

학번 : \_\_\_\_\_ 학년 : \_\_\_\_\_ 이름 : \_\_\_\_\_

[공통사항]

- 기말고사 대체 과제 계정(FE학번)으로 오라클에 접속하여, 제공 파일(SQL, PL/SQL)을 실행할 것
- 제공된 SQL 파일은 수강신청 시스템과 스키마는 동일함(데이터는 일부 변경됨)
- 기본으로 제공한 파일명 및 <소스코드>는 수정 불가하고 번호가 표시된 빈칸을 작성하여 문제를 해결할 것  
(단, 오라클 접속 계정 입력 부분 제외)
- 문제의 모든 화면은 로그인 화면에서 교수 아이디와 패스워드로 로그인해야 확인이 가능함

[제공파일]

- 기본파일
  - SQL : FinalExam\_Schema.sql, Date2EnrollYear.sql(함수), Date2EnrollSemester.sql(함수)
  - JSP : main.jsp, top.jsp, login.jsp, login\_verify.jsp, logout.jsp
- [문제 1] 교수정보변경
  - SQL : BeforeUpdateProfessor.sql(트리거)
  - JSP : update\_professor.jsp, update\_professor\_verify.jsp
- [문제 2] 강좌개설신청
  - SQL : DeleteTeach.sql(프로시저)
  - JSP : insert\_teach.jsp, insert\_teach\_verify.jsp
  - JAVA : CourseMgr.java, Course.java
- [문제 3] 과목정보수정
  - SQL : UpdateClass.sql(프로시저)
  - JSP : update\_class.jsp, update\_class\_verify.jsp
- [문제 4] 수강정원변경
  - SQL : UpdateTeach.sql(프로시저)
  - JSP : update\_teach.jsp, update\_teach\_verify.jsp
- [문제 5] 성적등록
  - SQL : Date2GradeYear.sql(함수), Date2GradeSemester.sql(함수), ChangeGrade.sql(함수), InsertGrade.sql(프로시저)
  - JSP : grade.jsp, grade\_verify.jsp, update\_score\_verify.jsp
  - JAVA : GradeMgr.java, Grade.java

[문제 1] 아래의 <교수정보변경 화면>은 로그인 한 교수의 정보를 출력하고, 해당 교수의 연구분야와 패스워드를 수정하는 화면이다. <요구사항>을 참고하여, 교수의 연구분야 및 패스워드를 수정하기 위해 기존의 Professor 테이블의 스키마를 변경하고, JSP (update\_professor.jsp/update\_professor\_verify.jsp)와 트리거(BeforeUpdate Professor)를 완성하시오.

<교수정보변경 화면>

로그아웃	교수정보변경	강좌개설신청	수강정원변경	성적등록
이름	홍길동			
학부	IT학부			
전공	컴퓨터공학			
연구분야	데이터베이스			
패스워드	....			
수정				

<요구사항>

- 연구분야를 입력할 수 있도록 Professor 테이블에 p\_research 컬럼을 추가할 것(데이터 타입(크기) : VARCHAR2(100))
- ‘수정’ 버튼을 클릭할 경우, update\_professor\_verify.jsp로 수정한 정보를 전달하고 처리할 것
- <UPDATE 조건>을 참고하여 조건 불만족 시 예외를 발생시키는 BeforeUpdateProfessor 트리거를 생성할 것
- BeforeUpdateProfessor 트리거에서 예외 발생 시, <UPDATE 조건>과 같이 경고 창으로 메시지를 출력할 것

<UPDATE 조건>

- ① 패스워드는 문자와 숫자를 조합하여 입력해야 함 (조건 불만족 시 경고 창에 메시지 출력)

localhost:8080 내용:

패스워드는 문자와 숫자를 조합해야 합니다.

확인

- ② 연구분야는 ‘데이터베이스’, ‘네트워크’, ‘운영체제’만 입력해야 함 (조건 불만족 시 경고 창에 메시지 출력)

localhost:8080 내용:

연구분야를 다시 입력해주세요.

확인

- ③ 패스워드는 기존에 설정한 패스워드와 다르게 입력해야 함 (조건 불만족 시 경고 창에 메시지 출력)

- 단, 연구분야를 수정한 경우 기존에 설정한 패스워드를 변경하지 않아도 됨

localhost:8080 내용:

패스워드는 기존과 동일하게 변경할 수 없습니다.

확인

## <소스코드>

### [update\_professor.jsp]

```
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.sql.*" %>
<html>
  <head>
    <title>교수정보변경</title>
  </head>
  <body>
    <%@ include file="top.jsp" %>
    <%
      if (session_id==null) response.sendRedirect("login.jsp");

      Connection myConn = null;
      Statement stmt = null;
      ResultSet myResultSet = null;
      String mySQL = "";

      String dburl = "jdbc:oracle:thin:@210.94.199.20:1521:DBLAB";
      String user = "아이디"; // 본인 아이디(ex.FE0000000000)
      String passwd = "패스워드"; // 본인 패스워드(ex.FE0000000000)
      String dbdriver = "oracle.jdbc.driver.OracleDriver";

      try {
        【1-①】
      } catch(SQLException ex) {
        System.err.println("SQLException: " + ex.getMessage());
      }
      【1-②】
      if (myResultSet.next()) {
        String p_name = myResultSet.getString("p_name");
        String p_college = myResultSet.getString("p_college");
        String p_major = myResultSet.getString("p_major");
        String p_research = myResultSet.getString("p_research");
        String p_pwd = myResultSet.getString("p_pwd");
    %>
    <form name="update_prof" method="post" action="update_professor_verify.jsp">
      <input type="hidden" name="p_id" size="30" value="<%= session_id %>">
      <table width="75%" align="center" border>
        <tr>
          <th>이름</th>
          <td><input type="text" name="p_name" size="50" value="<%= p_name %>" disabled></td>
        </tr>
        <tr>
          <th>학부</th>
          <td><input type="text" name="p_college" size="50" value="<%= p_college %>" disabled> </td>
        </tr>
        <tr>
          <th>전공</th>
          <td><input type="text" name="p_major" size="50" value="<%= p_major %>" disabled></td>
        </tr>
        <tr>
          <th>연구분야</th>
          <td><input type="text" name="p_research" size="50" value="<%= p_research %>"></td>
        </tr>
        <tr>
          <th>패스워드</th>
          <td><input type="password" name="p_pwd" size="20" value="<%= p_pwd %>"></td>
        </tr>
        <tr>
          <td colspan="2" align="center"><input type="submit" value="수정"></td>
        </tr>
      </table>
    </form>
    <%
      }
      stmt.close();
      myConn.close();
    %>
  </body>
</html>
```

## [update\_professor\_verify.jsp]

```
<%@ page language="java" contentType="text/html; charset=utf-8" pageEncoding="utf-8"%>
<%@ page import="java.sql.*" %>
<!DOCTYPE html>
<html>
  <head>
    <title>교수정보변경</title>
  </head>
  <body>
    <%
      request.setCharacterEncoding("utf-8");
      String p_id = request.getParameter("p_id");
      String p_pwd = request.getParameter("p_pwd");
      String p_research = request.getParameter("p_research");

      Connection myConn = null;
      Statement stmt = null;
      String mySQL = "";

      String dburl = "jdbc:oracle:thin:@210.94.199.20:1521:DBLAB";
      String user = "아이디"; // 본인 아이디(ex.FE0000000000)
      String passwd = "패스워드"; // 본인 패스워드(ex.FE0000000000)
      String dbdriver = "oracle.jdbc.driver.OracleDriver";

      try {
        【1-㉓】
      } catch(SQLException ex) {
        System.err.println("SQLException: " + ex.getMessage());
      }
      【1-㉔】
      try {
        stmt.executeUpdate(mySQL);
    %>
    <script>
      alert("정보가 수정되었습니다.");
      location.href="update_professor.jsp";
    </script>
    <%
      } catch(SQLException ex) {
        String sMessage;
        【1-㉕】
    %>
    <script>
      alert("<%= sMessage %>");
      location.href = "update_professor.jsp";
    </script>
    <%
      } finally {
        if (stmt != null) {
          try {
            stmt.close();
            myConn.close();
          } catch(SQLException ex) { }
        }
      }
    %>
  </body>
</html>
```

## [BeforeUpdateProfessor.sql]

```
CREATE OR REPLACE TRIGGER BEFOREUPDATEPROFESSOR
```

【1-㉖】

```
END;
```

[문제 2] 아래의 <강좌개설신청 화면>은 현재 날짜를 기준으로 개설신청 현황을 보여주고, 교수가 과목의 개설신청/취소를 하는 화면이다. <요구사항>을 참고하여, 과목의 개설신청/취소를 위한 JSP(insert\_teach.jsp/insert\_teach\_verify.jsp), JAVA(CourseMgr.java), 프로시저(DeleteTeach)를 완성하시오.

강좌개설신청 화면					
로그아웃		교수정보변경		강좌개설신청	
로그아웃		교수정보변경		수강정원변경	
로그아웃		교수정보변경		성적등록	
과목번호	분반	과목명	학점	교수	개설신청
C100	1	컴퓨터 프로그래밍	3	박구곤	
C100	2	컴퓨터 프로그래밍	3		신청
C200	1	자료구조	3	홍길동	취소
C300	1	알고리즘	3	홍길동	취소
C400	1	데이터베이스 시스템	3	이은선	
C400	2	데이터베이스 시스템	3	홍길동	취소
C500	1	운영체제	3	서의홍	
C600	1	소프트웨어 공학	3	홍길동	취소
C700	1	네트워크	3		신청
C700	2	네트워크	3	홍길동	취소
C800	1	데이터베이스 프로그래밍	3	김심선	
C900	1	객체지향 윈도우즈 프로그래밍	3	이상헌	
C900	2	객체지향 윈도우즈 프로그래밍	3		신청
M100	1	멀티미디어 개론	3	김정미	
M100	2	멀티미디어 개론	3	홍길동	취소
M200	1	선형대수	3	신기웅	
M300	1	그래픽 활용	3	이상헌	
M400	1	윈도우즈 프로그래밍	3	김정미	
M400	2	윈도우즈 프로그래밍	3		신청
M500	1	컴퓨터 그래픽스	3		신청
M600	1	멀티미디어 처리	3		신청
M700	1	게임 프로그래밍	3	이상헌	
M700	2	게임 프로그래밍	3		신청

<요구사항>

- 미리 정의된 Date2EnrollYear 함수와 Date2EnrollSemester 함수를 호출하여 리턴받은 해당 년도/학기를 매개변수로 전달하고, 과목번호, 분반을 오름차순으로 정렬하여 개설신청 현황 목록을 출력할 것
  - 개설신청 현황 목록은 CourseMgr.getCourseList()를 사용하여 처리할 것
  - 담당교수가 없는 과목은 개설신청에 “신청” 버튼을 표시할 것
  - 로그인 한 교수가 이미 개설 신청한 과목은 개설신청에 “취소” 버튼을 표시할 것
  - ‘신청’, ‘취소’ 버튼을 클릭할 경우, insert\_teach\_verify.jsp로 정보를 전달하고 처리할 것
  - ‘신청’은 CourseMgr.insertTeach()를 사용하여 처리하는데, 해당 교과목 정보를 입력한 후 <신청 완료>와 같이 메시지를 경고 창으로 출력할 것(단, 시간(t\_time), 강의실(t\_where)은 NULL 값, 수강정원(t\_max)는 5로 입력할 것)
  - <취소 조건>과 같이 처리 결과를 반환하는 DeleteTeach 프로시저를 생성할 것(단, <취소 조건> ①은 예외 처리)
  - ‘취소’는 CourseMgr.deleteTeach() 사용하여 처리하는데, DeleteTeach 프로시저의 결과 메시지를 경고 창으로 출력할 것
  - 로그인 한 교수가 이미 개설 신청한 과목에 한해서 과목명을 클릭하면 해당 과목의 강의실과 시간을 입력하는 update\_class.jsp로 정보를 전달하고 처리할 것
- [문제 3] 참고

<신청 완료>

localhost:8080 내용:  
해당 과목의 개설 신청이 완료되었습니다.

확인

<취소 조건>

- ① 해당 과목의 수강 신청 인원이 수강 정원의 50%이상이면 취소할 수 없음 (조건 만족 시 경고 창에 메시지 출력)

localhost:8080 내용:  
해당 과목은 수강인원이 정원의 50%보다 많아 개설 신청을 취소할 수 없습니다.

확인

- ② 해당 과목의 수강 신청 인원이 수강 정원의 50%미만이면 취소할 수 있음 (조건 만족 시 경고 창에 메시지 출력)

localhost:8080 내용:  
해당 과목의 개설 신청이 취소되었습니다.

확인

<소스코드>

[insert\_teach.jsp]

```
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.util.*, eduBean.*" %>
<html>
  <head>
    <title>강좌개설신청</title>
  </head>
  <body>
    <%@ include file="top.jsp" %>
    <%
      if (session_id==null) response.sendRedirect("login.jsp");
    %>
    <table width="75%" align="center" border>
      <tr>
        <th>과목번호</th>
        <th>분반</th>
        <th>과목명</th>
        <th>학점</th>
        <th>교수</th>
        <th>개설신청</th>
      </tr><br>
      <tr>
        <td align="center"><%=en.getCId()%></td>
        <td align="center"><%=en.getCIdNo()%></td>
        <td align="center">
          <% if(session_id.equals(en.getPId())) { %>
            <a href="update_class.jsp?p_id=<%=en.getPId()%>&c_id=<%=
en.getCIdNo() %>"><%=en.getCName()%>
          </td>
        <td align="center"><%=en.getUnit()%></td>
        <td align="center">
          <% if (en.getPName() == null) { %>
            &nbsp;
          <% } else { %>

```

```

        <%=en.getPName()%>
    <% } %>
    </td>
    <td align="center">
    <% if(en.getPId() == null) { %>
        <a href="insert_teach_verify.jsp?c_id=<%= en.getCId() %>&c_id_no=<%= en.getCIdNo() %>&type=1">
신청</a>
    <% } else if(session_id.equals(en.getPId())) { %>
        <a href="insert_teach_verify.jsp?c_id=<%= en.getCId() %>&c_id_no=<%= en.getCIdNo() %>&type=2">
취소</a>
    <% } else { %>
        &nbsp;
    <% } %>
    </td>
    <% } %>
</tr>
</table>
</body>
</html>

```

### [insert\_teach\_verify.jsp]

```

<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="eduBean.*" %>
<html>
<head>
    <title> 강좌개설신청 </title>
</head>
<body>
<%
    String p_id = (String)session.getAttribute("user");
    String c_id = request.getParameter("c_id");
    int c_id_no = Integer.parseInt(request.getParameter("c_id_no"));
    int type = Integer.parseInt(request.getParameter("type"));
    String result = null;
%>
    [2-②]

<%
    if (type == 1) {
        [2-③]

%>
        <script>
            alert("해당 과목의 개설 신청이 완료되었습니다.");
        </script>
<%
    } else {
        [2-④]

%>
        <script>
            alert("<%=result%>");
        </script>
<%
    } %>
    <script>
        location.href="insert_teach.jsp";
    </script>
</body>
</html>

```

### [CourseMgr.java]

```

package eduBean;
import java.sql.*;
import java.util.*;
import javax.sql.*;
import oracle.jdbc.driver.*;
import oracle.jdbc.pool.*;
import eduBean.*;

public class CourseMgr {
    private OracleConnectionPoolDataSource ocpsds = null;
    private PooledConnection pool = null;

    public CourseMgr() {

```

```

try{
    ocpds = new OracleConnectionPoolDataSource();

    ocpds.setURL("jdbc:oracle:thin:@210.94.199.20:1521:DBLAB");
    ocpds.setUser("아이디"); // 본인 아이디(ex.FE0000000000)
    ocpds.setPassword("비밀번호"); // 본인 패스워드(ex.FE0000000000)

    pool = ocpds.getPooledConnection();
} catch(Exception e) {
    System.out.println("Error : Connection Failed");
}
}
// 해당 년도 조회
public int getFullYear() {
    int nYear=0;
    Connection conn = null;
    CallableStatement cstmt = null;
    try {
        [2-⑤]

    } catch (Exception ex) {
        System.out.println("Exception" + ex);
    }
    return nYear;
}

// 해당 학기 조회
public int getFullYearSemester() {
    int nSemester=0;
    Connection conn = null;
    CallableStatement cstmt = null;

    try {
        [2-⑥]

    } catch (Exception ex) {
        System.out.println("Exception" + ex);
    }
    return nSemester;
}

// 개설신청현황 조회
public Vector getCourseList() {
    Connection conn = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    Vector vecList = new Vector();

    try {
        [2-⑦]

        while (rs.next()) {
            Course en = new Course();
            en.setCid(rs.getString("cid"));
            en.setCidNo(rs.getInt("cid_no"));
            en.setCName(rs.getString("cname"));
            en.setCUnit(rs.getInt("cunit"));
            en.setPid(rs.getString("pid"));
            en.setPName(rs.getString("pname"));
            vecList.add(en);
        }
        pstmt.close();
        conn.close();

    } catch (Exception ex) {
        System.out.println("Exception" + ex);
    }
    return vecList;
}

// 강좌개설신청
public void insertTeach(String p_id, String c_id, int c_id_no) {
    Connection conn = null;
    PreparedStatement pstmt = null;

    try {

```



**[2- ⑧]**

```

    } catch (Exception ex) {
        System.out.println("Exception" + ex);
    }
}

```

// 강좌개설신청 취소

```

public String deleteTeach(String p_id, String c_id, int c_id_no) {
    Connection conn = null;
    CallableStatement cstmt = null;
    String Result = null;

```

```

    try {

```

**[2- ⑨]**

```

    } catch (Exception ex) {
        System.out.println("Exception" + ex);
    }

```

```

    return Result;

```

```

}

```

```

}

```

**[Course.java]**

```

package eduBean;

```

```

public class Course {
    private String c_id;
    private int c_id_no;
    private String c_name;
    private int c_unit;
    private int t_max;
    private String p_id;
    private String p_name;

```

```

    public Course() {
        c_id = null;
        c_id_no = 0;
        c_name = null;
        c_unit = 0;
        t_max = 0;
        p_id = null;
        p_name = null;
    }

```

```

    public void setCId(String c_id) {
        this.c_id = c_id;
    }

```

```

    public void setCIdNo(int c_id_no) {
        this.c_id_no = c_id_no;
    }

```

```

    public void setCName(String c_name) {
        this.c_name = c_name;
    }

```

```

    public void setCUnit(int c_unit) {
        this.c_unit = c_unit;
    }

```

```

    public void setTMax(int t_max) {
        this.t_max = t_max;
    }

```

```

    public void setPId(String p_id) {
        this.p_id = p_id;
    }

```

```

    public void setPName(String p_name) {
        this.p_name = p_name;
    }

```

```

    public String getCId() {
        return c_id;
    }

```

```
}

public int getCIdNo() {
    return c_id_no;
}

public String getCName() {
    return c_name;
}

public int getCUnit() {
    return c_unit;
}

public int getTMax() {
    return t_max;
}

public String getPId() {
    return p_id;
}

public String getPName() {
    return p_name;
}

}
```

#### [DeleteTeach.sql]

```
CREATE OR REPLACE PROCEDURE DELETETEACH
```

**【2- ⑩】**

```
END;
```

[문제 3] 아래의 <과목정보수정 화면>은 [문제 2]에서 구현한 개설신청 등록 화면에서 과목명을 클릭할 경우 과목 정보가 출력되며, 해당 과목의 강의실과 시간을 수정하는 화면이다. <요구사항>을 참고하여, 해당 과목의 강의실과 시간을 수정하기 위한 JSP(update\_class.jsp/update\_class\_verify.jsp), 프로시저(UpdateClass)를 완성하시오.

<과목정보수정 화면>

로그아웃	교수정보변경	강좌개설신청	수강정원변경	성적등록
과목명	데이터베이스 시스템			
분반	2			
교수명	홍길동			
강의실	<div>▼</div>			
시간	<div></div>			
<div>수정</div>				

<요구사항>

- ‘수정’ 버튼을 클릭할 경우, update\_class\_verify.jsp로 수정한 정보를 전달하고 처리할 것
- <UPDATE 조건>을 참고하여 조건 불만족 시 예외를 발생시키는 UpdateClass 프로시저를 생성할 것
- UpdateClass 프로시저에서 예외 발생 시, <UPDATE 조건>과 같이 경고 창으로 메시지를 출력할 것

<UPDATE 조건>

- ① 강의실과 시간은 기존에 저장된 데이터와 다르게 입력해야 함 (조건 불만족 시 경고 창에 메시지 출력)

localhost:8080 내용:
변경된 내용이 없습니다.

확인

- ② 강의실과 시간은 이미 등록된 강의실 및 시간과 중복되게 입력할 수 없음 (조건 불만족 시 경고 창에 메시지 출력, 단, 중복되는 과목명을 함께 출력함)

localhost:8080 내용:
자료구조 과목과 중복되어 처리할 수 없습니다.

확인

- ①, ②의 조건에 모두 만족한다면 변경 처리함 (조건 만족 시 경고 창에 메시지 출력)

localhost:8080 내용:
과목정보를 변경하였습니다.

확인

<소스코드>

[update\_class.jsp]

```

<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.sql.*" %>
<html>
  <head>
    <title>과목정보수정</title>
  </head>
  <body>
    <%@ include file="top.jsp" %>
    <%

```

```

if (session_id==null) response.sendRedirect("login.jsp");

Connection conn = null;
Statement stmt = null;
ResultSet myResultSet = null;
CallableStatement cstmt1 = null;
CallableStatement cstmt2 = null;
int nYear = 0;
int nSemester = 0;
String mySQL = "";

String dburl = "jdbc:oracle:thin:@210.94.199.20:1521:DBLAB";
String user = "아이디"; // 본인 아이디(ex,FE0000000000)
String passwd = "패스워드"; // 본인 패스워드(ex,FE0000000000)
String dbdriver = "oracle.jdbc.driver.OracleDriver";

String cid = request.getParameter("c_id");
String cidno = request.getParameter("c_id_no");

```

```
try {
```

**[3- ①]**

```

} catch(SQLException ex) {
    System.err.println("SQLException: " + ex.getMessage());
}

```

```

if (myResultSet.next()) {
    String c_name = myResultSet.getString("cname");
    String c_twhere = myResultSet.getString("twhere");
    String c_time = myResultSet.getString("ctime");
    String p_name = myResultSet.getString("pname");
}

```

```
%>
```

```

<form name="update_class" method="post" action="update_class_verify.jsp">
  <input type="hidden" name="p_id" size="30" value="<%= session_id %>">
  <table width="75%" align="center" border>
    <tr>
      <th>과목명</th>
      <td><%= c_name %></td>
    </tr>
    <tr>
      <th>분반</th>
      <td><%= cidno %><input type="hidden" name="c_id_no" value="<%=cidno%>"></td>
    </tr>
    <tr>
      <th>교수명</th>
      <td><%= p_name %></td>
    </tr>
    <tr>
      <th>강의실</th>
      <td>
        <select name="t_where">
          <option value=""></option>
          <option value="인-201">인-201</option>
          <option value="인-202">인-202</option>
          <option value="인-309">인-309</option>
          <option value="인-310">인-310</option>
          <option value="중-201">중-201</option>
          <option value="중-202">중-202</option>
          <option value="중-301">중-301</option>
          <option value="중-417">중-417</option>
        </select>
      </td>
    </tr>
    <tr>
      <th>시간</th>
      <td>
        <% if (c_time == null) {%>
          <td><input type="text" name="t_time" size="5" value=""></td>
        <% } else { %>
          <td><input type="text" name="t_time" size="5" value="<%= c_time %>"></td>
        <% } %>
      </tr>
    <tr>
      <td colspan="2" align="center">
        <input type="hidden" name="c_id" value="<%=cid%>">
        <input type="submit" value="수정">
      </td>
    </tr>
  </table>
</form>
<script>
  document.update_class.t_where.value = "<%=c_twhere%>";
</script>
<%
}
cstmt1.close();
cstmt2.close();
stmt.close();
conn.close();

```

```
%>
</body>
</html>
```

### [update\_class\_verify.jsp]

```
<%@page import="com.sun.xml.internal.bind.v2.runtime.Location"%>
<%@ page language="java" contentType="text/html; charset=utf-8" pageEncoding="utf-8"%>
<%@ page import="java.sql.*"%>
<html>
<head>
<title>과목정보수정</title>
</head>
<body>
<%
    request.setCharacterEncoding("utf-8");
    String session_id = request.getParameter("p_id");
    String c_id = request.getParameter("c_id");
    int c_id_no = Integer.parseInt(request.getParameter("c_id_no"));
    String t_where = request.getParameter("t_where");

    Connection myConn = null;
    CallableStatement cstmt = null;
    String sMessage = null;

    String dburl = "jdbc:oracle:thin:@210.94.199.20:1521:DBLAB";
    String user = "아이디"; // 본인 아이디(ex.FE0000000000)
    String passwd = "패스워드"; // 본인 패스워드(ex.FE0000000000)
    String dbdriver = "oracle.jdbc.driver.OracleDriver";

    if (t_where.equals("")){
        %>
        <script>
            alert("입력란을 입력해주세요.");
            location.href = "update_class.jsp?p_id="+<%=session_id%>+"&c_id="+<%=c_id%>+"&c_id_no="+
                "<%=c_id_no%>";
        </script>
    }else{
        try{
            int t_time = Integer.parseInt(request.getParameter("t_time"));

            try {
                【3- ②】

                %>
                <script>
                    alert("<%=sMessage%>");
                    location.href="insert_teach.jsp";
                </script>
                %>
                cstmt.close();
                myConn.close();
            } catch(SQLException ex) {
                System.err.println("SQLException: " + ex.getMessage());
            }
        } catch (Exception ex) {
            %>
            <script>
                alert("입력란을 입력해주세요.");
                location.href = "update_class.jsp?p_id="+<%=session_id%>+"&c_id="+<%=c_id%>+"&c_id_no="+
                    "<%=c_id_no%>";
            </script>
            %>
        }
    }
}
%>
</body>
</html>
```

### [UpdateClass.sql]

```
CREATE OR REPLACE PROCEDURE UPDATECLASS
```

**【3- ③】**

```
END;
```

[문제 4] 아래의 <수강정원변경 화면>은 교수가 개설 신청한 과목의 수강정원을 변경하는 화면이다. <요구사항>을 참고하여, 수강정원 변경을 위한 JSP(update\_teach.jsp/update\_teach\_verify.jsp), 프로시저(UpdateTeach)를 완성하시오.

로그아웃

교수정보변경

강좌개설신청

수강정원변경

성적등록

과목번호	분반	과목명	학점	수강정원
C200	1	자료구조	3	6
C300	1	알고리즘	3	7
C400	2	데이터베이스 시스템	3	5
C600	1	소프트웨어 공학	3	5
C700	2	네트워크	3	5
M100	2	멀티미디어 개론	3	5

수강정원변경

<요구사항>

- ‘수강정원변경’ 버튼을 클릭할 경우, update\_teach\_verify.jsp로 수정한 정보를 전달하고 처리할 것
- <수강정원변경 조건>을 참고하여 조건 불만족 시 예외를 발생시키는 UpdateTeach 프로시저를 생성할 것
- UpdateTeach 프로시저에서 예외 발생 시, <수강정원변경 조건>과 같이 경고 창으로 메시지를 출력할 것
- UpdateTeach 프로시저에서 Date2EnrollYear 함수, Date2EnrollSemester 함수를 사용하여 처리할 것

<수강정원변경 조건>

- \* ①-⑤의 조건 불만족 시 경고 창에 한 번에 메시지를 출력
- ① 수강정원에 공백(스페이스) 값을 입력할 수 없음
  - ② 수강정원에 숫자 외에 다른 문자를 입력할 수 없음
  - ③ 수강정원에 0~100 사이의 값만 입력할 수 있음
  - ④ 수강정원에 NULL 값을 입력할 수 없음
  - ⑤ 수강정원은 현재 수강 신청 인원 미만으로 입력할 수 없음

localhost:8080 내용:

자료구조 : 변경 불가(사유 : 수강정원에는 공백을 입력할 수 없습니다.)

알고리즘 : 변경 불가(사유 : 숫자 외에 다른 문자를 입력할 수 없습니다.)

데이터베이스 시스템 : 변경 불가(사유 : 수강정원은 0~100사이의 값만 입력 가능합니다.)

소프트웨어공학 : 변경 불가(사유 : 수강정원을 입력하지 않았습니다.)

네트워크 : 변경 실패(사유 : 강의를 신청한 학생 수 5명 이상의 수를 입력해야 합니다.)

멀티미디어 개론 : 변경완료

확인

## <소스코드>

### [update\_teach.jsp]

```
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.util.*"%>
<%@ page import="java.sql.*"%>
<html>
  <head>
    <title>수강정원변경</title>
  </head>
  <body>
    <%@ include file="top.jsp" %>
    <% if (session_id==null) response.sendRedirect("login.jsp"); %>
    <form method="post" action="update_teach_verify.jsp">
      <table width="75%" align="center" border>
        <tr>
          <th>과목번호</th>
          <th>분반</th>
          <th>과목명</th>
          <th>학점</th>
          <th>수강정원</th>
        </tr>
        <tr>
          <td align="center"><%=cid%><input type="hidden" name="cid" value=<%=cid%>></td>
          <td align="center"><%=cid_no%><input type="hidden" name="cidno" value=<%=cid_no%>></td>
          <td align="center"><%=cname%></td>
          <td align="center"><%=cunit%><input type="hidden" name="cunit" value=<%=cunit%>></td>
          <td align="center"><input type="text" name="tmax" size=5 value=<%=tmax%>></td>
        </tr>
      </table>
      <br>
      <table align="center">
        <tr>
          <td><input type="submit" name="submit" value="수강정원변경"></td>
        </tr>
      </table>
    </form>
  </body>
</html>

<%
  Connection conn = null;
  PreparedStatement pstmt = null;
  CallableStatement cstmt1 = null;
  CallableStatement cstmt2 = null;
  String mySQL = "";
  String p_id = (String)session.getAttribute("user");
  int nYear = 0;
  int nSemester = 0;
  ResultSet myResultSet = null;

  String dburl = "jdbc:oracle:thin:@210.94.199.20:1521:DBLAB";
  String user = "아이디"; // 본인 아이디(ex.FE0000000000)
  String passwd = "패스워드"; // 본인 패스워드(ex.FE0000000000)
  String dbdriver = "oracle.jdbc.driver.OracleDriver";

  try {
    [4-①]
  } catch(SQLException ex) {
    System.err.println("SQLException: " + ex.getMessage());
  }

  [4-②]

  while (myResultSet.next()) {
    String cid = myResultSet.getString("cid");
    String cid_no = myResultSet.getString("cid_no");
    String cname = myResultSet.getString("cname");
    String cunit = myResultSet.getString("cunit");
    String tmax = myResultSet.getString("tmax");

    <tr>
      <td align="center"><%=cid%><input type="hidden" name="cid" value=<%=cid%>></td>
      <td align="center"><%=cid_no%><input type="hidden" name="cidno" value=<%=cid_no%>></td>
      <td align="center"><%=cname%></td>
      <td align="center"><%=cunit%><input type="hidden" name="cunit" value=<%=cunit%>></td>
      <td align="center"><input type="text" name="tmax" size=5 value=<%=tmax%>></td>
    </tr>
  }
  pstmt.close();
  conn.close();
%>
```

## [update\_teach\_verify.jsp]

```
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.util.*"%>
<%@ page import="java.sql.*" %>
<html>
  <head>
    <title> 수강정원변경 </title>
  </head>
  <body>
    <script>
      var message = "";
    <%
      String p_id = (String)session.getAttribute("user");
      String result = null;
      String cid[] = request.getParameterValues("cid");
      String cidno[] = request.getParameterValues("cidno");
      String cunit[] = request.getParameterValues("cunit");
      String tmax[] = request.getParameterValues("tmax");

      Connection myConn = null;
      CallableStatement cstmt = null;
      String mySQL = "";

      String dburl = "jdbc:oracle:thin:@210.94.199.20:1521:DBLAB";
      String user = "아이디"; // 본인 아이디(ex.FE0000000000)
      String passwd = "패스워드"; // 본인 패스워드(ex.FE0000000000)
      String dbdriver = "oracle.jdbc.driver.OracleDriver";

      try {

        [4- ③]

      }
    <%>

    [4- ④]

    <%
      }
    } catch(SQLException ex) {
      System.err.println("SQLException: " + ex.getMessage());
    }
    <%>
    alert(message);
    location.href = "update_teach.jsp";
  </script>
</body>
</html>
```

## [UpdateTeach.sql]

```
CREATE OR REPLACE PROCEDURE UPDATETEACH
```

```
[4- ⑤]
```

```
END;
```



[문제 5] 아래의 <성적등록 화면>은 현재 날짜를 기준으로 성적 처리가 가능한 학생들의 수강 정보를 보여주고, 성적을 등록하는 화면이다. 아래의 <요구사항>을 참고하여, 성적 등록을 위해 기존의 Enroll 테이블의 스키마를 변경하고, JSP(grade.jsp/grade\_verify.jsp/update\_score\_verify.jsp), Java(GradeMgr.java), 함수(Date2Grade Year/Date2GradeSemester/ChangeGrade), 프로시저(InsertGrade)를 완성하시오.

**<성적등록 화면>**

로그아웃
교수정보변경
강좌개설신청
수강정원변경
성적등록

과목번호	분반	과목명	학번	학생	점수	성적
C200	1	자료구조	20011234	신경화	<input type="text" value="65"/>	D
C200	1	자료구조	20011235	서용만	<input type="text" value="78"/>	C
C200	1	자료구조	20021245	전용호	<input type="text" value="0"/>	F
C200	1	자료구조	20021246	한신동	<input type="text" value="88"/>	B
C200	1	자료구조	20021247	노호현	<input type="text" value="90"/>	A

성적일괄등록

2020
년도
2
학기
조회

**<요구사항>**

- 성적을 입력할 수 있도록 Enroll 테이블에 e\_grade 컬럼을 추가할 것(데이터 타입(크기) : VARCHAR2(5))
- 기본 조회 조건의 년도는 현재년도를 반환하는 Date2GradeYear 함수, 학기는 3월~8월인 경우 1학기, 9월~2월인 경우 2학기를 반환하는 Date2GradeSemester 함수를 생성하여 처리할 것
- ‘조회’ 버튼을 클릭할 경우, grade\_verify.jsp로 정보를 전달하고 처리할 것  
(조회 조건 값으로 년도/학기를 입력하여 성적을 조회할 수 있음)
- ‘조회’는 GradeMgr.getScoreList()를 사용하여 처리할 것
- ‘성적일괄등록’ 버튼을 클릭할 경우, score\_verify.jsp로 정보를 전달하고 처리할 것
- 점수를 입력받아 성적을 반환하는 ChangeGrade 함수를 생성하여 처리할 것  
(점수에 따른 성적 조건 - 90점 이상 : A / 80~89 : B / 70~79 : C / 60~69 : D / 60점 미만 : F)
- <성적일괄등록 조건>을 참고하여 InsertGrade 프로시저를 생성할 것
- InsertGrade 프로시저에서 Date2GradeYear 함수, Date2GradeSemester 함수, ChangeGrade 함수를 사용하여 성적을 처리할 것
- ‘성적일괄등록’은 GradeMgr.setScore()를 사용하여 처리하는데, InsertGrade 프로시저에서 작성한 예외처리 메시지를 경고 창으로 출력할 것

**<성적일괄등록 조건>**

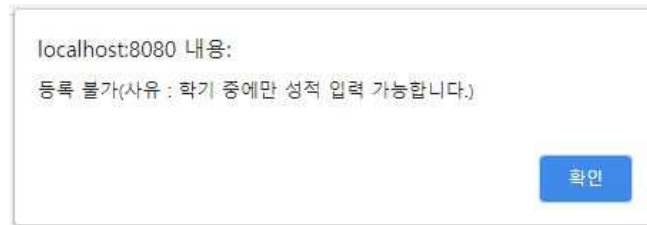
- ① 현재 년도/학기의 성적만 처리할 수 있음 (조건 불만족 시 경고 창에 메시지 출력)

localhost:8080 내용:

등록 불가(사유 : 현재 학기 성적만 입력 가능합니다.)

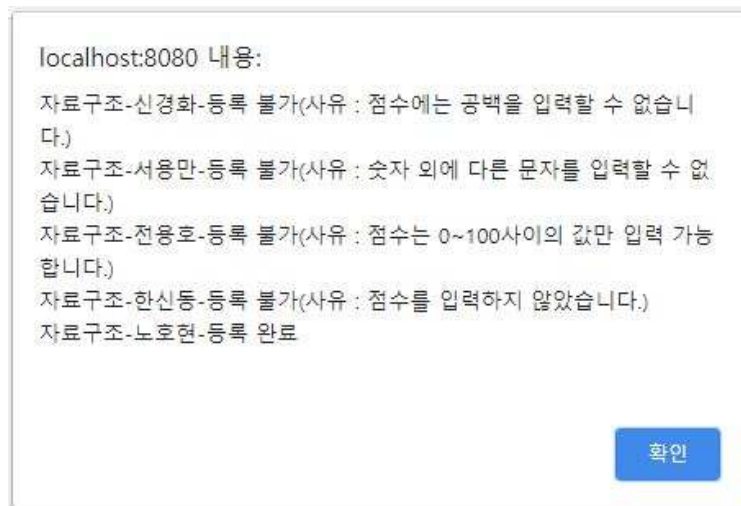
확인

② 현재 년도/학기 중에서 방학 기간(7-8월, 1-2월)에는 성적을 처리할 수 없음 (조건 불만족 시 경고 창에 메시지 출력)



\* ③-⑥의 조건 불만족 시 경고 창에 한 번에 메시지를 출력

- ③ 점수에 공백(스페이스) 값을 입력할 수 없음
- ④ 점수에 숫자 외에 다른 문자를 입력할 수 없음
- ⑤ 점수에 0~100 사이의 값만 입력할 수 있음
- ⑥ 점수에 NULL 값을 입력할 수 없음



### <소스코드>

#### [grade.jsp]

```
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.util.*, eduBean.*"%>
<html>
  <head>
    <title>성적등록</title>
  </head>
  <body>
    <%@ include file="top.jsp" %>
    <% if (session_id==null) response.sendRedirect("login.jsp"); %>
    <form method="post" action="update_score_verify.jsp">
      <table width="75%" align="center" border>
        <tr>
          <th>과목번호</th>
          <th>분반</th>
          <th>과목명</th>
          <th>학번</th>
          <th>학생</th>
          <th>점수</th>
          <th>성적</th>
        </tr><br>
        <tr>
          <td colspan="7">[5-①]</td>
        </tr>
      </table>
    </form>
    <%
      Vector vlist = null;
      int year, semester;

      if (request.getParameter("year") == null && request.getParameter("semester") == null) {
        year = gradeMgr.getCurrentYear();
      }
    %>
  </body>
</html>
```

```

        semester = gradeMgr.getCurrentSemester();
    } else {
        year = Integer.parseInt(request.getParameter("year"));
        semester = Integer.parseInt(request.getParameter("semester"));
    }

    vlist = gradeMgr.getScoreList(session_id, year, semester);
    int counter = vlist.size();

    for(int i=0; i<vlist.size(); i++) {
        Grade tr = (Grade)vlist.elementAt(i);
%>
        <tr>
            <td align="center">
                <%=tr.getCId()%><input type="hidden" name="cid" value="<%=tr.getCId()%>">
            </td>
            <td align="center">
                <%=tr.getCIdNo()%><input type="hidden" name="cidno" value="<%=tr.getCIdNo()%>">
            </td>
            <td align="center">
                <%=tr.getCName()%><input type="hidden" name="cname" value="<%=tr.getCName()%>">
            </td>
            <td align="center">
                <%=tr.getSId()%><input type="hidden" name="sid" value="<%=tr.getSId()%>">
            </td>
            <td align="center">
                <%=tr.getSName()%><input type="hidden" name="sname" value="<%=tr.getSName()%>">
            </td>
            <td align="center">
                <input type="text" name="score" size="5" value="<%=tr.getEScore()%>">
            </td>
            <td align="center">
                <% if (tr.getEGrade() == null) {%>
                    &nbsp;
                <% } else { %>
                    <%=tr.getEGrade()%>
                <% } %>
            </td>
        </tr>
    <% } %>
    <input type="hidden" name="counter" value="<%=vlist.size()%>">
    <input type="hidden" name="p_id" size="30" value="<%=session_id%>">
    <input type="hidden" name="year" value="<%=year%>">
    <input type="hidden" name="semester" size="30" value="<%=semester%>">
</table><br>
<table align="center">
    <tr>
        <td>
            <input type="submit" name="submit" value="성적일괄등록">
        </td>
    </tr>
</table>
</form><br><br>
<form method="post" action="grade_verify.jsp" >
    <table width="30%" align="center">
        <tr>
            <td>
                <input type="text" name="year" value="<%= year %>" size=4> 년도
                <input type="text" name="semester" value="<%= semester %>" size=1> 학기
                <input type="submit" name="submit" value="조회">
            </td>
        </tr>
    </table>
</form>
</body>
</html>

```

#### [grade\_verify.jsp]

```

<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.sql.*" %>
<html>
    <head>
        <title> 성적등록 </title>
    </head>
    <body>

```

**【5-②】**

```

<%
String year = request.getParameter("year");
String semester = request.getParameter("semester");
if (year.length()!=0 & semester.length()!=0){
    String url = "grade.jsp?year="+ year + "&semester=" + semester;
    response.sendRedirect(url);
}else{
    year = Integer.toString(gradeMgr.getCurrentYear());
    semester = Integer.toString(gradeMgr.getCurrentSemester());
    String url = "grade.jsp?year="+ year + "&semester=" + semester;
}%>
<script>
    alert("년도 혹은 학기를 입력해주세요.");
    location.href = "<%=url%>";
</script>
<% } %>
</body>
</html>

```

### [update\_score\_verify.jsp]

```

<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ page import="java.sql.*, eduBean.*" %>
<html>
<head>
    <title> 성적등록 </title>
</head>
<body>
    [5- ③]

    <script>
        var message = "";
    <%
        String sMessage;

        request.setCharacterEncoding("utf-8");
        int year = Integer.parseInt(request.getParameter("year"));
        int semester = Integer.parseInt(request.getParameter("semester"));

        String result = null;
        String cid[] = request.getParameterValues("cid");
        String cidnos[] = request.getParameterValues("cidno");
        int [] cidno = new int [cidnos.length];
        for(int i = 0; i < cidnos.length; i++) { cidno[i] = Integer.parseInt(cidnos[i]); }
        String cname[] = request.getParameterValues("cname");
        String sid[] = request.getParameterValues("sid");
        String sname[] = request.getParameterValues("sname");
        String scores[] = request.getParameterValues("score");
        String scounter = request.getParameter("counter");
        int counter = Integer.parseInt(scounter);

        String dburl = "jdbc:oracle:thin:@210.94.199.20:1521:DBLAB";
        String user = "아이디"; // 본인 아이디(ex.FE0000000000)
        String passwd = "패스워드"; // 본인 패스워드(ex.FE0000000000)
        String dbdriver = "oracle.jdbc.driver.OracleDriver";

        Connection myConn = null;
        CallableStatement cstmt = null;

        try {
            [5- ④]
            if (result.equals("20002")){
                result = "등록 불가(사유 : 현재 학기 성적만 입력 가능합니다.)";
            }
            message = message + "<%=result%>" + "Wn"
            break;
        } else if(result.equals("20003")){
            result = "등록 불가(사유 : 학기 중에만 성적 입력 가능합니다.)";
            message = message + "<%=result%>" + "Wn"
            break;
        } else {
            result = cname[i] + "-" + sname[i] + "-" + result;
            message = message + "<%=result%>" + "Wn"
        }
    }
    </script>

```

```

        <%
            }
        } catch(SQLException ex) {
            System.err.println("SQLException: " + ex.getMessage());
        }
    %>
    alert(message);
    location.href = "grade.jsp?year=<%=year%>&semester=<%=semester%>";
</script>

</body>
</html>

```

## [GradeMgr.java]

```

package eduBean;

import java.sql.*;
import java.util.*;
import javax.sql.*;
import oracle.jdbc.driver.*;
import oracle.jdbc.pool.*;
import eduBean.*;

public class GradeMgr {

    private OracleConnectionPoolDataSource ocpds = null;
    private PooledConnection pool = null;

    public GradeMgr() {
        try{
            ocpds = new OracleConnectionPoolDataSource();

            ocpds.setURL("jdbc:oracle:thin:@210.94.199.20:1521:DBLAB");
            ocpds.setUser("아이디"); // 본인 아이디(ex.FE0000000000)
            ocpds.setPassword("패스워드"); // 본인 패스워드(ex.FE0000000000)

            pool = ocpds.getPooledConnection();
        } catch(Exception e) {
            System.out.println("Error : Connection Failed");
        }
    }

    // 현재 년도/학기에 담당하고 있는 강의과목의 학생 성적 조회
    public Vector getScoreList(String p_id, int nYear, int nSemester) {
        Connection conn = null;
        PreparedStatement pstmt = null;
        ResultSet rs = null;
        Vector vecList = new Vector();

        try {
            【5- ⑤】

            while(rs.next()){
                Grade tr = new Grade();
                tr.setCid(rs.getString("cid"));
                tr.setCidNo(rs.getInt("cid_no"));
                tr.setCName(rs.getString("cname"));
                tr.setSid(rs.getString("sid"));
                tr.setSName(rs.getString("sname"));
                tr.setEScore(rs.getInt("escore"));
                tr.setEGrade(rs.getString("egrade"));
                vecList.add(tr);
            }
            pstmt.close();
            conn.close();
        } catch(Exception ex) {
            System.out.println("Exception" + ex);
        }
        return vecList;
    }

    // 성적등록
    public String setScore(String score, String sid, String cid, int cidno, int year, int semester) {
        String sMessage = null;
    }

```

```

Connection conn = null;
CallableStatement cstmt = null;

try {
    【5-⑥】
} catch (SQLException ex) {
    if (ex.getErrorCode() == 20002) {
        sMessage = Integer.toString(ex.getErrorCode());
    } else if (ex.getErrorCode() == 20003) {
        sMessage = Integer.toString(ex.getErrorCode());
    }
    System.out.println("Exception" + ex);
}
return sMessage;
}

// 현재 년도 조회
public int getCurrentYear() {
    int nYear = 0;
    Connection conn = null;
    CallableStatement cstmt = null;

    try {
        【5-⑦】
    } catch (Exception ex) {
        System.out.println("Exception" + ex);
    }
    return nYear;
}

// 현재 학기 조회
public int getCurrentSemester() {
    int nSemester = 0;
    Connection conn = null;
    CallableStatement cstmt = null;

    try {
        【5-⑧】
    } catch (Exception ex) {
        System.out.println("Exception" + ex);
    }

    return nSemester;
}
}

```

### [Grade.java]

```

package eduBean;

public class Grade {
    private String c_id;
    private int c_id_no;
    private String c_name;
    private String s_id;
    private String s_name;
    private int e_score;
    private String e_grade;

    public Grade() {
        c_id = null;
        c_id_no = 0;
        c_name = null;
        s_id = null;
        s_name = null;
        e_score = 0;
        e_grade = null;
    }

    public void setCId(String c_id) {
        this.c_id = c_id;
    }

    public void setCIdNo(int c_id_no) {
        this.c_id_no = c_id_no;
    }
}

```

```

    public void setCName(String c_name) {
        this.c_name = c_name;
    }

    public void setSId(String s_id) {
        this.s_id = s_id;
    }

    public void setSName(String s_name) {
        this.s_name = s_name;
    }

    public void setEScore(int e_score) {
        this.e_score = e_score;
    }

    public void setEGrade(String e_grade) {
        this.e_grade = e_grade;
    }

    public String getCId() {
        return c_id;
    }

    public int getCIdNo() {
        return c_id_no;
    }

    public String getCName() {
        return c_name;
    }

    public String getSId() {
        return s_id;
    }

    public String getSName() {
        return s_name;
    }

    public int getEScore() {
        return e_score;
    }

    public String getEGrade() {
        return e_grade;
    }
}

```

#### [Date2GradeYear.sql]

```
CREATE OR REPLACE FUNCTION DATE2GRADEYEAR
```

**[5- 9]**

```
END;
```

#### [Date2GradeSemester.sql]

```
CREATE OR REPLACE FUNCTION DATE2GRADESEMESTER
```

**[5- 10]**

```
END;
```

#### [ChangeGrade.sql]

```
CREATE OR REPLACE FUNCTION CHANGEGRADE
```

**[5- 11]**

```
END;
```

#### [InsertGrade.sql]

```
CREATE OR REPLACE PROCEDURE INSERTGRADE
```

**[5- 12]**

```
END;
```

## <<제출 방식>>

- 제출 기한 : 2020년 12월 10일(목) 12:00(정오)
- 제출 형식 : 압축해서 하나의 파일로 제출
- 제출 방법 : e-Class 과제 란을 통해 제출
  - \* 만일 e-Class 시스템 장애 시 이메일 sein3051@dongguk.edu 제출
  - \* 단, 메일 제출 시 제출 기한 내 수신된 것에 한하여 인정
- 제출 내용 : ROOT 폴더 압축 파일(jsp 파일, Java 파일, sql 파일이 모두 저장되어 있어야 함)
  - \* 제공한 소스코드 파일에서 답안을 작성하여 제출
  - \* 제공한 소스코드 파일명 및 내용은 수정 불가
  - \* 제공한 소스코드 파일 외에 다른 파일 추가 불가

### [작성 방법]

- ◆ jsp 파일 (제출하는 ROOT 폴더 안에 저장되어 있어야 함)
  - 제공한 소스코드(.jsp)는 수정 불가하고, 문제 설명에 따라 번호가 표시된 빈칸에 소스코드를 작성하여 완성해야 함(소스코드에 주석으로 번호가 표시되어 있음)
    - > 소스코드 파일명을 수정할 경우 채점 불가
    - > 제공한 소스코드 파일 외에 다른 파일을 추가할 경우 채점 불가
- ◆ Java 파일 (제출하는 ROOT 폴더 - WEB-INF 폴더 - classes 폴더(별도 생성) 안에 저장되어 있어야 함)
  - 제공한 소스코드(.java)는 수정 불가하고, 문제 설명에 따라 번호가 표시된 빈칸에 소스코드를 작성하여 완성해야 함(소스코드에 주석으로 번호가 표시되어 있음)
  - Java 파일(.java)은 컴파일하여 class 파일을 생성하고 기존 Java 파일과 함께 제출할 것
    - > 소스코드 파일명을 수정할 경우 채점 불가
    - > 제공한 소스코드 파일 외에 다른 파일을 추가할 경우 채점 불가
    - > 컴파일하지 않을 경우 채점 불가
- ◆ sql 파일 (제출하는 ROOT 폴더 - SQL 폴더(별도 생성) 안에 저장되어 있어야 함)
  - 제공한 소스코드(.sql)는 수정 불가하고, 문제 설명에 따라 번호가 표시된 빈칸에 소스코드를 작성하여 완성해야 함(소스코드에 주석으로 번호가 표시되어 있음)
  - 테이블 스키마 구조 변경 관련 답안은 제출하지 않아도 됨
  - Oracle의 DBLAB 서버에 학생 계정으로 연결하여 데이터베이스(테이블, 프로시저, 함수, 트리거)를 생성할 것
    - ▶ Oracle 연결 계정 : 아이디 - FE학번 / 비밀번호 - FE학번 / 서버명 : DBLAB
      - \* 비밀번호 변경 구문(필요 시 사용) : ALTER USER FE학번 IDENTIFIED BY 변경할 비밀번호;
      - \* 해당 계정은 기말고사 대체 과제 시행 기간 동안에만 사용 가능함
  - 문제에서 요구하는 프로시저, 함수, 트리거는 문제 설명에 따라 번호가 표시된 빈칸을 작성하여 생성해야 함
    - > 소스코드 파일명을 수정할 경우 채점 불가
    - > Oracle의 DBLAB 서버에 생성한 데이터베이스(테이블, 프로시저, 함수, 트리거)가 없을 경우 채점 불가
    - > 제공한 프로시저명, 함수명, 트리거명을 수정할 경우 채점 불가
    - > 제출한 파일(.sql)과 Oracle 서버에 데이터베이스(테이블, 프로시저, 함수, 트리거)가 다른 경우 채점 불가



※ 주의사항

- \* 기말고사 대체 과제는 30점 만점으로 점수 부여
- \* 제출 기한 위반 시 감점 적용 (1시간 이내 15점)
- 기말고사 대체이므로 마감 후 1시간 이내로 과제 결과를 제출하지 않으면 성적 F를 부여함
- 도서관 인터넷 검색을 참고하여 과제 수행 가능
- 타인의 도움을 받지 않고 스스로 과제를 수행하여야 함  
(데이터베이스 프로그래밍 관련 학생 간 도움이나 자료 공유 불가)
- 학생 간 자료 공유 적발 시 미제출 처리함 (성적 F) (교내 표절 검사 시스템으로 확인 가능)
- 한 장소에 모여서 함께 과제를 수행하면 안 됨 (최근 다른 대학에서 문제 발생)
- 한 장소에 모여서 함께 과제를 수행한 사실 확인 시 미제출 처리함 (성적 F)
- 과제에 대한 질문은 조교에게 문의 (02-2290-1423, sein3051@dongguk.edu)
  - \* 과제의 정답과 관련된 질문에는 답변 불가
  - \* 전화 문의 가능 일시 : 12월 9일(수) 10:00 ~ 18:00, 12월 10일(목) 10:00 ~ 12:00
  - \* 이메일 문의 가능 일시 : 12월 9일(수) 10:00 ~ 21:00, 12월 10일(목) 10:00 ~ 12:00

※ 교무학생처의 지침에 따른 부정행위 방지 노력 안내

- 부정행위 적발 시 학생의 해당 시험은 무효로 하며, 학생상벌위원회 회부 가능

※ 참고 : 부정행위 유형

- 1) 타 학생과 공모하여 답안지를 작성하는 경우
- 2) 대리시험(위탁자 및 피위탁자 모두 부정행위에 해당)
- 3) 교강사가 허용하지 않는 행위를 하는 경우
- 4) 기타 정상적인 시험응시 행위로 볼 수 없는 경우
- 5) 대체과제에서 제시한 부정행위 관련 준수 사항을 지키지 않은 경우

※ 기말고사 대체 과제 답안을 제출할 경우, 다음의 내용에 동의한 것으로 간주함

2020-2 데이터베이스프로그래밍 기말고사 대체 과제

<부정행위 방지 서약>

나는 정직하게 기말고사(대체 과제)에 임할 것이며, 절대로 부정행위를 하지 않을 것을 서약합니다.

2020년 12월 일 요일

학과/전공 : 학년 : 학번 : 성명 :