

```

1  1. 환경설정
2      1)Copy mybatis-3.4.6.jar to /WEB-INF/lib
3      2)Copy odbc6.jar to /WEB-INF/lib
4      3)Copy jstl.jar, standard.jar to /WEB-INF/lib
5
6
7  2. Create user and table
8      ALTER SESSION SET "_ORACLE_SCRIPT"=true;
9      CREATE USER sungjukmgmt IDENTIFIED BY sungjukmgmt
10     DEFAULT TABLESPACE USERS
11     TEMPORARY TABLESPACE TEMP;
12
13     ALTER USER sungjukmgmt
14     DEFAULT TABLESPACE USERS
15     QUOTA UNLIMITED ON USERS;
16
17     GRANT resource, connect TO sungjukmgmt;
18
19     CONNECT sungjukmgmt/sungjukmgmt;
20
21     CREATE TABLE Student
22     (
23         hakbun CHAR(4),
24         irum   VARCHAR2(20) NOT NULL,
25         kor    NUMBER(3)    NOT NULL,
26         eng    NUMBER(3)    NOT NULL,
27         mat    NUMBER(3)    NOT NULL,
28         edp    NUMBER(3)    NOT NULL,
29         sum    NUMBER(3)    DEFAULT 0,
30         avg    NUMBER(5, 2) DEFAULT 0.0,
31         grade  CHAR(1),
32         CONSTRAINT student_hakbun_pk PRIMARY KEY(hakbun)
33     );
34
35
36  3. src/dbinfo.properties
37     db.driver=oracle.jdbc.driver.OracleDriver
38     db.url=jdbc:oracle:thin:@192.168.56.2:1521:XE
39     db.username=sungjukmgmt
40     db.password=sungjukmgmt
41
42
43  4. src/com.javasoft.libs.models.StudentVO.java
44     package com.javasoft.libs.models;
45
46     public class StudentVO {
47         private String hakbun;
48         private String irum;
49         private int kor,eng,mat,edp,sum;
50         private double avg;
51         private char grade;
52         public String getHakbun() {
53             return hakbun;
54         }
55         public void setHakbun(String hakbun) {
56             this.hakbun = hakbun;
57         }
58         public String getIrum() {
59             return irum;
60         }
61         public void setIrum(String irum) {
62             this.irus = irum;
63         }
64         public int getKor() {
65             return kor;
66         }
67         public void setKor(int kor) {

```

```

68         this.kor = kor;
69     }
70     public int getEng() {
71         return eng;
72     }
73     public void setEng(int eng) {
74         this.eng = eng;
75     }
76     public int getMat() {
77         return mat;
78     }
79     public void setMat(int mat) {
80         this.mat = mat;
81     }
82     public int getEdp() {
83         return edp;
84     }
85     public void setEdp(int edp) {
86         this.edp = edp;
87     }
88     public int getSum() {
89         return sum;
90     }
91     public void setSum(int sum) {
92         this.sum = sum;
93     }
94     public double getAvg() {
95         return avg;
96     }
97     public void setAvg(double avg) {
98         this.avg = avg;
99     }
100    public char getGrade() {
101        return grade;
102    }
103    public void setGrade(char grade) {
104        this.grade = grade;
105    }
106
107    public String toString() {
108        return String.format("{\\"hakbun\":"\"%s\\", \\"irum\":"\"%s\\", \\"kor\":"%d, \\"eng\":"%d,
109                                \\"mat\":"%d, \\"edp\":"%d, \\"sum\":"%d, \\"avg\":"%.2f, \\"grade\":"\"%c\\",
110                                this.hakbun, this.irum, this.kor, this.eng, this.mat, this.edp,
111                                this.sum, this.avg, this.grade);
112    }
113 }

```

```

114 5. src/com.javasoft.libs.models/Student.xml
115 <?xml version="1.0" encoding="UTF-8" ?>
116 <!DOCTYPE mapper
117     PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
118     "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
119
120 <mapper namespace="Student">
121
122 </mapper>
123
124
125 6. src/mybatis-config.xml
126 <?xml version="1.0" encoding="UTF-8" ?>
127 <!DOCTYPE configuration
128     PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
129     "http://mybatis.org/dtd/mybatis-3-config.dtd">
130
131 <configuration>
132     <properties resource="dbinfo.properties"/>

```

```

133 <typeAliases>
134   <typeAlias type="com.javasoft.libs.models.StudentVO" alias="StudentVO"/>
135 </typeAliases>
136
137 <environments default="development">
138   <environment id="development">
139     <transactionManager type="JDBC"/>
140     <dataSource type="POOLED">
141       <property name="driver" value="${db.driver}"/>
142       <property name="url" value="${db.url}"/>
143       <property name="username" value="${db.username}"/>
144       <property name="password" value="${db.password}"/>
145     </dataSource>
146   </environment>
147 </environments>
148 <mappers>
149   <mapper resource="com/javasoft/libs/models/Student.xml"/>
150 </mappers>
151 </configuration>

```

7. src/com.javasoft.libs.dao/SqlSessionFactoryBean.java

```

155 package com.javasoft.libs.dao;
156
157 import java.io.FileNotFoundException;
158 import java.io.IOException;
159 import java.io.Reader;
160
161 import org.apache.ibatis.io.Resources;
162 import org.apache.ibatis.session.SqlSession;
163 import org.apache.ibatis.session.SqlSessionFactory;
164 import org.apache.ibatis.session.SqlSessionFactoryBuilder;
165
166 public class SqlSessionFactoryBean {
167   private static SqlSessionFactory sessionFactory = null;
168   static {
169     try {
170       if(sessionFactory == null) {
171         Reader reader = Resources.getResourceAsReader("mybatis-config.xml");
172         sessionFactory = new SqlSessionFactoryBuilder().build(reader);
173       }
174     } catch(FileNotFoundException ex) {
175       System.out.println("File Not Found");
176     } catch(IOException ex) {
177       System.out.println(ex);
178     }
179   }
180   public static SqlSession getInstance() {
181     return sessionFactory.openSession();
182   }
183 }

```

8. Test

```

187 -index.jsp
188 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
189 <%@ page import="com.javasoft.libs.dao.SqlSessionFactoryBean" %>
190 <%@ page import="org.apache.ibatis.session.SqlSession" %>
191
192 <%
193   SqlSession sqlSession = SqlSessionFactoryBean.getInstance();
194 %>

```

9. src/com.javasoft.libs.dao/StudentDAO.java

```

198 package com.javasoft.libs.dao;

```

```

200 public interface StudentDAO {
201 }
202
203
204 10. src/com.javasoft.libs.dao/StudentDAOImpl.java
205 package com.javasoft.libs.dao;
206
207 public class StudentDAOImpl implements StudentDAO {
208     private SqlSession mybatis;
209
210     public StudentDAOImpl() {
211         this.mybatis = SqlSessionFactoryBean.getInstance();
212     }
213 }
214
215
216 11. Select All
217 1)Student.xml
218     <resultMap type="StudentVO" id="selectMap">
219         <result property="hakbun" column="hakbun"/>
220         <result property="irum" column="irum"/>
221         <result property="kor" column="kor"/>
222         <result property="eng" column="eng"/>
223         <result property="mat" column="mat"/>
224         <result property="edp" column="edp"/>
225         <result property="sum" column="sum"/>
226         <result property="avg" column="avg"/>
227         <result property="grade" column="grade"/>
228     </resultMap>
229
230     <select id="selectAll" resultMap="selectMap">
231         SELECT * FROM Student ORDER BY sum DESC
232     </select>
233
234 2)src/com.javasoft.libs.dao/StudentDAO.java
235 package com.javasoft.libs.dao;
236
237 import java.util.List;
238 import com.javasoft.libs.models.StudentVO;
239
240 public interface StudentDAO {
241     List<StudentVO> selectAll();
242 }
243
244 3)src/com.javasoft.libs.dao/StudentDAOImpl.java
245 package com.javasoft.libs.dao;
246
247 import java.util.List;
248 import org.apache.ibatis.session.SqlSession;
249 import com.javasoft.libs.models.StudentVO;
250
251 public class StudentDAOImpl implements StudentDAO {
252     private SqlSession mybatis;
253
254     public StudentDAOImpl() {
255         this.mybatis = SqlSessionFactoryBean.getInstance();
256     }
257
258     @Override
259     public List<StudentVO> selectAll() {
260         return this.mybatis.selectList("Student.selectAll");
261     }
262 }
263
264 4)index.jsp
265 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
266 <%@ page import="java.util.List" %>

```

```

267 <%@ page import="com.javasoft.libs.dao.*" %>
268 <%@ page import="com.javasoft.libs.models.StudentVO" %>
269
270 <%
271     StudentDAO dao = new StudentDAOImpl();
272     List<StudentVO> list = dao.selectAll();
273     String str = "";
274     for(StudentVO student : list){
275         str += student.toString() + ",";
276     }
277     if(str.length() > 0) str = str.substring(0, str.length() - 1);
278 %>
279 {
280     "code" : "success",
281     "data" : [<%=str %>]
282 }
283
284 5)css/style.css
285 @charset "UTF-8";
286 h1{
287     text-align:center;
288 }
289
290 table:nth-of-type(1){
291     width:600px;margin-left:auto;margin-right:auto;
292     border-width:0px;
293 }
294 table:nth-of-type(1) td{
295     border-width:0px;
296 }
297 table:nth-of-type(1) td:nth-of-type(2){
298     text-align:right;
299 }
300
301 table:nth-of-type(2){
302     width:600px;margin-left:auto;margin-right:auto;
303     border : 1px solid #fff;
304     border-collapse: collapse;
305 }
306 table:nth-of-type(2) th, td{
307     border : 1px solid #fff;
308 }
309
310 6)index.html
311 <!DOCTYPE html>
312 <html>
313     <head>
314         <meta charset="UTF-8">
315         <title>성적 관리 프로그램</title>
316         <link rel="stylesheet" type="text/css" href="css/bootstrap.css" />
317         <link rel="stylesheet" type="text/css" href="css/style.css" />
318         <script src="js/jquery-1.12.4.js"></script>
319         <script>
320             $(function(){
321                 $.get("index.jsp", function(data){
322                     var obj = JSON.parse(data);
323                     var students = obj.data;
324                     $('#count').text(students.length);
325                     var str = "";
326                     for(var i = 0 ; i < students.length ; i++){
327                         str += "<tr>";
328                         str += "<td><a href='detail.html?hakbun=" + students[i].hakbun + "'>" +
329                             students[i].hakbun + "</a></td>" +
330                             "<td>" + students[i].irum + "</td>" +
331                             "<td>" + students[i].kor + "</td>" +
332                             "<td>" + students[i].eng + "</td>" +
333                             "<td>" + students[i].mat + "</td>" +

```

```

333         "<td>" + students[i].edp + "</td>";
334         str += "</tr>";
335     }
336     $('#result').html(str);
337 });
338 });
339 </script>
340 </head>
341 <body>
342     <h1 class="text-center">Bit High School Students List</h1>
343     <table class="table">
344         <tr>
345             <td>[<a href="input.html">새로 입력</a>]</td>
346             <td class="text-right">[<span id="count"></span>명 입력]</td>
347         </tr>
348     </table>
349     <table class="table table-hover">
350         <thead>
351             <tr class="info">
352                 <th>학번</th> <th>이름</th> <th>국어</th>
353                 <th>영어</th> <th>수학</th> <th>전산</th>
354             </tr>
355         </thead>
356         <tbody id="result">
357         </tbody>
358     </table>
359 </body>
360 </html>

```

12. Insert

1) Student.xml

```

365 <insert id="insert" parameterType="StudentVO">
366     INSERT INTO Student(hakbun, irum, kor, eng, mat, edp, sum, avg, grade)
367     VALUES(#{hakbun}, #{irum}, #{kor}, #{eng}, #{mat}, #{edp}, #{sum}, #{avg}, #{grade})
368 </insert>

```

2) StudentDAO.java

```

371 void insert(StudentVO studentVO);

```

3) StudentDAOImpl.java

```

374 @Override
375 public void insert(StudentVO studentVO) {
376     this.mybatis.insert("Student.insert", studentVO);
377     this.mybatis.commit();
378 }

```

4) input.jsp

```

381 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
382 <%@ page import="com.javasoft.libs.models.StudentVO" %>
383 <%@ page import="com.javasoft.libs.dao.*" %>
384 <%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
385 <fmt:requestEncoding value="utf-8"/>
386 <jsp:useBean id="stu" class="com.javasoft.libs.models.StudentVO" />
387 <jsp:setProperty property="*" name="stu" />
388 <%
389     calc(stu);
390     StudentDAO dao = new StudentDAOImpl();
391     dao.insert(stu);
392     response.sendRedirect("/SungjukMgmt/");
393 %>
394
395 <%!
396     private void calc(StudentVO stu){
397         int sum = stu.getKor() + stu.getEng() + stu.getMat() + stu.getEdp();
398         double avg = sum / 4.;
399         char grade = (avg <= 100 && avg >= 90) ? 'A' :

```

```

400             (avg < 90 && avg >= 80) ? 'B' :
401             (avg < 80 && avg >= 70) ? 'C' :
402             (avg < 70 && avg >= 60) ? 'D' : 'F';
403         stu.setSum(sum); stu.setAvg(avg); stu.setGrade(grade);
404     }
405     %>
406

```

5)input.html

```

408     <!DOCTYPE html>
409     <html>
410     <head>
411     <meta charset="UTF-8">
412     <title>Insert title here</title>
413     </head>
414     <body>
415     <h1>새 학생 데이터 입력</h1>
416     <form action="input.jsp" method="post">
417     <ul style="list-style-type:none">
418     <li>학번 : <input type="text" name="hakbun" /></li>
419     <li>이름 : <input type="text" name="irum" /></li>
420     <li>국어 : <input type="text" name="kor" /></li>
421     <li>영어 : <input type="text" name="eng" /></li>
422     <li>수학 : <input type="text" name="mat" /></li>
423     <li>전산 : <input type="text" name="edp" /></li>
424     </ul>
425     <input type="submit" value="입력하기" />
426     </form>
427     </body>
428     </html>
429

```

13. Select One

1)Student.xml

```

433     <select id="selectOne" parameterType="String" resultType="StudentVO">
434     SELECT * FROM Student WHERE hakbun = #{hakbun}
435     </select>
436

```

2)src/com.javasoft.libs.dao/StudentDAO.java

```

438     StudentVO selectOne(String hakbun);
439

```

3)src/com.javasoft.libs.dao/StudentDAOImpl.java

```

441     @Override
442     public StudentVO selectOne(String hakbun) {
443     return this.mybatis.selectOne("Student.selectOne", hakbun);
444     }
445

```

4)detail.jsp

```

447     <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
448     <%@ page import="com.javasoft.libs.dao.*" %>
449     <%@ page import="com.javasoft.libs.models.StudentVO" %>
450
451     <%
452     String hakbun = request.getParameter("hakbun");
453     StudentDAO dao = new StudentDAOImpl();
454     StudentVO student = dao.selectOne(hakbun);
455     %>
456     {
457     "code":"success",
458     "data":<%=student %>
459     }
460

```

5)detail.html

```

462     <!DOCTYPE html>
463     <html lang="en">
464     <head>
465     <meta charset="UTF-8">
466     <title>성적 정보 창</title>

```

```
<link rel="stylesheet" type="text/css" href="css/bootstrap.css" />
<script src="js/jquery-1.12.4.js"></script>
<script>
    $(function(){
        var hakbun = location.search.substring(1).split("=")[1];

        $("#btnDelete").bind("click", function(){
            if(confirm("정말 삭제하시겠습니까?")){
                location.href = "delete.jsp?hakbun=" + hakbun;
            }
        });

        $.post("detail.jsp", {hakbun:hakbun}, function(data){
            var obj = JSON.parse(data);
            var student = obj.data;
            $('#stuIrum').text(student.irus);
            $('#hakbun').val(student.hakbun);
            $('#irus').val(student.irus);
            $('#kor').val(student.kor);
            $('#eng').val(student.eng);
            $('#mat').val(student.mat);
            $('#edp').val(student.edp);
            $('#sum').val(student.sum);
            $('#avg').val(student.avg);
            $('#grade').val(student.grade);
        });
    });
</script>
</head>
<body>
    <h1><span id="stuIrum"></span>의 성적 데이터</h1>
    <form method="POST" action="update.jsp">
        <ul style="list-style-type:none">
            <li>학번 : <input type="text" name="hakbun" readonly id="hakbun" /></li>
            <li>이름 : <input type="text" name="irus" readonly id="irus" /></li>
            <li>국어 : <input type="number" name="kor" id="kor" /></li>
            <li>영어 : <input type="number" name="eng" id="eng" /></li>
            <li>수학 : <input type="number" name="mat" id="mat" /></li>
            <li>전산 : <input type="number" name="edp" id="edp" /></li>
            <li>총점 : <input type="number" name="sum" id="sum" readonly/></li>
            <li>평균 : <input type="number" name="avg" id="avg" readonly/></li>
            <li>학점 : <input type="text" name="grade" id="grade" readonly/></li>
        </ul>
        <input type="submit" value="수정하기" />&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~
        <input type="button" value="목록으로" onclick="javascript:history.back();" />&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~
        <input type="button" value="삭제하기" id="btnDelete"/>&nbsp;&nbsp;&nbsp;&nbsp;&~
    </form>
</body>
</html>
```

14. Update

1)Student.xml

```
<update id="update" parameterType="StudentVO">
    UPDATE Student SET kor=#{kor}, eng=#{eng}, mat=#{mat}, edp=#{edp},
    sum=#{sum},avg=#{avg},grade=#{grade}
    WHERE hakbun = #{hakbun}
</update>
```

2)StudentDAO.java

```
void update(StudentVO studentVO);
```

3) StudentDAOImpl.java

```
@Override
public void update(StudentVO studentVO) {
    this.mybatis.update("Student.update", studentVO);
    this.mybatis.commit();
}
```



```

534     }
535
536 4)update.jsp
537   <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
538   <%@ page import="com.javasoft.libs.models.StudentVO" %>
539   <%@ page import="com.javasoft.libs.dao.*" %>
540   <jsp:useBean id="stu" class="com.javasoft.libs.models.StudentVO" />
541   <jsp:setProperty name="stu" property="*" />
542   <%
543       calc(stu);
544       StudentDAO dao = new StudentDAOImpl();
545       dao.update(stu);
546       response.sendRedirect("/SungjukMgmt/");
547   %>
548
549
550   <%!
551       private void calc(StudentVO stu){
552           int sum = stu.getKor() + stu.getEng() + stu.getMat() + stu.getEdp();
553           double avg = sum / 4.;
554           char grade = (avg <= 100 && avg >= 90) ? 'A' :
555                       (avg < 90 && avg >= 80) ? 'B' :
556                       (avg < 80 && avg >= 70) ? 'C' :
557                       (avg < 70 && avg >= 60) ? 'D' : 'F';
558           stu.setSum(sum); stu.setAvg(avg); stu.setGrade(grade);
559       }
560   %>
561
562
563 15. Delete
564 1)Student.xml
565   <delete id="delete" parameterType="String">
566       DELETE FROM Student WHERE hakbun = #{hakbun}
567   </delete>
568
569 2)StudentDAO.java
570   void delete(String hakbun);
571
572 3)StudentDAOImpl.java
573   @Override
574   public void delete(String hakbun) {
575       this.mybatis.delete("Student.delete", hakbun);
576       this.mybatis.commit();
577   }
578
579 4)delete.jsp
580   <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
581   <%@ page import="com.javasoft.libs.dao.*" %>
582   <%
583       String hakbun = request.getParameter("hakbun");
584       StudentDAO dao = new StudentDAOImpl();
585       dao.delete(hakbun);
586       response.sendRedirect("/SungjukMgmt/");
587   %>
588
589
590 16. Stored Procedure - delete
591   CREATE OR REPLACE PROCEDURE sp_student_delete
592   (
593       v_hakbun    IN    Student.hakbun%TYPE
594   )
595   IS
596   BEGIN
597       DELETE FROM Student
598       WHERE hakbun = v_hakbun;
599       COMMIT;
600   END;

```

```

601 /
602
603 1)Student.xml
604     <parameterMap type="String" id="delParamMap">
605         <parameter property="hakbun" javaType="java.lang.String" mode="IN" jdbcType="CHAR"/>
606     </parameterMap>
607
608     <delete id="deleteSP" parameterType="String" parameterMap="delParamMap">
609         { call sp_student_delete(?) } <---여기서 {다음에 space를 한칸 혹은 두칸만 띄워야 한다.
610     </delete>
611
612 2)StudentDAOImpl.java
613     @Override
614     public void delete(String hakbun) {
615         this.mybatis.delete("Student.deleteSP", hakbun);
616     }
617
618 17. Stored Procedure - Insert
619 CREATE OR REPLACE PROCEDURE sp_student_insert
620 (
621     v_hakbun IN Student.hakbun%TYPE,
622     v_irm IN Student.irm%TYPE,
623     v_kor IN Student.kor%TYPE,
624     v_eng IN Student.eng%TYPE,
625     v_mat IN Student.mat%TYPE,
626     v_edp IN Student.edp%TYPE
627 )
628 IS
629     v_sum Student.sum%TYPE;
630     v_avg Student.avg%TYPE;
631     v_grade Student.grade%TYPE;
632 BEGIN
633     v_sum := v_kor + v_eng + v_mat + v_edp;
634     v_avg := v_sum / 4.;
635     v_grade := NULL;
636
637     CASE
638         WHEN v_avg >= 90 THEN v_grade := 'A';
639         WHEN v_avg >= 80 THEN v_grade := 'B';
640         WHEN v_avg >= 70 THEN v_grade := 'C';
641         WHEN v_avg >= 60 THEN v_grade := 'D';
642         ELSE v_grade := 'F';
643     END CASE;
644
645     INSERT INTO Student(hakbun, irm, kor, eng, mat, edp, sum, avg, grade)
646     VALUES(v_hakbun, v_irm, v_kor, v_eng, v_mat, v_edp, v_sum, v_avg, v_grade);
647     COMMIT;
648 END;
649 /
650
651 1)Student.xml
652     <parameterMap type="StudentVO" id="insertParamMap">
653         <parameter property="hakbun" javaType="java.lang.String" jdbcType="CHAR" mode="IN"/>
654         <parameter property="irm" javaType="java.lang.String" jdbcType="VARCHAR" mode="IN"/>
655         <parameter property="kor" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
656         <parameter property="eng" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
657         <parameter property="mat" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
658         <parameter property="edp" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
659     </parameterMap>
660
661     <insert id="insertSP" parameterMap="insertParamMap">
662         { call sp_student_insert(?,?,?,?,?) }
663     </insert>
664
665 2)StudentDAOImpl.java
666     @Override
667     public void insert(StudentVO studentVO) {

```

```

668         this.mybatis.insert("Student.insertSP", studentVO);
669     }
670
671
672 18. Stored Procedure - Update
673 CREATE OR REPLACE PROCEDURE sp_student_update
674 (
675     v_kor    IN    Student.kor%TYPE,
676     v_eng    IN    Student.eng%TYPE,
677     v_mat    IN    Student.mat%TYPE,
678     v_edp    IN    Student.edp%TYPE,
679     v_hakbun IN    Student.hakbun%TYPE
680 )
681 IS
682     v_sum    Student.sum%TYPE;
683     v_avg    Student.avg%TYPE;
684     v_grade  Student.grade%TYPE;
685 BEGIN
686     v_sum := v_kor + v_eng + v_mat + v_edp;
687     v_avg := v_sum / 4.;
688     v_grade := NULL;
689
690     CASE
691         WHEN v_avg >= 90 THEN v_grade := 'A';
692         WHEN v_avg >= 80 THEN v_grade := 'B';
693         WHEN v_avg >= 70 THEN v_grade := 'C';
694         WHEN v_avg >= 60 THEN v_grade := 'D';
695         ELSE v_grade := 'F';
696     END CASE;
697
698     UPDATE Student SET kor = v_kor, eng = v_eng, mat = v_mat, edp = v_edp,
699     sum = v_sum, avg = v_avg, grade = v_grade
700     WHERE hakbun = v_hakbun;
701     COMMIT;
702 END;
703 /
704
705 1)Student.xml
706 <update id="updateSP" parameterType="StudentVO">
707     { call sp_student_update(#{kor},#{eng},#{mat},#{edp},#{hakbun})}
708 </update>
709
710 2)StudentDAOImpl.java
711 @Override
712 public void update(StudentVO studentVO) {
713     this.mybatis.update("Student.updateSP", studentVO);
714 }
715
716
717 19. Stored Procedure - Select One
718 CREATE OR REPLACE PROCEDURE sp_student_select_one
719 (
720     v_hakbun IN    Student.hakbun%TYPE,
721     v_student OUT SYS_REFCURSOR
722 )
723 AS
724 BEGIN
725     OPEN v_student FOR
726     SELECT hakbun, irum, kor, eng, mat, edp, sum, avg, grade
727     FROM Student
728     WHERE hakbun = v_hakbun;
729 END;
730 /
731
732 1)Student.xml
733 <parameterMap id="studentOneMap" type="hashMap">
734     <parameter property="hakbun" javaType="String" jdbcType="CHAR" mode="IN"/>

```

```

735         <parameter property="result" javaType="ResultSet" jdbcType="CURSOR" mode="OUT"
736         resultMap="selectMap"/>
737     </parameterMap>
738     <select id="selectOneSP" parameterMap="studentOneMap" statementType="CALLABLE">
739         { call sp_student_select_one(?,?) }
740     </select>
741
742 2)StudentDAO.java
743     void selectOneSP(Map map);
744
745 3)StudentDAOImpl.java
746     @Override
747     public void selectOneSP(Map map){
748         this.mybatis.selectOne("Student.selectOneSP", map);
749     }
750
751 4)detail.jsp
752     <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
753     <%@ page import="com.javasoft.libs.dao.*" %>
754     <%@ page import="com.javasoft.libs.models.StudentVO" %>
755     <%@ page import="java.util.HashMap, java.util.Map, java.util.List" %>
756
757     <%
758         String hakbun = request.getParameter("hakbun");
759         StudentDAO dao = new StudentDAOImpl();
760         Map<String, Object> map = new HashMap<String, Object>();
761         map.put("hakbun", hakbun);
762         dao.selectOneSP(map);
763         List<StudentVO> list = (List<StudentVO>)map.get("result");
764         StudentVO student = list.get(0);
765     %>
766     {
767         "code":"success",
768         "data":<%=student %>
769     }
770
771
772 20. Stored Procedure - Select All
773 CREATE OR REPLACE PROCEDURE sp_students
774 (
775     v_students OUT SYS_REFCURSOR
776 )
777 AS
778 BEGIN
779     OPEN v_students FOR
780     SELECT hakbun, irum, kor, eng, mat, edp, sum, avg, grade
781     FROM Student
782     ORDER BY sum DESC;
783 END;
784 /
785
786 1)Student.xml
787     <parameterMap id="studentsMap" type="hashMap">
788         <parameter property="results" javaType="ResultSet" jdbcType="CURSOR" mode="OUT"
789         resultMap="selectMap"/>
790     </parameterMap>
791     <select id="selectAllSP" statementType="CALLABLE" parameterMap="studentsMap">
792         { call sp_students(?) }
793     </select>
794
795 2)StudentDAO.java
796     void selectAllSP(Map map);
797
798 3)StudentDAOImpl.java
799     @Override

```

```

800     public void selectAllSP(Map map) {
801         this.mybatis.selectList("Student.selectAllSP", map);
802     }
803
804 4)index.jsp
805     <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
806     <%@ page import="java.util.List, java.util.Map, java.util.HashMap" %>
807     <%@ page import="com.javasoft.libs.dao.*" %>
808     <%@ page import="com.javasoft.libs.models.StudentVO" %>
809
810     <%
811         StudentDAO dao = new StudentDAOImpl();
812         Map<String, Object> map = new HashMap<String, Object>();
813         dao.selectAllSP(map);
814         List<StudentVO> list = (List<StudentVO>)map.get("results");
815         String str = "";
816         for(StudentVO student : list){
817             str += student.toString() + ",";
818         }
819         if(str.length() > 0) str = str.substring(0, str.length() - 1);
820     %>
821     {
822         "code" : "success",
823         "data" : [<%=str %>]
824     }

```