1. 환경설정

```
1)Copy mybatis-3.4.6.jar to /WEB-INF/lib
2)Copy odbc6.jar to /WEB-INF/lib
3)Copy jstl.jar, standard.jar to /WEB-INF/lib
```

2. Create user and table

```sql
ALTER SESSION SET "_ORACLE_SCRIPT"=true;
CREATE USER sungjukmgmt IDENTIFIED BY sungjukmgmt
DEFAULT TABLESPACE USERS
TEMPORARY TABLESPACE TEMP;

ALTER USER sungjukmgmt
DEFAULT TABLESPACE USERS
QUOTA UNLIMITED ON USERS;

GRANT  resource, connect   TO   sungjukmgmt;

CONNECT sungjukmgmt/sungjukmgmt;

CREATE TABLE Student
(
     hakbun  CHAR(4),
     irum    VARCHAR2(20)   NOT NULL,
     kor     NUMBER(3)      NOT NULL,
     eng     NUMBER(3)      NOT NULL,
     mat     NUMBER(3)      NOT NULL,
     edp     NUMBER(3)      NOT NULL,
     sum     NUMBER(3)      DEFAULT 0,
     avg     NUMBER(5, 2)   DEFAULT 0.0,
     grade   CHAR(1),
     CONSTRAINT student_hakbun_pk  PRIMARY KEY(hakbun)
);
```

3. src/dbinfo.properties

```
db.driver=oracle.jdbc.driver.OracleDriver
db.url=jdbc:oracle:thin:@192.168.56.2:1521:XE
db.username=sungjukmgmt
db.password=sungjukmgmt
```

4. src/com.javasoft.libs.models.StudentVO.java

```java
package com.javasoft.libs.models;

public class StudentVO {
    private String hakbun;
    private String irum;
    private int kor,eng,mat,edp,sum;
    private double avg;
    private char grade;
    public String getHakbun() {
        return hakbun;
    }
    public void setHakbun(String hakbun) {
        this.hakbun = hakbun;
    }
    public String getIrum() {
        return irum;
    }
    public void setIrum(String irum) {
        this.irum = irum;
    }
    public int getKor() {
        return kor;
    }
    public void setKor(int kor) {
        this.kor = kor;
    }
    public int getEng() {
        return eng;
    }
    public void setEng(int eng) {
        this.eng = eng;
    }
    public int getMat() {
        return mat;
    }
    public void setMat(int mat) {
        this.mat = mat;
    }
    public int getEdp() {
        return edp;
    }
    public void setEdp(int edp) {
        this.edp = edp;
    }
    public int getSum() {
        return sum;
    }
    public void setSum(int sum) {
        this.sum = sum;
    }
    public double getAvg() {
        return avg;
    }
    public void setAvg(double avg) {
        this.avg = avg;
    }
    public char getGrade() {
        return grade;
    }
    public void setGrade(char grade) {
        this.grade = grade;
    }

    public String toString() {
        return String.format("{\"hakbun\":\"%s\", \"irum\":\"%s\", \"kor\":%d, \"eng\":%d,
        \"mat\":%d, \"edp\":%d, \"sum\":%d, \"avg\":%.2f, \"grade\":\"%c\"}",
                                                this.hakbun, this.irum, this.kor, this.eng, this.mat, this.edp,
                                                this.sum, this.avg, this.grade);
    }
}
```

5. src/com.javasoft.libs.models/Student.xml

```
    <?xml version="1.0" encoding="UTF-8" ?>
    <!DOCTYPE mapper
        PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
            "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

    <mapper namespace="Student">

    </mapper>
```

6. src/mybatis-config.xml

```
    <?xml version="1.0" encoding="UTF-8" ?>
    <!DOCTYPE configuration
        PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
            "http://mybatis.org/dtd/mybatis-3-config.dtd">

    <configuration>
        <properties resource="dbinfo.properties"/>
        <typeAliases>
            <typeAlias type="com.javasoft.libs.models.StudentVO" alias="StudentVO"/>
        </typeAliases>

        <environments default="development">
            <environment id="development">
                <transactionManager type="JDBC"/>
                <dataSource type="POOLED">
                    <property name="driver" value="${db.driver}"/>
                    <property name="url" value="${db.url}"/>
                    <property name="username" value="${db.username}"/>
                    <property name="password" value="${db.password}"/>
                </dataSource>
            </environment>
        </environments>
        <mappers>
            <mapper resource="com/javasoft/libs/models/Student.xml"/>
        </mappers>
    </configuration>
```

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

```
package com.javasoft.libs.dao;

import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.Reader;

import org.apache.ibatis.io.Resources;
import org.apache.ibatis.session.SqlSession;
import org.apache.ibatis.session.SqlSessionFactory;
import org.apache.ibatis.session.SqlSessionFactoryBuilder;

public class SqlSessionFactoryBean {
    private static SqlSessionFactory sessionFactory = null;
    static {
        try {
            if(sessionFactory == null) {
                Reader reader = Resources.getResourceAsReader("mybatis-config.xml");
                sessionFactory = new SqlSessionFactoryBuilder().build(reader);
            }
        }catch(FileNotFoundException ex) {
            System.out.println("File Not Found");
        }catch(IOException ex) {
            System.out.println(ex);
        }
    }
    public static SqlSession getInstance() {
        return sessionFactory.openSession();
    }
}
```

8. Test

-index.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="com.javasoft.libs.dao.SqlSessionFactoryBean" %>
<%@ page import="org.apache.ibatis.session.SqlSession" %>

<%
    SqlSession sqlSession = SqlSessionFactoryBean.getInstance();
%>
```

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

```
package com.javasoft.libs.dao;
```

```
    public interface StudentDAO {
    }
```

10. src/com.javasoft.libs.dao/StudentDAOImpl.java

```
    package com.javasoft.libs.dao;

    public class StudentDAOImpl implements StudentDAO {
        private SqlSession mybatis;

        public StudentDAOImpl() {
            this.mybatis = SqlSessionFactoryBean.getInstance();
        }
    }
```

11. Select All

    1)Student.xml

```
        <resultMap type="StudentVO" id="selectMap">
            <result property="hakbun" column="hakbun"/>
            <result property="irum" column="irum"/>
            <result property="kor" column="kor"/>
            <result property="eng" column="eng"/>
            <result property="mat" column="mat"/>
            <result property="edp" column="edp"/>
            <result property="sum" column="sum"/>
            <result property="avg" column="avg"/>
            <result property="grade" column="grade"/>
        </resultMap>

        <select id="selectAll" resultMap="selectMap">
            SELECT * FROM Student ORDER BY sum DESC
        </select>
```

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

```
        package com.javasoft.libs.dao;

        import java.util.List;
        import com.javasoft.libs.models.StudentVO;

        public interface StudentDAO {
            List<StudentVO> selectAll();
        }
```

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

```
        package com.javasoft.libs.dao;

        import java.util.List;
        import org.apache.ibatis.session.SqlSession;
        import com.javasoft.libs.models.StudentVO;

        public class StudentDAOImpl implements StudentDAO {
            private SqlSession mybatis;

            public StudentDAOImpl() {
                this.mybatis = SqlSessionFactoryBean.getInstance();
            }

            @Override
            public List<StudentVO> selectAll() {
                return this.mybatis.selectList("Student.selectAll");
            }
        }
```

    4)index.jsp

```
        <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
        <%@ page import="java.util.List" %>
```

```
<%@ page import="com.javasoft.libs.dao.*" %>
<%@ page import="com.javasoft.libs.models.StudentVO" %>

<%
    StudentDAO dao = new StudentDAOImpl();
    List<StudentVO> list = dao.selectAll();
    String str = "";
    for(StudentVO student : list){
        str += student.toString() + ",";
    }
    if(str.length() > 0) str = str.substring(0, str.length() - 1);
%>
{
    "code" : "success",
    "data" : [<%=str %>]
}
```

5)css/style.css

```
@charset "UTF-8";
h1{
    text-align:center;
}

table:nth-of-type(1){
    width:600px;margin-left:auto;margin-right:auto;
    border-width:0px;
}
table:nth-of-type(1)  td{
    border-width:0px;
}
table:nth-of-type(1)  td:nth-of-type(2){
    text-align:right;
}

table:nth-of-type(2){
    width:600px;margin-left:auto;margin-right:auto;
    border : 1px solid #fff;
    border-collapse: collapse;
}
table:nth-of-type(2) th, td{
    border : 1px solid #fff;
}
```

6)index.html

```
<!DOCTYPE html>
<html>
    <head>
    <meta charset="UTF-8">
    <title>성적 관리 프로그램</title>
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css" />
    <link rel="stylesheet" type="text/css" href="css/style.css" />
    <script src="js/jquery-1.12.4.js"></script>
    <script>
        $(function(){
            $.get("index.jsp", function(data){
                var obj = JSON.parse(data);
                var students = obj.data;
                $('#count').text(students.length);
                var str = "";
                for(var i = 0 ; i < students.length ; i++){
                    str += "<tr>";
                    str += "<td><a href='detail.html?hakbun=" + students[i].hakbun + "'>" + students[i].hakbun + "</a></td>" +
                                "<td>" + students[i].irum + "</td>" +
                                "<td>" + students[i].kor + "</td>" +
                                "<td>" + students[i].eng + "</td>" +
                                "<td>" + students[i].mat + "</td>" +
                                "<td>" + students[i].edp + "</td>";
                    str += "</tr>";
                }
                $('#result').html(str);
            });
        });
    </script>
    </head>
    <body>
        <h1 class="text-center">Bit High School Students List</h1>
        <table class="table"">
            <tr>
                <td>[<a href="input.html">새로 입력</a>]</td>
                <td class="text-right">[<span id="count"></span>명 입력]</td>
            </tr>
        </table>
        <table class="table table-hover">
            <thead>
                <tr class="info">
                    <th>학번</th><th>이름</th><th>국어</th>
                    <th>영어</th><th>수학</th><th>전산</th>
                </tr>
            </thead>
            <tbody id="result">
            </tbody>
        </table>
    </body>
</html>
```

12. Insert

1)Student.xml

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

2)StudentDAO.java

```
void insert(StudentVO studentVO);
```

3)StudentDAOImpl.java

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

4)input.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="com.javasoft.libs.models.StudentVO" %>
<%@ page import="com.javasoft.libs.dao.*" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<fmt:requestEncoding value="utf-8"/>
<jsp:useBean id="stu" class="com.javasoft.libs.models.StudentVO" />
<jsp:setProperty property="*" name="stu" />
<%
    calc(stu);
    StudentDAO dao = new StudentDAOImpl();
    dao.insert(stu);
    response.sendRedirect("/SungjukMgmt/");
%>

<%!
    private void calc(StudentVO stu){
        int sum = stu.getKor() + stu.getEng() + stu.getMat() + stu.getEdp();
        double avg = sum / 4.;
        char grade = (avg <= 100 && avg >= 90) ? 'A' :
```

```
                                (avg < 90 && avg >= 80) ? 'B' :
                                    (avg < 80 && avg >= 70) ? 'C' :
                                        (avg < 70 && avg >= 60) ? 'D' : 'F';
        stu.setSum(sum);  stu.setAvg(avg);   stu.setGrade(grade);
    }
%>
```

5)input.html

```
<!DOCTYPE html>
<html>
    <head>
    <meta charset="UTF-8">
    <title>Insert title here</title>
    </head>
    <body>
        <h1>새 학생 데이터 입력</h1>
        <form action="input.jsp" method="post">
            <ul style="list-style-type:none">
                <li>학번 : <input type="text" name="hakbun" /></li>
                <li>이름 : <input type="text" name="irum" /></li>
                <li>국어 : <input type="text" name="kor" /></li>
                <li>영어 : <input type="text" name="eng" /></li>
                <li>수학 : <input type="text" name="mat" /></li>
                <li>전산 : <input type="text" name="edp" /></li>
            </ul>
            <input type="submit" value="입력하기" />
        </form>
    </body>
</html>
```

13. Select One

1)Student.xml

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

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

```
StudentVO selectOne(String hakbun);
```

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

```
@Override
public StudentVO selectOne(String hakbun) {
    return this.mybatis.selectOne("Student.selectOne", hakbun);
}
```

4)detail.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="com.javasoft.libs.dao.*" %>
<%@ page import="com.javasoft.libs.models.StudentVO" %>

<%
    String hakbun = request.getParameter("hakbun");
    StudentDAO dao = new StudentDAOImpl();
    StudentVO student = dao.selectOne(hakbun);
%>
{
    "code":"success",
    "data":<%=student %>
}
```

5)detail.html

```
<!DOCTYPE html>
<html lang="en">
<head>
     <meta charset="UTF-8">
     <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.irum);
            $('#hakbun').val(student.hakbun);
            $('#irum').val(student.irum);
            $('#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="irum" readonly id="irum" /></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;
        <input type="button" value="목록으로" onclick="javascript:history.back();"/>&nbsp;&nbsp;
        <input type="button" value="삭제하기" id="btnDelete"/>&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();
}
```

4)update.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="com.javasoft.libs.models.StudentVO" %>
<%@ page import="com.javasoft.libs.dao.*" %>
<jsp:useBean id="stu" class="com.javasoft.libs.models.StudentVO" />
<jsp:setProperty name="stu" property="*" />
<%
    calc(stu);
    StudentDAO dao = new StudentDAOImpl();
    dao.update(stu);
    response.sendRedirect("/SungjukMgmt/");
%>


<%!
    private void calc(StudentVO stu){
        int sum = stu.getKor() + stu.getEng() + stu.getMat() + stu.getEdp();
        double avg = sum / 4.;
        char grade = (avg <= 100 && avg >= 90) ? 'A' :
                        (avg < 90 && avg >= 80) ? 'B' :
                            (avg < 80 && avg >= 70) ? 'C' :
                                (avg < 70 && avg >= 60) ? 'D' : 'F';
        stu.setSum(sum);  stu.setAvg(avg);   stu.setGrade(grade);
    }
%>
```

15. Delete

1)Student.xml

```
<delete id="delete" parameterType="String">
    DELETE FROM Student WHERE hakbun = #{hakbun}
</delete>
```

2)StudentDAO.java

```
void delete(String hakbun);
```

3)StudentDAOImpl.java

```
@Override
public void delete(String hakbun) {
    this.mybatis.delete("Student.delete", hakbun);
    this.mybatis.commit();
}
```

4)delete.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="com.javasoft.libs.dao.*" %>
<%
    String hakbun = request.getParameter("hakbun");
    StudentDAO dao = new StudentDAOImpl();
    dao.delete(hakbun);
    response.sendRedirect("/SungjukMgmt/");
%>
```

16. Stored Procedure - delete

```
CREATE OR REPLACE PROCEDURE  sp_student_delete
(
     v_hakbun    IN     Student.hakbun%TYPE
)
IS
BEGIN
     DELETE FROM Student
     WHERE hakbun = v_hakbun;
     COMMIT;
END;
```

```
/
```

1)Student.xml

```
<parameterMap type="String" id="delParamMap">
    <parameter property="hakbun" javaType="java.lang.String" mode="IN" jdbcType="CHAR"/>
</parameterMap>

<delete id="deleteSP" parameterType="String" parameterMap="delParamMap">
    { call sp_student_delete(?)}  <---여기서 {다음에 space를 한칸 혹은 두칸만 띄워야 한다.
</delete>
```

2)StudentDAOImpl.java

```
@Override
public void delete(String hakbun) {
    this.mybatis.delete("Student.deleteSP", hakbun);
}
```

17. Stored Procedure - Insert

```
CREATE OR REPLACE PROCEDURE  sp_student_insert
(
     v_hakbun    IN      Student.hakbun%TYPE,
     v_irum    IN      Student.irum%TYPE,
     v_kor    IN      Student.kor%TYPE,
     v_eng    IN      Student.eng%TYPE,
     v_mat    IN      Student.mat%TYPE,
     v_edp    IN      Student.edp%TYPE
)
IS
     v_sum    Student.sum%TYPE;
     v_avg    Student.avg%TYPE;
     v_grade    Student.grade%TYPE;
BEGIN
    v_sum := v_kor + v_eng + v_mat + v_edp;
    v_avg := v_sum / 4.;
    v_grade := NULL;

    CASE
        WHEN v_avg >= 90 THEN v_grade := 'A';
        WHEN v_avg >= 80 THEN v_grade := 'B';
        WHEN v_avg >= 70 THEN v_grade := 'C';
        WHEN v_avg >= 60 THEN v_grade := 'D';
        ELSE v_grade := 'F';
    END CASE;

     INSERT INTO Student(hakbun, irum, kor, eng, mat, edp, sum, avg, grade)
     VALUES(v_hakbun, v_irum, v_kor, v_eng, v_mat, v_edp, v_sum, v_avg, v_grade);
     COMMIT;
END;
/
```

1)Student.xml

```
<parameterMap type="StudentVO" id="insertParamMap">
    <parameter property="hakbun" javaType="java.lang.String" jdbcType="CHAR" mode="IN"/>
    <parameter property="irum" javaType="java.lang.String" jdbcType="VARCHAR" mode="IN"/>
    <parameter property="kor" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
    <parameter property="eng" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
    <parameter property="mat" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
    <parameter property="edp" javaType="java.lang.Integer" jdbcType="INTEGER" mode="IN"/>
</parameterMap>

<insert id="insertSP" parameterMap="insertParamMap">
    { call sp_student_insert(?,?,?,?,?,?)}
</insert>
```

2)StudentDAOImpl.java

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

18. Stored Procedure - Update

```
CREATE OR REPLACE PROCEDURE sp_student_update
(
     v_kor      IN    Student.kor%TYPE,
     v_eng      IN    Student.eng%TYPE,
     v_mat      IN    Student.mat%TYPE,
     v_edp      IN    Student.edp%TYPE,
     v_hakbun      IN    Student.hakbun%TYPE
)
IS
    v_sum    Student.sum%TYPE;
    v_avg    Student.avg%TYPE;
    v_grade    Student.grade%TYPE;
BEGIN
    v_sum := v_kor + v_eng + v_mat + v_edp;
    v_avg := v_sum / 4.;
    v_grade := NULL;

    CASE
        WHEN v_avg >= 90 THEN v_grade := 'A';
        WHEN v_avg >= 80 THEN v_grade := 'B';
        WHEN v_avg >= 70 THEN v_grade := 'C';
        WHEN v_avg >= 60 THEN v_grade := 'D';
        ELSE v_grade := 'F';
    END CASE;

    UPDATE Student SET kor = v_kor, eng = v_eng, mat = v_mat, edp = v_edp,
    sum = v_sum, avg = v_avg, grade = v_grade
    WHERE hakbun = v_hakbun;
    COMMIT;
END;
/

1)Student.xml
    <update id="updateSP" parameterType="StudentVO">
        { call sp_student_update(#{kor},#{eng},#{mat},#{edp},#{hakbun})}
    </update>

2)StudentDAOImpl.java
    @Override
    public void update(StudentVO studentVO) {
        this.mybatis.update("Student.updateSP", studentVO);
    }
```

19. Stored Procedure - Select One

```
CREATE OR REPLACE PROCEDURE sp_student_select_one
(
     v_hakbun      IN    Student.hakbun%TYPE,
     v_student     OUT   SYS_REFCURSOR
)
AS
BEGIN
    OPEN v_student FOR
        SELECT hakbun, irum, kor, eng, mat, edp, sum, avg, grade
        FROM Student
        WHERE hakbun = v_hakbun;
END;
/

1)Student.xml
    <parameterMap id="studentOneMap" type="hashMap">
        <parameter property="hakbun" javaType="String" jdbcType="CHAR" mode="IN"/>
```

```xml
        <parameter property="result" javaType="ResultSet" jdbcType="CURSOR" mode="OUT" resultMap="selectMap"/>
    </parameterMap>

    <select id="selectOneSP" parameterMap="studentOneMap" statementType="CALLABLE">
        { call sp_student_select_one(?,?)}
    </select>
```

2)StudentDAO.java

```java
    void selectOneSP(Map map);
```

3)StudentDAOImpl.java

```java
    @Override
    public void selectOneSP(Map map){
        this.mybatis.selectOne("Student.selectOneSP", map);
    }
```

4)detail.jsp

```jsp
    <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
    <%@ page import="com.javasoft.libs.dao.*" %>
    <%@ page import="com.javasoft.libs.models.StudentVO" %>
    <%@ page import="java.util.HashMap, java.util.Map, java.util.List" %>

    <%
        String hakbun = request.getParameter("hakbun");
        StudentDAO dao = new StudentDAOImpl();
        Map<String, Object> map = new HashMap<String, Object>();
        map.put("hakbun", hakbun);
        dao.selectOneSP(map);
        List<StudentVO> list = (List<StudentVO>)map.get("result");
        StudentVO student = list.get(0);
    %>
    {
        "code":"success",
        "data":<%=student %>
    }
```

20. Stored Procedure - Select All

```sql
CREATE OR REPLACE PROCEDURE  sp_students
(
     v_students    OUT   SYS_REFCURSOR
)
AS
BEGIN
     OPEN v_students FOR
     SELECT hakbun, irum, kor, eng, mat, edp, sum, avg, grade
     FROM Student
     ORDER BY sum DESC;
END;
/
```

1)Student.xml

```xml
    <parameterMap id="studentsMap" type="hashMap">
        <parameter property="results" javaType="ResultSet" jdbcType="CURSOR" mode="OUT" resultMap="selectMap"/>
    </parameterMap>

    <select id="selectAllSP" statementType="CALLABLE" parameterMap="studentsMap">
        { call sp_students(?) }
    </select>
```

2)StudentDAO.java

```java
    void selectAllSP(Map map);
```

3)StudentDAOImpl.java

```java
    @Override
    public void selectAllSP(Map map) {
        this.mybatis.selectList("Student.selectAllSP", map);
    }
```

4)index.jsp

```
    <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
    <%@ page import="java.util.List, java.util.Map, java.util.HashMap" %>
    <%@ page import="com.javasoft.libs.dao.*" %>
    <%@ page import="com.javasoft.libs.models.StudentVO" %>

    <%
        StudentDAO dao = new StudentDAOImpl();
        Map<String, Object> map = new HashMap<String, Object>();
        dao.selectAllSP(map);
        List<StudentVO> list = (List<StudentVO>)map.get("results");
        String str = "";
        for(StudentVO student : list){
            str += student.toString() + ",";
        }
        if(str.length() > 0) str = str.substring(0, str.length() - 1);
    %>
    {
        "code" : "success",
        "data" : [<%=str %>]
    }
```