

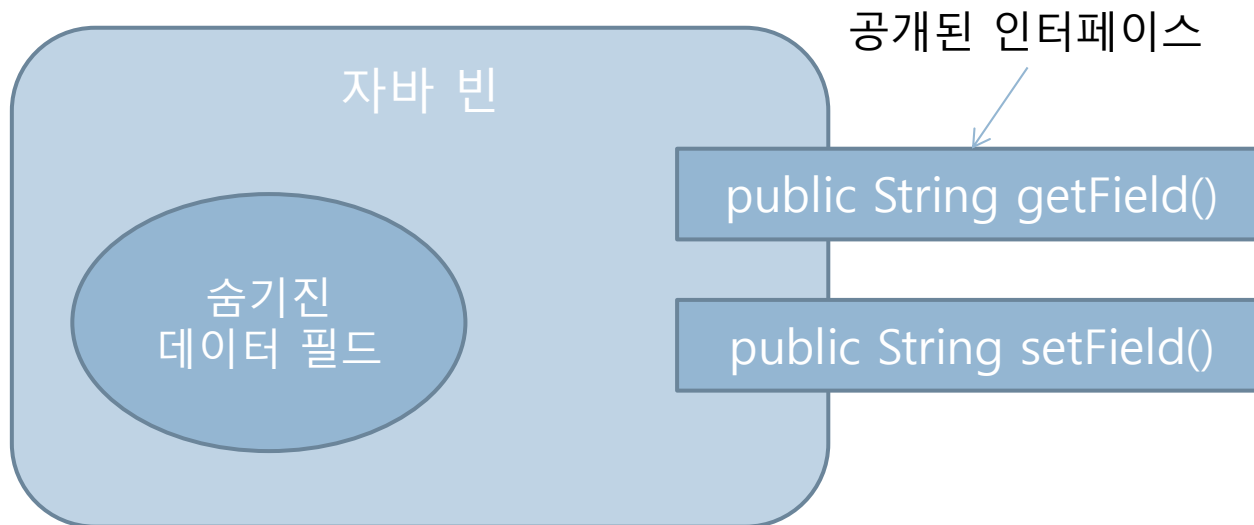
6. JSP 자바빈과 액션 태그

1. 자바빈의 개요 및 작성
2. 자바빈과 연동하는 액션 태그
3. 자바빈 사용 회원 관리 시스템

1. 자바빈의 개요 및 작성

□ 자바빈

- ▣ 정보의 덩어리로 정보의 저장소라고 정의
 - 프로그램에서 사용되는 정보가 여러 개인 경우 이를 여러 개의 변수에 저장하는 대신 자바빈 을 사용하여 객체 단위로 처리 가능.
 - 자바빈 클래스를 작성하여 클래스 단위 처리
- ▣ 자바의 데이터 은닉 개념 사용



1. 자바빈의 개요 및 작성

□ 자바빈 사용예:

▣ 일반적인 jsp 코드

```
<%  
    member.setName(request.getParameter("name"));  
    member.setId(request.getParameter("id"));  
    member.setNickname(request.getParameter("nickname"));  
    member.setPwd(request.getParameter("pwd"));  
    member.setEmail(request.getParameter("Email"));  
    member.setPhone(request.getParameter("phone"));  
%>
```

▣ 액션 태그 사용

```
<jsp:setProperty property="*" name="member">
```

1. 자바빈의 개요 및 작성

□ 자바빈(JavaBean) 클래스 작성

▣ 자바빈 작성하기

- 자바빈은 자바로 작성되어진 컴포넌트
- 클래스들로 이루어진 복합적인 구조
- 자바 언어의 프로그램 작성 규칙과 문법을 따름

▣ 자바의 클래스를 만드는 것과 같은 규칙

▣ 작성 순서

1. package 패키지명; //없으면 생략가능
2. import 패키지명을 포함한 클래스의 풀네임; //없으면 생략가능
3. class 클래스명{ //필수정의, 생략 불가능
 }

□ 자바빈(JavaBean) 클래스 작성

- 자바 클래스 선언 형식
 - 접근제어자 [키워드] class 클래스명 {}
- 선언 예
 - public class DbDataLogin {}
- 자바빈 클래스 선언-접근 제어자 public
- 멤버 변수-접근 제어자 private

(예)

```
package bean.logon;  
public class DbDataLogin{  
    private String id;  
    public void setId(String id){  
        this.id = id.trim();  
    }  
    public String getId(){  
        return id;  
    }  
}
```

□ Setter/Getter 메소드 작성

- ▣ 프로퍼티(property): 값을 저장하기 위한 멤버 필드
 - 접근 제어자를 private로 선언해서 작성
 - 보안이 강한 private 프로퍼티에 접근해 정보를 얻어내기 위해 getter/setter사용
- ▣ setter – 프로퍼티에 값 저장
 - setXxx() 메소드 사용
 - Xxx는 프로퍼티명으로 첫 글자는 대문자
 - 프로퍼티명이 id인 경우 setter는 setId()

```
public void setId(String id){  
    this.id = id.trim();  
}
```

□ Setter/Getter 메소드 작성

- ▣ getter – 프로퍼티에 저장된 값 사용
 - getXxx() 메소드 사용
 - Xxx는 프로퍼티명으로 첫 글자는 대문자
 - 프로퍼티명이 id인 경우 getter는 getId()

```
public String getId(){  
    return id;  
}
```

□ 자바빈 작성하기

□ 가상 환경 작성 위치

- [프로젝트명]-[Java Resources]-[src] 폴더

□ 실제 환경에서 위치

- [웹 애플리케이션 폴더\WEB-INF\classes] 폴더

□ 작성 예

```
package ch06.bean;
public class TestBean{
    private String id;
    public String getId(){
        return id;
    }
    public void setId(String id){
        this.id = id;
    }
}
```

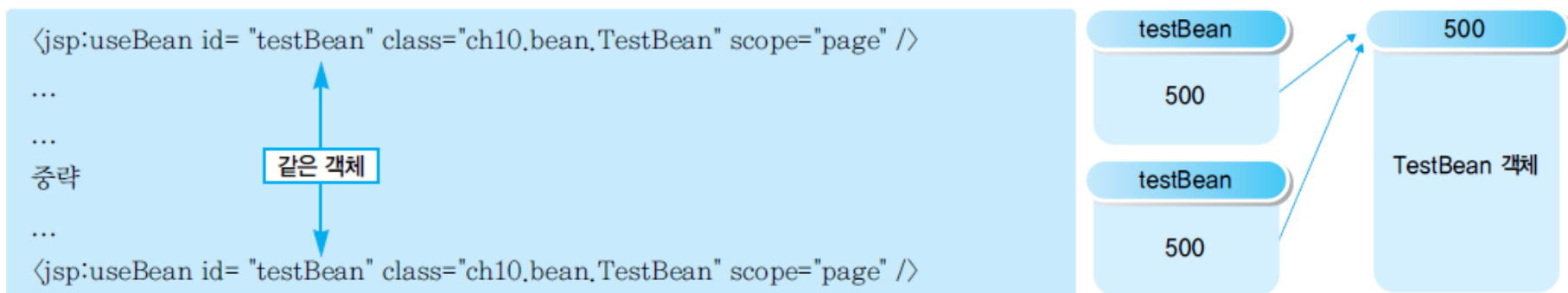

2. 자바빈과 연동하는 액션 태그

□ 자바빈 객체 생성-useBean 액션 태그(<jsp:useBean>)

▣ 사용하는 방법

```
<jsp:useBean id="빈이름" class = "자바빈 클래스이름" scope = "범위" />  
<jsp:useBean id="testBean" class="ch08.bean.TestBean" scope="page" />
```

- id 속성 값에 지정한 객체의 레퍼런스명이 이미 존재하는 경우, 기존에 생성된 객체를 그대로 사용



▲ 같은 testBean 객체를 사용

□ 자바빈의 프로퍼티 값 저장-setProperty 액션 태그(<jsp:setProperty>)

▣ 사용 방법

```
<jsp:setProperty name= "빈 이름" property="프로퍼티 이름"  
                 value="프로퍼티에 저장할 값" />
```

▣ 사용 예

```
<jsp:useBean id= "testBean" class="ch08.bean.TestBean" scope="page" >  
  <jsp:setProperty name="testBean" property="id"/>  
</jsp:useBean>
```

□ 자바빈의 프로퍼티 값 저장-setProperty 액션 태그(<jsp:setProperty>)

▣ 사용 예 : 많은 프로퍼티 한 번에 지정

■ 사용자 입력 폼

```
<input type="text" name="id" size="10" maxlength="10">
```

■ 자바빈 클래스

```
public void setId(String id) {this.id = id;}
```

■ 자바빈을 사용하는 JSP 페이지

```
<jsp:useBean id="inDb" scope="page" class="bean.logon.DbDataLogin">  
  <jsp:setProperty name="inDb" property="*/>  
</jsp:useBean>
```

□ 자바빈의 프로퍼티 값 저장-setProperty 액션 태그(<jsp:setProperty>)

- ▣ 사용 예 : 많은 프로퍼티 한 번에 지정
 - 자바빈을 사용하는 JSP 페이지

```
<jsp:useBean id="inDb" scope="page" class="bean.logon.DbDataLogin">  
  <jsp:setProperty name="inDb" property="*/>  
</jsp:useBean>
```

□ 자바빈의 프로퍼티 값 얻기-getProperty 액션 태그(<jsp:getProperty>)

▣ 사용 방법

```
<jsp:getProperty name= "빈 이름" property="프로퍼티 이름"/>
```

▣ 사용 예

```
<jsp:useBean id= "testBean" class="ch08.bean.TestBean" scope="page" />  
<jsp:getProperty name="testBean" property="id"/>
```

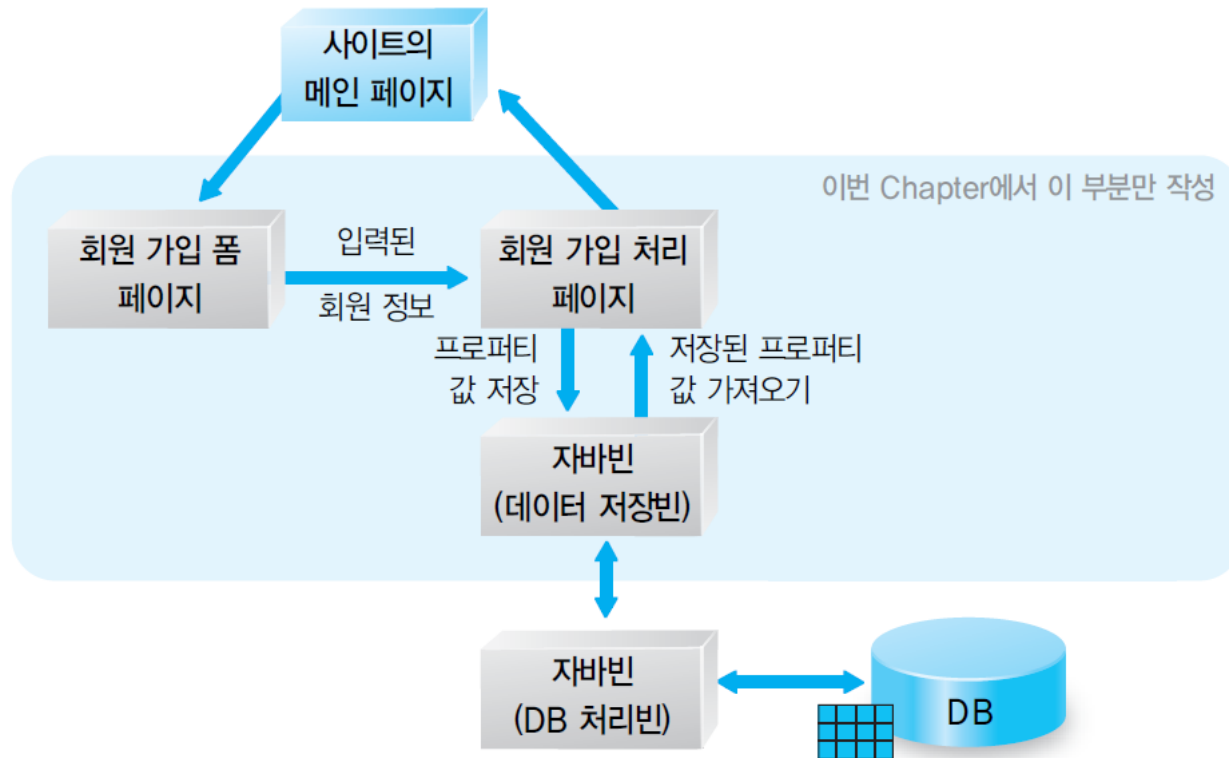
3. 자바빈을 사용한 회원 관리 시스템의 회원 가입 부분 작성

□ 회원 가입 시스템 개요

- ▣ 회원 관리 시스템은 크게 회원 가입, 회원 인증, 회원 정보 수정, 회원 탈퇴로 이루어짐
- ▣ 회원 가입 부분 작성
 - 회원 정보 폼으로부터 회원 정보를 입력 받아 필요한 정보를 저장

□ 회원 가입 시스템 개요

▣ 회원 가입 부분의 구조



▲ 회원 가입 부분의 구조

□ 회원 가입 정보 저장을 위한 자바빈 작성하기

- ▣ 4개의 프로퍼티를 가진 데이터 저장빈

```
private String idt; //아이디  
private String passwd; //비밀번호  
private String name; //이름  
private Timestamp reg_date; //가입일
```

□ 회원 가입 폼 및 가입된 정보 확인 페이지 작성하기

- ▣ 회원가입 폼: registerForm.jsp 페이지
- ▣ 정보 확인 : registerPro.jsp 페이지
- ▣ 스타일시트 : style.css