
                       LIMIT 0, 10;

#추가
INSERT INTO  EXAMS (EXAM_ID, EXAM_DATE, EXAM_FILE, OPENSUBJECT_ID)
        VALUES ('EX01', '2017-03-05', 'OPC01_OPS01_1.zip', 'OPS01');
INSERT INTO  SUBJECT_SCORE (SUBJECTPOINTS_ID, ATTENDANCE_POINTS, WRITTEN_POINTS, P
RACTICE_POINTS, EXAM_ID)
        VALUES ('SP01', '20', '30', '50', 'EX01');

#수정
UPDATE SUBJECT_SCORE
        SET EXAM_DATE = ''
    , ATTENDANCE_POINTS =''
    , WRITTEN_POINTS =''
    , PRACTICE_POINTS =''
    WHERE EXAM_ID = 'EX01';

UPDATE EXAMS
        SET EXAM_DATE = ''
    , EXAM_FILE =''
    WHERE EXAM_ID = 'EX01';

#삭제
DELETE
    FROM EXAMS
    WHERE exam_id = 'EX01';
```

Teacher_4_1



성적 관리

수강생의 성적 조회, 등록, 삭제할 수 있습니다.

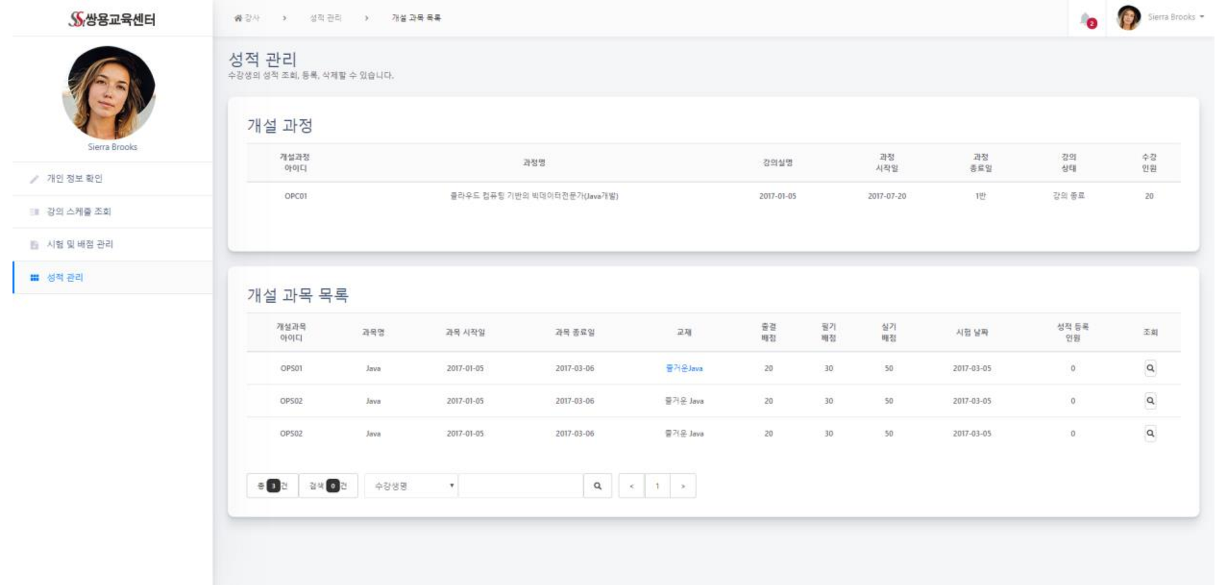
과정 내역

개설과정 아이디	과정명	강의실명	과정 시작일	과정 종료일	강의 상태	수강 인원	상세보기
OPC01	클라우드 컴퓨팅 기반의 빅데이터전문가(Java개발)	2017-01-05	2017-07-20	1반	강의 종료	20	🔍
OPC02	Java&python 응용 SW실무 개발자 양성 과정	2017-01-05	2017-07-20	3반	강의 중	23	🔍

#과정내역

```
SELECT opencourse_id, course_name
    , start_opencourse, end_opencourse
    ,classroom_name, opencourse, student
        FROM teacher_view1
    WHERE opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = '강사아이디')
        LIMIT 0, 10;
```

Teacher_4_2



성적 관리
수강생의 성적 조회, 등록, 삭제할 수 있습니다.

개설 과정

개설과정 아이디	과정명	강의실명	과정 시작일	과정 종료일	강의 상태	수강 인원
OPC01	클라우드 컴퓨팅 기반의 빅데이터전문가(Java개발)	2017-01-05	2017-07-20	1반	강의 종료	20

개설 과목 목록

개설과목 아이디	과목명	과목 시작일	과목 종료일	교재	출결 배점	필기 배점	실기 배점	시험 날짜	성적 등록 인원	조회
OPS01	Java	2017-01-05	2017-03-06	즐거운Java	20	30	50	2017-03-05	0	🔍
OPS02	Java	2017-01-05	2017-03-06	즐거운 Java	20	30	50	2017-03-05	0	🔍
OPS02	Java	2017-01-05	2017-03-06	즐거운 Java	20	30	50	2017-03-05	0	🔍

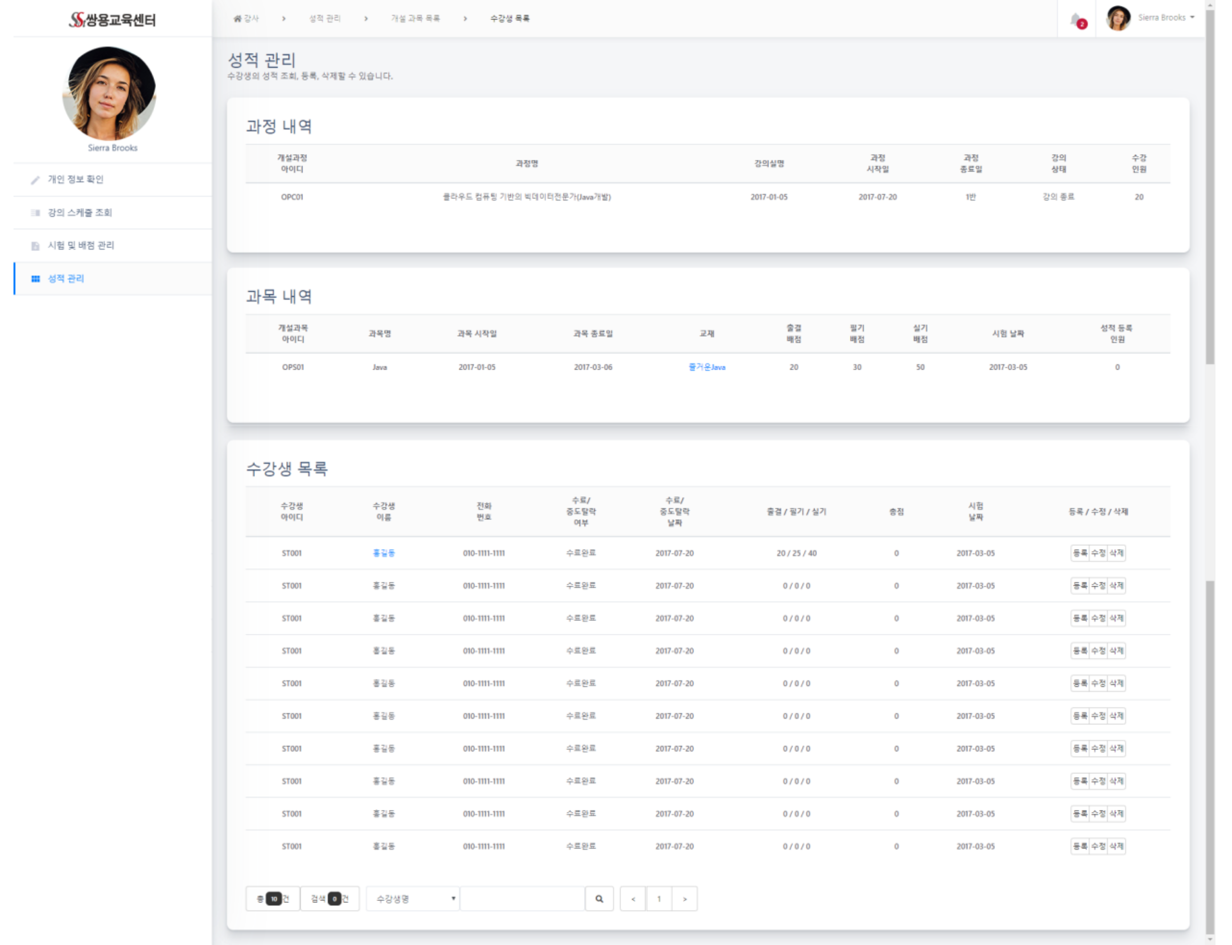
#개설과정

```
SELECT opencourse_id, course_name
    , start_opencourse, end_opencourse
    ,classroom_name, opencourse, student
        FROM teacher_view1
    WHERE opencourse_id = 'OPC02'
        AND opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = '강사아이디');
```

#개설 과목 목록

```
SELECT v.openSubject_id, subject_name, start_openSubject, end_openSubject
        , book_name, attendance_points, written_points, practice_points, exam_id
    , exam_date, (SELECT COUNT(*) FROM student_grade WHERE exam_id = v.exam_id) AS cou
nt_
    FROM teacher_view3_3 v LEFT JOIN books b
    ON v.book_id = b.book_id
    WHERE v.openCourse_id = 'OPC01'
    AND v.opencourse_id IN (SELECT os.opencourse_id
                                FROM teachers ins, open_subject os
                                WHERE ins.teacher_id = os.teacher_id
                                AND os.teacher_id = '강사아이디')
        LIMIT 0, 10;
```

Teacher_4_3



성적 관리

수강생의 성적 조회, 등록, 삭제할 수 있습니다.

과정 내역

개설과정 아이디	과정명	강의실명	과정 시작일	과정 종료일	강의 상태	수강 인원
OPC01	클라우드 컴퓨팅 기반의 빅데이터전문가(Java개발)	2017-01-05	2017-07-20	1반	강의 종료	20

과목 내역

개설과목 아이디	과목명	과목 시작일	과목 종료일	교재	출결 배점	필기 배점	실기 배점	시험 날짜	성적 등록 인원
OPS01	Java	2017-01-05	2017-03-06	즐거운Java	20	30	50	2017-03-05	0

수강생 목록

수강생 아이디	수강생 이름	전화 번호	수료/ 중도탈락 여부	수료/ 중도탈락 날짜	출결 / 필기 / 실기	총점	시험 날짜	등록 / 수정 / 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	20 / 25 / 40	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제
ST001	홍길동	010-1111-1111	수료완료	2017-07-20	0 / 0 / 0	0	2017-03-05	등록 수정 삭제

```
#개설과정
SELECT  opencourse_id,  course_name
       ,  start_opencourse,  end_opencourse
       ,classroom_name,  opencourse,  student
             FROM  teacher_view1
       WHERE  opencourse_id  =  'OPC02'
             AND  opencourse_id  IN  (SELECT  os.opencourse_id
                                          FROM  teachers  ins,  open_subject  os
                                          WHERE  ins.teacher_id  =  os.teacher_id
                                          AND  os.teacher_id  =  '강사아이디');


#개설  과목
SELECT  opensubject_id,  subject_name,  start_opensubject
            ,  end_opensubject,  b.book_name,  attendance_points
            ,  written_points,  practice_points,  exam_date,  exam_file_name,  exam_id
            ,  (SELECT  COUNT(*)  FROM  exams
              WHERE  opensubject_id  =  v.opensubject_id)  examCount
```

```
FROM teacher_view3_3 v LEFT JOIN books b
   ON v.book_id = b.book_id
   WHERE v.openCourse_id = 'OPC07'
   AND v.opencourse_id IN (SELECT os.opencourse_id
                   FROM teachers ins, open_subject os
                   WHERE ins.teacher_id = os.teacher_id
                   AND os.teacher_id = '강사아이디')
                   LIMIT 0, 10;

#teacher_view4_1
CREATE OR REPLACE VIEW teacher_view4_1
AS
 SELECT ch.student_id, ch.opencourse_id, stop_date
    FROM course_history ch left join completion_course cc
    on (ch.student_id=cc.student_id AND ch.opencourse_id=cc.opencourse_id)
    ORDER BY ch.student_id;

#teacher_view4_2
CREATE OR REPLACE VIEW teacher_view4_2
AS
SELECT t.student_id, opencourse_id, stop_date
        , student_name, student_phone, student_regDate
        FROM  teacher_view4_1 t LEFT JOIN students s ON t.student_id=s.student_id ORDER B
Y t.student_id;

#teacher_view4_3
CREATE OR REPLACE VIEW teacher_view4_3
AS
SELECT t.student_id, t.opencourse_id, stop_date
        , student_name, student_phone, student_regDate
    , (CASE
    WHEN NOW() < end_opencourse AND stop_date IS NULL THEN '진행중'
    WHEN NOW() > end_opencourse AND stop_date IS NULL THEN '수료완료'
    WHEN stop_date IS NOT NULL THEN '중도탈락'
    END)AS completion_status
        , (CASE
    WHEN NOW() < end_opencourse AND stop_date IS NULL THEN null
    WHEN NOW() > end_opencourse AND stop_date IS NULL THEN end_opencourse
    WHEN NOW() IS NOT NULL THEN stop_date
    END)AS completion_date
    FROM teacher_view4_2 t LEFT JOIN open_course oc ON t.opencourse_id=oc.opencourse_id
    ORDER BY t.student_id;

#teacher_view4_4
CREATE OR REPLACE VIEW teacher_view4_4
AS
SELECT t.student_id, t.opencourse_id, stop_date
        , student_name, student_phone, student_regDate
    , completion_status,completion_date, openSubject_id, teacher_id
    FROM teacher_view4_3 t LEFT JOIN open_subject os on t.opencourse_id=os.opencourse_id
    ORDER BY t.student_id;

#teacher_view4_5
CREATE OR REPLACE VIEW teacher_view4_5
AS
SELECT t.student_id, t.opencourse_id, stop_date
        , student_name, student_phone, student_regDate
    , completion_status, completion_date, exam_id, t.openSubject_id, exam_date, exam_file_name
    FROM teacher_view4_4 t LEFT JOIN exams ex ON t.opensubject_id=ex.opensubject_id
    ORDER BY t.student_id;

#teacher_view4_6
CREATE OR REPLACE VIEW teacher_view4_6
AS
SELECT t.student_id, opencourse_id, stop_date, student_name, student_phone, student_regDate
        ,completion_status, completion_date, t.exam_id, opensubject_id, exam_date, exam_file_name
    , attendance_score, written_score, practice_score
    , (attendance_score + written_score + practice_score)AS total_score
FROM teacher_view4_5 t LEFT JOIN student_grade sg ON (t.student_id=sg.student_id AND t.exam_id=sg.exam_id);

#수강생 목록
SELECT v.student_id, stop_date, student_name, student_phone
        , completion_status, completion_date
        , exam_date, attendance_score, written_score, practice_score, total_score, exam_id
    , student_picture_name
        FROM teacher_view4_6 v, student_pictureList sp
        WHERE v.student_id = sp.student_id
    AND opensubject_id='ops01'
        AND exam_id='ex01'
        LIMIT 0, 10;

#추가
INSERT INTO STUDENT_GRADE (STUDENTGRADE_ID, ATTENDANCE_SCORE, WRITTEN_SCORE, PRACTICE_SCORE, EXAM_ID, STUDENT_ID)
        VALUES ('SG001', '20', '18', '50', 'EX01', 'ST001');

#수정
UPDATE STUDENT_GRADE
        SET ATTENDANCE_SCORE = ''
    , WRITTEN_SCORE = ''
    , PRACTICE_SCORE = ''
    WHERE STUDENT_ID = 'ST001';

#삭제
DELETE FROM STUDENT_GRADE
        WHERE STUDENT_ID = 'ST001'
    AND EXAM_ID = '';
```