

제 10 장 자바 빈즈



❖ 자바 빈즈(Java Beans)

- 자바 프로그램에서 특정한 작업인 비즈니스 로직을 독립적으로 수행하는 하나의 프로그램 단위
- 큰 프로그램에서 독립적으로 수행되는 하나의 작은 프로그램 부품을 컴포넌트(Component)라 함
- 비즈니스 로직 부분을 담당하는 자바 프로그램 단위
 - 자바 빈즈는 프로그램의 **비즈니스 로직** 부분과 **프리젠테이션** 부분을 분리

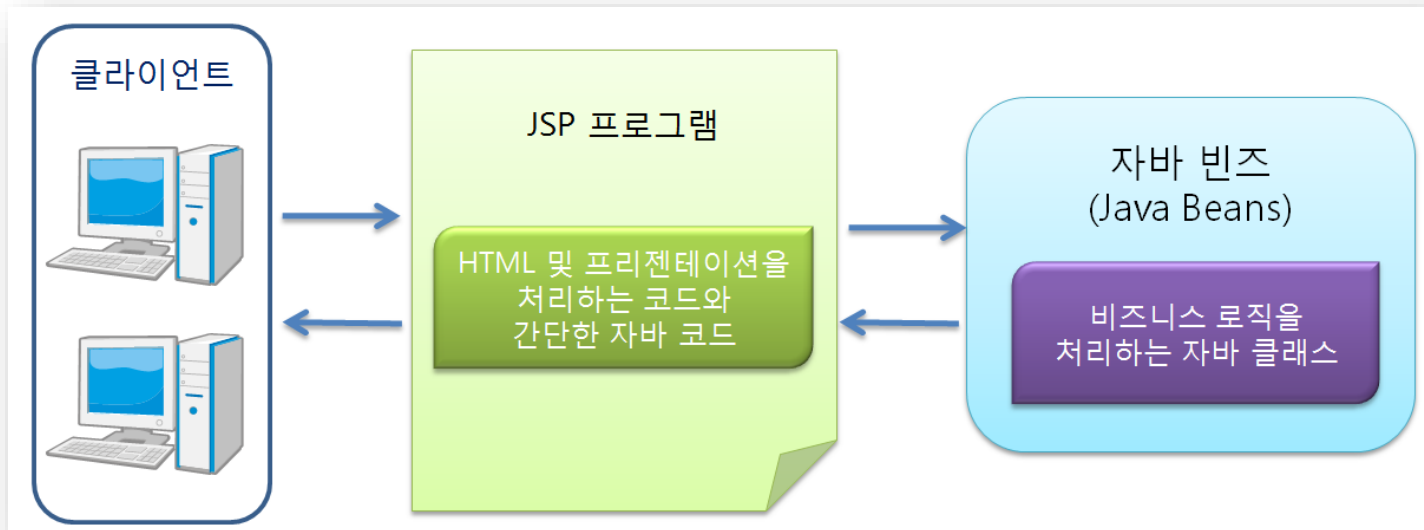
❖ 장점

- 자바 빈즈를 이용하면 JSP 페이지가 복잡한 자바 코드로 구성되는 것을 피하고,
- JSP 페이지에는 HTML 코드와 쉽고 간단한 자바 코드만을 구성
- 한번 작성된 자바 빈즈를 여러 응용 프로그램에서 재사용이 가능
- 프로그램의 개발 기간도 단축



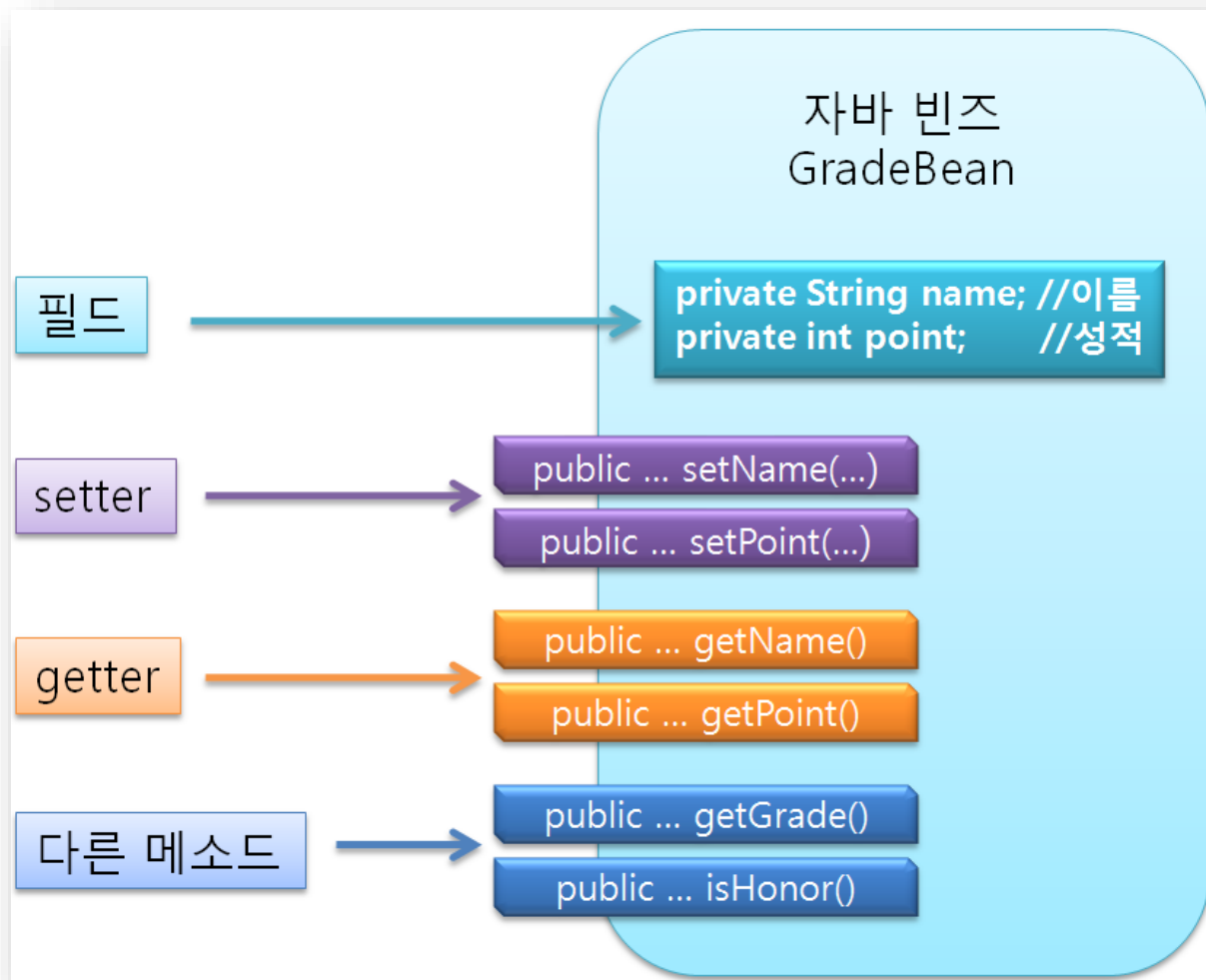
❖ 자바 클래스인 자바 빈즈

- 비즈니스 로직을 처리하는 자바 클래스인 자바 빈즈





❖ 필드와 메소드





❖ 액션 태그(Action tag)

액션	내용
<code><jsp:useBean id="login" ... /></code>	JSP와 연관시켜 자바 빈즈를 생성
<code><jsp:setProperty name="login" ... /></code>	생성된 자바 빈즈의 객체를 이용해 setter 에 속성 값을 전달
<code><jsp:getProperty name="login" ... /></code>	생성된 자바 빈즈의 객체를 이용해 getter 로 속성 값을 반환

자바 빈즈 액션 태그의 속성



액션	속성	값 유형	설명
<jsp:useBean ... />	id	문자열	JSP 페이지내에서 자바 빈즈의 참조 변수를 저장하는 변수 이름 을 지정
	class	문자열	생성할 자바 빈즈의 클래스 이름
	scope	page request session application	자바 빈즈의 유효 범위를 나타내며, 지정하지 않으면 기본 값은 page
<jsp:setProperty ... />	name	문자열	<jsp:useBean>에서 지정한 id로 지정
	property	문자열	자바 빈즈의 setter()의 이름 setName()에서 set을 제거한 name으로 지정하며, 값이 "*"이면 피라미터의 모든 값을 지정하는 의미
	param	문자열	속성 property와 함께 쓰이며, 지정된 파라미터로 전달받은 파라미터의 이름을 지정
	value	문자열	속성 property와 함께 쓰이며, 자바 빈즈의 setter()의 setName(value)에 지정하는 인자(매개변수) 값인 value를 지정
<jsp:getProperty ... />	name	문자열	<jsp:useBean>에서 지정한 id로 지정
	property	문자열	자바 빈즈의 getter() 이름 getName()에서 name으로 지정



❖ 태그 <jsp:useBean ... />에서 속성 scope

- 자바 빈즈의 유효 범위
- page, request, session, application 중 하나의 값
- 지정하지 않으면 기본 값은 page

액션	내용
page	자바 빈즈가 현재의 JSP 페이지 내에서만 사용 가능하며, 기본 값이므로 특별히 지정하지 않으면 이 옵션이 적용, 가장 좁은 범위 scope 값
request	JSP 페이지는 request 객체가 영향을 미치는 모든 JSP 페이지까지 자바 빈즈 이용 가능
session	세션이 유효한 페이지까지 자바 빈즈 이용 가능
application	응용 프로그램의 모든 페이지에서 자바 빈즈 객체 사용 가능하며, 이 값은 가장 넓은 범위 scope 값



❖ 자바 빈즈를 이용하려는 선언 문장

- JSP 프로그램에서 자바 빈즈를 이용하려는 선언 문장
- 적어도 속성 **id**와 **class**는 필요
- **id**는 객체 참조를 저장하는 **변수 이름**
- **class**는 객체 참조의 **클래스 이름**

```
<jsp:useBean id="test" class="ClassName" />
<% ClassName test = new ClassName(); %>
```

그럼, 태그 <jsp:useBean ... >와 같은 의미의 자바 코드



❖ setter를 호출

- 이미 선언된 자바 빈즈에서 속성 **property**로 지정된 이름을 갖는 메소드 **setter**를 호출하는 문장
- 적어도 속성 **name**과 **property**는 있어야 하며
 - 속성 **name**은 반드시 태그 <jsp:useBean id="test"... />에서 이미 지정한 id 값과 일치

```
<jsp:useBean id="test" class="ClassName" scope="application" />↵  
↵  
<jsp:setProperty name="test" property="name" value="김성민" />  
↵  
그림. 태그 <jsp:setProperty ... >의 이용 ↵
```

태그 <jsp:setProperty ... />

- 속성 property는 호출할 setter 이름이 setName()이라면
 - property="name"으로 지정
- 속성 value는 메소드 setter를 호출할 때의 인자 값

```
<jsp:setProperty name="test" property="name" value="김성민" />
<% test.setName("김성민"); %>
```

그림. 태그 <jsp:setProperty ... >와 같은 의미의 자바 코드

- 속성 property는 다음 4 개 중에 하나의 형태로 이용
 - 속성 property="*"이면 파라미터의 모든 값으로, 모든 setter를 호출하는 문장을 의미

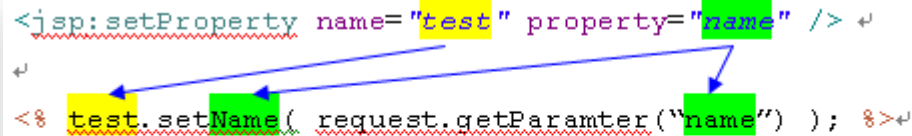
```
<jsp:setProperty name="test" property="*" />
<jsp:setProperty name="test" property="name" />
<jsp:setProperty name="test" property="name" param="username" />
<jsp:setProperty name="test" property="name" value="김성민" />
```

그림. 태그 <jsp:setProperty ... >에서 속성 property 이용 방법

태그 <jsp:setProperty ... />

- 속성 name과 property가 있으면
 - property로 지정된 같은 이름으로 파라미터 인자를 이용하는 문장

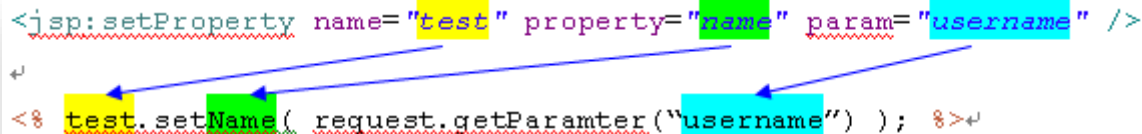
```
<jsp:setProperty name="test" property="name" />
<% test.setName( request.getParameter("name") ); %>
```



그럼. 태그 <jsp:setProperty ...>에서 속성 value가 없는 태그와 같은 의미의 자바 코드

- 속성 name과 property, param이 모두 있으면
 - 지정된 param으로 파라미터 인자를 이용하는 문장

```
<jsp:setProperty name="test" property="name" param="username" />
<% test.setName( request.getParameter("username") ); %>
```

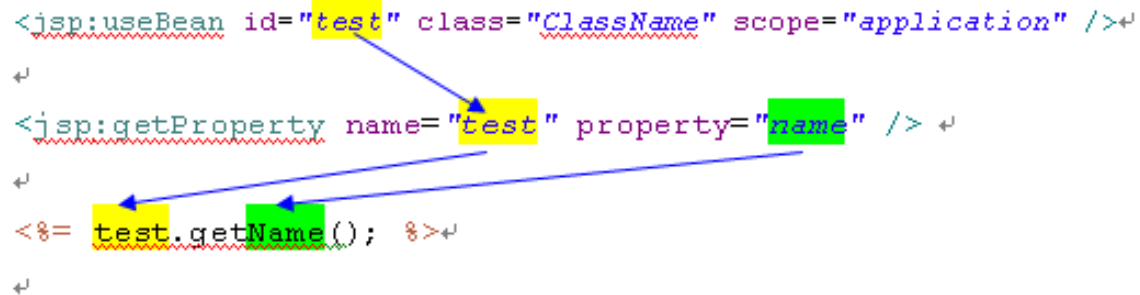


그럼. 태그 <jsp:setProperty ... param=...>와 같은 의미의 자바 코드

태그 <jsp:getProperty ... />

- 2개의 속성 name과 property 모두 필요
- 속성 name
 - 반드시 태그 <jsp:useBean id="test"... />에서 지정한 id 값과 일치
- 속성 property
 - 호출할 getter 이름이 getName()이라면 property="name"으로 지정

```
<jsp:useBean id="test" class="ClassName" scope="application" />
<jsp:getProperty name="test" property="name" />
<%= test.getName(); %>
```



그럼. 태그 <jsp: getProperty ... >와 같은 의미의 자바 코드

자바 빈즈■ 이용한 자료 값의 저장과 처리

❖ 학생의 이름과 성적 정보

JSP 프로그램

```
<jsp:useBean id="score" class="javabeen.ScoreBean" scope="page" />
```

```
<jsp:setProperty name="score" property="name" value="김성민"/>  
<jsp:setProperty name="score" property="point" value="85"/>
```

```
이름 : <jsp:getProperty name="score" property="name" /> <BR>  
성적 : <jsp:getProperty name="score" property="point" /> <BR>
```

자바 빈즈 생성
javabeen.ScoreBean

자바 빈즈
javabeen.ScoreBean

```
public ... setName(...)
```

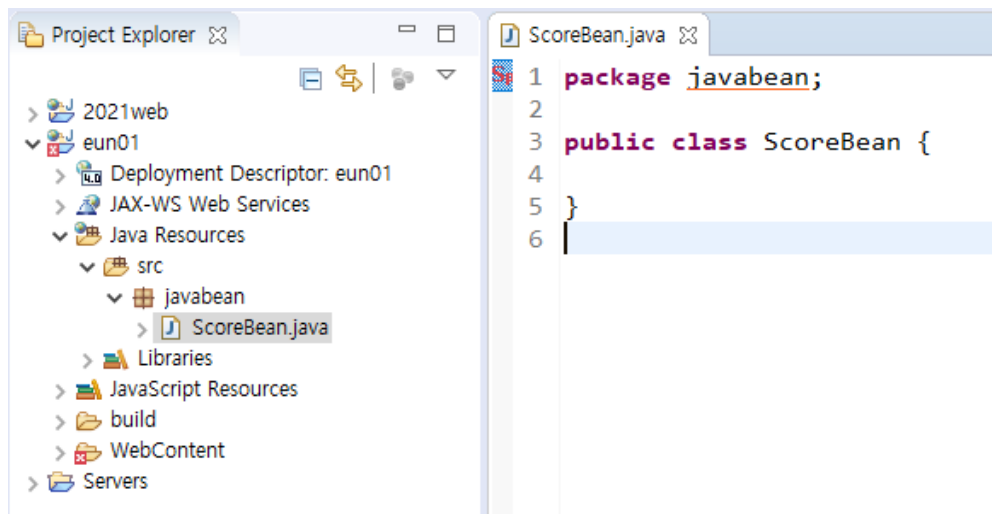
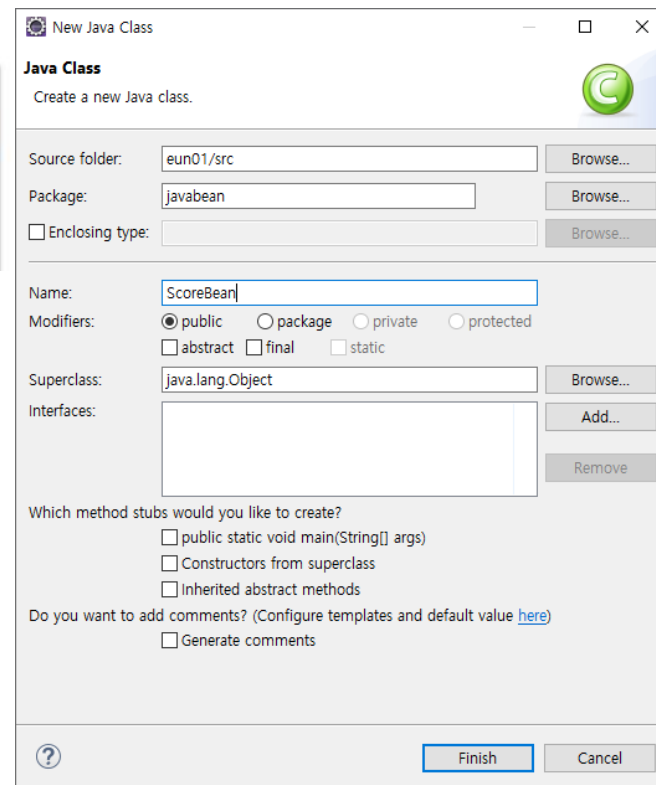
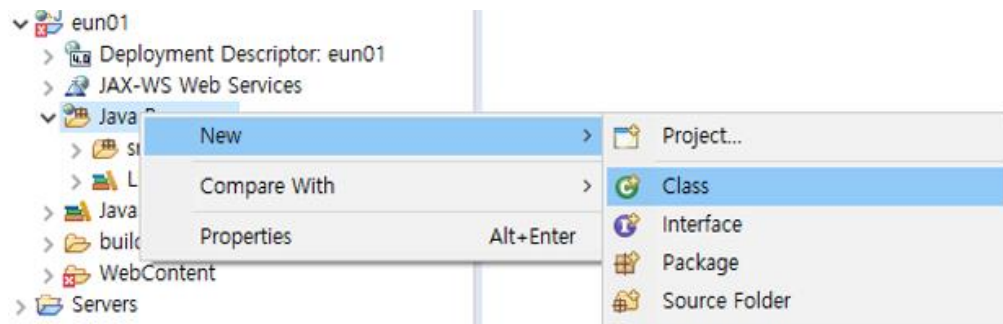
```
public ... setPoint(...)
```

```
private String name; //이름  
private int point;   //성적
```

```
public ... getName()
```

```
public ... getPoint()
```

❖ 메뉴 [Java Resources]–[New]–[Class]



필드와 *getter, setter*

❖ 필드

```
public class ScoreBean {  
    private String name; //이름  
    private int point;    //성적  
}
```

```
package javabean;  
  
public class ScoreBean {  
    private String name;    //이름  
    private int point;      //성적  
}
```

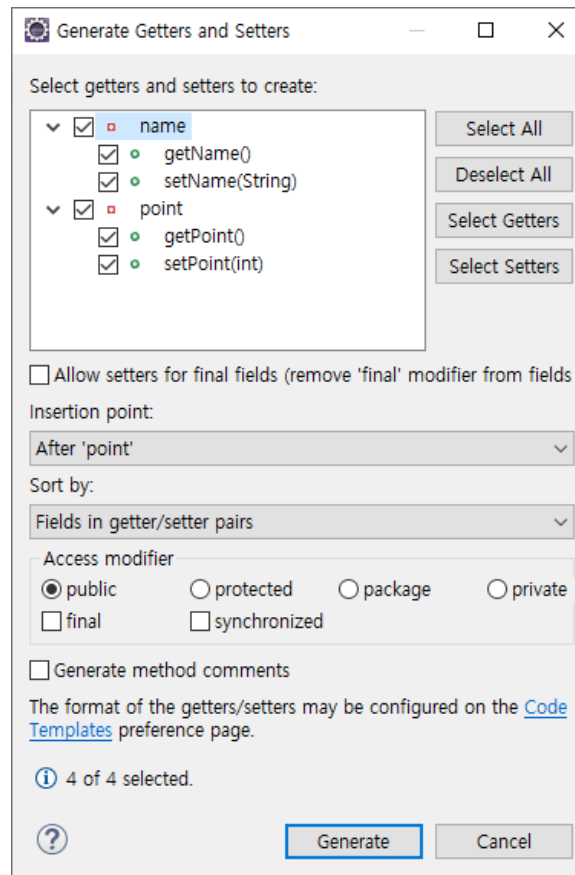
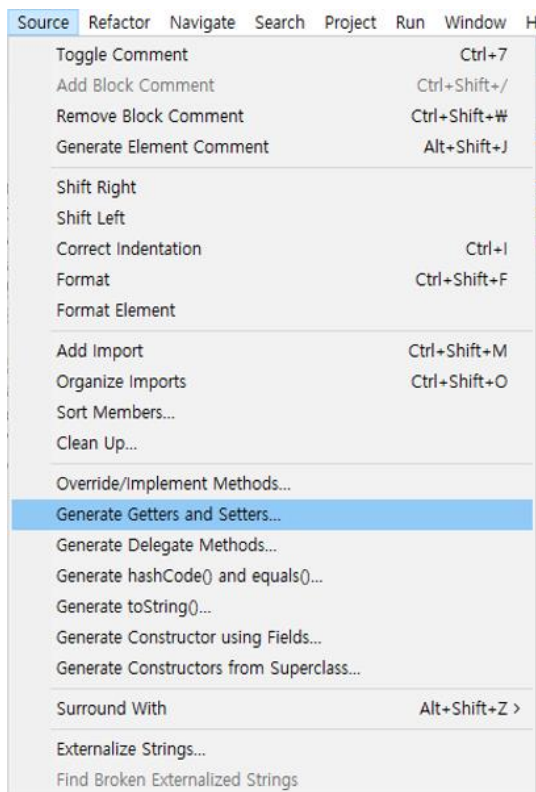
❖ 자바 빈즈의 정보를 저장, 조회하는 getter와 setter

- getter는 메소드 이름 getXxx()으로 만들며 setter는 setXxx(type xxx)으로 작성
 - 메소드 이름 xxx는 필드의 이름
 - 메소드 이름에서 get과 set 다음에 나오는 첫 글자는 대문자로 작성하는 것이 관례
 - getter와 setter는 소속 변수 중에서 저장과 조회가 필요한 필드에 대하여 생성

```
public String getName() {  
    return name;  
}  
public void setName(String name) {  
    this.name = name;  
}
```

getter와 setter의 자동 생성

- ❖ 이클립스에서 메뉴 [source]/[Generate Getter and Setters ...]를 이용하여 일괄적으로 생성





❖ ScoreBean.java

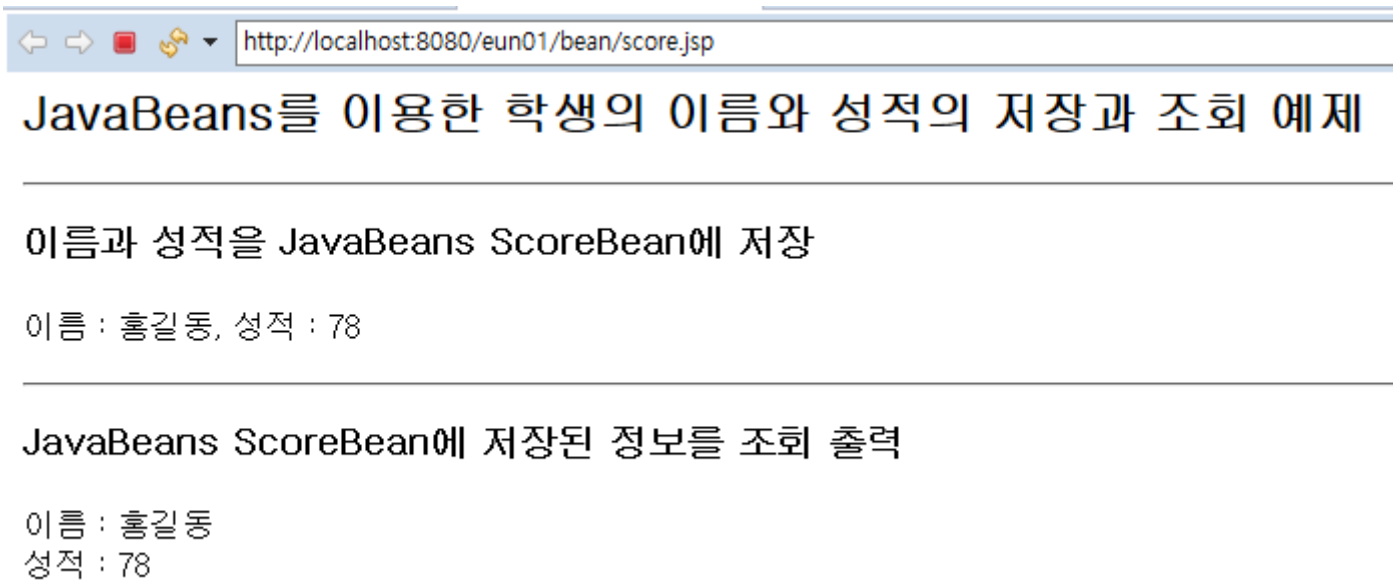
```
1 package javabean;
2
3 public class ScoreBean {
4     private String name;    //이름
5     private int point;      //성적
6
7     public String getName() {
8         return name;
9     }
10    public void setName(String name) {
11        this.name = name;
12    }
13    public int getPoint() {
14        return point;
15    }
16    public void setPoint(int point) {
17        this.point = point;
18    }
19
20 }
```



❖ score.jsp

```
ScoreBean.java score.jsp
1 <%@ page language="java" contentType="text/html; charset=EUC-KR" pageEncoding="EUC-KR"%>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
5 <title>JSP 예제 : score.jsp</title>
6 </head>
7 <body>
8
9 <h2>JavaBeans를 이용한 학생의 이름과 성적의 저장과 조회 예제</h2>
10 <jsp:useBean id="score" class="javabeen.ScoreBean" scope="page" />
11
12 <HR>
13 <h3>이름과 성적을 JavaBeans ScoreBean에 저장</h3><p>
14 이름 : <%= "홍길동"%>,
15 성적 : <%= "78" %><p>
16 <jsp:setProperty name="score" property="name" value="홍길동"/>
17 <jsp:setProperty name="score" property="point" value="78"/>
18
19 <HR>
20 <h3>JavaBeans ScoreBean에 저장된 정보를 조회 출력</h3><p>
21 이름 : <jsp:getProperty name="score" property="name" /><BR>
22 성적 : <jsp:getProperty name="score" property="point" /><BR>
23
24 </body>
25 </html>
```

❖ score.jsp 실행





❖ 프로그램 구성

- grade.html
 - 이름과 점수를 입력 받는 폼을 구성하는 프로그램
- grade.jsp
 - grade.html에서 입력 받은 폼 정보를 다시 자바 빈즈에 전달하는 프로그램
- javabean.GradeBean.java
 - 이름과 점수를 저장하여 그 점수에 해당하는 학점을 반환하는 자바 빈즈 프로그램

자바 빈즈를 이용한 폼 입력 처리 프로그램 구성

```
<h2> 이름과 점수를 입력하세요 </h2>
<form method=post action=grade.jsp>
이름 : <input type=text name=na size=16><br>
점수 : <input type=text name=po size=3><p>
<input type="submit" value="입력완료" >
<input type="reset" value="다시쓰기" >
</form>
```

이름과 점수를 입력하세요

이름 :

점수 :

```
<jsp:useBean id="score" class="javabean.GradeBean" scope="page" />
```

```
<jsp:setProperty name="score" property="name" param="na" />
<jsp:setProperty name="score" property="point" param="po" />
```

```
이름 : <jsp:getProperty name="score" property="name" /><BR>
성적 : <jsp:getProperty name="score" property="point" /><BR>
등급 : <jsp:getProperty name="score" property="grade" /><BR>
```



❖ grade.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
5 <title>예제 grade</title>
6 </head>
7 <body>
8     <h2> 이름과 점수를 입력하세요 </h2>
9     <form method=post action=grade.jsp>
10     이름 : <input type=text name=name size=16><br>
11     점수 : <input type=text name=point size=3><p>
12     <input type="submit" value="입력완료" >
13     <input type="reset" value="다시쓰기" >
14     </form>
15 </body>
16 </html>
```



❖ GradeBean.java

```
3 public class GradeBean {
4
5     private String name;
6     private int point;
7
8     public String getName() {
9         return name;
10    }
11    public void setName(String name) {
12        this.name = name;
13    }
14    public int getPoint() {
15        return point;
16    }
17    public void setPoint(int point) {
18        this.point = point;
19    }
20
21    public String getGrade() {
22        String grade = "";
23        if (point >= 90)
24            grade = "A";
25        else if (point >= 80)
26            grade = "B";
27        else if (point >= 70)
28            grade = "C";
29        else if (point >= 60)
30            grade = "D";
31        else
32            grade = "F";
33        return grade;
34    }
35 }
```

❖ grade.jsp

```
1 <%@ page language="java" contentType="text/html; charset=EUC-KR" pageEncoding="EUC-KR"%>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
5 <title>JSP 예제 : grade.jsp</title>
6 </head>
7 <body>
8
9     <h2>JavaBeans를 이용한 학생의 점수에 따른 성적 처리 예제</h2>
10    <% request.setCharacterEncoding("euc-kr"); %>
11    <jsp:useBean id="score" class="javabeen.GradeBean" scope="page" />
12
13    <HR>
14    <h3>폼에서 전달받은 이름과 성적을 JavaBeans GradeBean에 저장</h3><p>
15    이름 : <%= request.getParameter("name") %>,
16    성적 : <%= request.getParameter("point") %><p>
17    <jsp:setProperty name="score" property="name" param="name" />
18    <jsp:setProperty name="score" property="point" param="point" />
19
20    <HR>
21    <h3>JavaBeans GradeBean에 저장된 정보를 조회 출력</h3><p>
22    이름 : <jsp:getProperty name="score" property="name" /><BR>
23    성적 : <jsp:getProperty name="score" property="point" /><BR>
24    등급 : <jsp:getProperty name="score" property="grade" /><BR>
25
26 </body>
27 </html>
```




← → ⏏ ↺ http://localhost:8080/eun01/bean/grade.html

이름과 점수를 입력하세요

이름 :

점수 :

입력완료

다시쓰기

← → ⏏ ↺ http://localhost:8080/eun01/bean/grade.jsp

JavaBeans를 이용한 학생의 점수에 따른 성적 처리 예제

폼에서 전달받은 이름과 성적을 JavaBeans GradeBean에 저장

이름 : 나관리, 성적 : 78

JavaBeans GradeBean에 저장된 정보를 조회 출력

이름 : 나관리

성적 : 78

등급 : C



- ❖ [과제]강의 동영상 자료에 있는 2가지 유형의 문제를 모두 프로그램 한 후 소스와 출력 결과를 편집하여 제출하세요.

-
- ❖ 제출방법 : ppt파일에 소스와 출력 결과 편집 후 제출
 - ❖ 제출마감일 : 2021년 05월17일 월요일 23시 55분까지