

# JSP/Servlet

## <Bean>

Bean은 Java 클래스를 기반으로 한 개체(객체)를 나타내는 개념입니다. Bean 클래스는 일반적으로 속성(멤버 변수)과 해당 속성에 대한 getter 및 setter 메서드를 포함합니다. 이러한 속성과 메서드는 객체의 상태를 나타내고 조작하는 데 사용됩니다.

Bean은 주로 데이터를 표현하고 관리하기 위해 사용됩니다. 예를 들어, 사용자의 정보를 저장하기 위한 UserBean 클래스를 만들 수 있습니다. UserBean 클래스는 사용자의 이름, 나이, 이메일 등과 같은 속성을 가지며, 각 속성에 대한 getter 및 setter 메서드를 제공합니다. 이렇게 생성된 UserBean 객체는 사용자 정보를 저장하고 필요한 곳에서 해당 정보에 접근할 수 있습니다.

Bean은 일반적으로 JavaBeans라고도 불리며, 다음과 같은 특징을 가지고 있습니다:

**속성(멤버 변수):** Bean 클래스는 데이터를 저장하기 위한 속성을 가집니다. 속성은 클래스의 상태를 나타냅니다.

**Getter 및 Setter 메서드:** 각 속성에는 getter 및 setter 메서드가 있습니다. getter 메서드는 속성의 값을 반환하고, setter 메서드는 속성의 값을 설정합니다. 이를 통해 속성에 접근하고 조작할 수 있습니다.

**기본 생성자:** Bean 클래스는 기본 생성자를 가지고 있어야 합니다. 기본 생성자는 객체를 생성할 때 호출되며, 초기화 작업을 수행할 수 있습니다.

**직렬화 가능:** Bean 클래스는 직렬화(Serialization)가 가능해야 합니다. 직렬화는 객체를 바이트 스트림으로 변환하여 저장하거나 네트워크를 통해 전송할 수 있는 기능입니다.

Bean은 자바 기반의 다양한 프레임워크 및 라이브러리에서 사용되며, 데이터의 캡슐화와 유지보수성을 향상시키는 데 도움이 됩니다.

```
<jsp:useBean id="Class에서 정의한 호출하고 싶은 id" class="class 경로" />
```

```
<jsp:getProperty name="정의한 name" property="정의한 id" /> 형태로 Bean에서 자료를 불러들이고 <% %>의 Scriptlet으로 값을 불러들인다.
```

<Bean 활용 - 이와 같은 형태로 만들어주면 된다.>

Java 패키지와 클래스 생성

```
package com.exam;
```

```
public class ScoreBean {
    private String name;
    private int kor;
    private int eng;
    private int math;

    //getter,setter
    public String getName() {
        return name;
    }

    public void setName(String name) {
```

```

        this.name = name;
    }
    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 getMath() {
        return math;
    }
    public void setMath(int math) {
        this.math = math;
    }
}

// 총점
public int getTotal() {
    return kor+eng+math;
}

// 평균
public float getAvg() {
    return getTotal()/3.0f;
}

// 학점
public String getGrade() {
    String grade="";
    switch ((int)(getAvg()/10)) {
        case 10:
        case 9: grade="A"; break;
        case 8: grade="B"; break;
        case 7 : grade="C"; break;
        default: grade="A"; break;
    }
    return grade;
}

}

```

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>

```

```

<meta charset="UTF-8">
<title>useBean</title>
<%
request.setCharacterEncoding("utf-8");
%>
<jsp:useBean id="sb" class="com.exam.ScoreBean"/>
<jsp:setProperty property="*" name="sb"/>
</head>
<body>
    이름 : <jsp:getProperty property="name" name="sb"/><br/>
    국어 : <jsp:getProperty property="kor" name="sb"/><br/>
    영어 : <jsp:getProperty property="eng" name="sb"/><br/>
    수학 : <jsp:getProperty property="math" name="sb"/><br/>
    <hr/>
    이름 : <%=sb.getName() %><br/>
    국어 : <%=sb.getKor() %><br/>
    영어 : <%=sb.getEng() %><br/>
    수학 : <%=sb.getMath() %><br/>
    총점 : <%=sb.getTotal() %><br/>
    평균 : <%=sb.getAvg() %><br/>
    학점 : <%=sb.getGrade() %>
</body>
</html>

```

## <DB와 JSP 연동>

(insert.jsp 생성)

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert</title>
</head>
<body>
<p><a href="list.jsp">전체보기</a></p>
<p></p>
<form action="insertPro.jsp" method="post">
<table border="1">
<tr>
<td colspan="2">주소록 등록하기</td>
</tr>
<tr>
<td>이름</td>
<td><input type="text" name="name"></td>
</tr>
<tr>
<td>우편번호</td>
<td><input type="text" name="zipcode" id="zipcode" size="5"><input type="button"
value="검색"></td>
</tr>

```

```

<td>주소</td>
<td><input type="text" name="addr" id="addr" size="50"></td>
</tr>
<tr>
<td>전화번호</td>
<td><input type="text" name="tel"></td>
</tr>
<tr>
<td colspan="2"><input type="submit" value="등록">
<input type="button" value="취소">
</td>
</tr>
</table>
</form>
</body>
</html>

```

<자바 패키지 및 클래스 생성>

```
package com.address;
```

```

public class Address {
    private int num;
    private String name;
    private String zipcode;
    private String addr;
    private String tel;

    //getter, setter
    public int getNum() {
        return num;
    }
    public void setNum(int num) {
        this.num = num;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getZipcode() {
        return zipcode;
    }
    public void setZipcode(String zipcode) {
        this.zipcode = zipcode;
    }
    public String getAddr() {
        return addr;
    }
    public void setAddr(String addr) {
        this.addr = addr;
    }
}

```

```

    }
    public String getTel() {
        return tel;
    }
    public void setTel(String tel) {
        this.tel = tel;
    }
}
}

```

<DB 드라이버 이용해서 불러들이기 및 DB와 자료 연동>

<드라이버에 JAR 파일 불러와서 lib에 담아야 한다.>

oracle DB가 있는 폴더에 가서 확인해보면 됨.

(AddressDAO.java 생성)

```

package com.address;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;

public class AddressDAO {
    String url;
    String user;
    String pwd;

    // DB Connection(DB 연결)
    public AddressDAO() {
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver")
            ; url = "jdbc:oracle:thin:@localhost:1521:xe";
            user = "scott";
            pwd = "tiger";
        } catch (ClassNotFoundException e) { e.printStackTrace(); }
    }

    // 추가 Insert
    // Connection Statement PreparedStatement(문자열처리)
    // ResultSet의 단계를 활용하여 SQL Query를 처리함.
    public void insert(Address ad) {
        Connection con = null;
        PreparedStatement ps = null;
    }
}

```

```

    try {
        con = DriverManager.getConnection(url, user, pwd);
        String sql = "insert into address values(address_seq.nextval,?,?,?,?)"; ps
        = con.prepareStatement(sql);
        ps.setString(1, ad.getName());
        ps.setString(2, ad.getZipcode());
        ps.setString(3, ad.getAddr());
        ps.setString(4, ad.getTel());
        ps.executeUpdate();

    } catch (SQLException e) {
        e.printStackTrace();
    }

}

// 전체 보기
public ArrayList<Address> list() {
    Connection con = null;
    Statement st = null;
    ResultSet rs = null;
    ArrayList<Address> arr = new ArrayList<>();

    try {
        con = DriverManager.getConnection(url, user, pwd);
        st = con.createStatement();
        String sql = "select * from address"; rs
        = st.executeQuery(sql);
        while (rs.next()) {
            Address ad = new Address();
            ad.setAddr(rs.getString("addr"));
            ad.setName(rs.getString("name"));
            ad.setNum(rs.getInt("num"));
            ad.setTel(rs.getString("tel"));
            ad.setZipcode(rs.getString("zipcode"));
            arr.add(ad);
        }

    } catch (SQLException e) {
        e.printStackTrace();
    }

    return arr;
}

}

```

<DB 연동준비>  
(insertPro.jsp 생성)

```

<%@page import="com.address.AddressDAO"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("utf-8");
%>

<jsp:useBean id="ad" class="com.address.Address"/>
<jsp:setProperty property="*" name="ad"/>
<%
AddressDAO dao = new AddressDAO();
dao.insert(ad);
%>

```

(list.jsp)

```

<%@page import="com.address.Address"%>
<%@page import="java.util.ArrayList"%>
<%@page import="com.address.AddressDAO"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>List</title>
</head>
<body>
    <h3>전체보기</h3>
    <%
AddressDAO dao = new AddressDAO();
ArrayList<Address> arr = dao.list();
%>
    <table border="1">
    <tr>
    <th>번호</th>
    <th>이름</th>
    <th>주소</th>
    <th>전화번호</th>
    </tr>
    <% for(Address ad : arr){
%>
    <tr> <!-- tr, td는 태그이므로 스크립틀릿이 동작 X(자바문법이 아니기 때문)
Scriptlet은 여달아주기!-->
    <td><%=ad.getNum()%></td>
    <td><%=ad.getName() %></td>
    <td><%=ad.getAddr() %></td>
    <td><%=ad.getTel() %></td>
    </tr>
    <%
}

```

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