

# 12: JSP 고급 개발 방법



## 학습 목표

- 디자인 패턴과 소프트웨어 개발 관계를 이해한다.
- MVC 패턴을 이해하고 적용 방법을 익힌다.

## 디자인 패턴

디자인 패턴은 기존 환경에서 반복적으로 일어나는 문제를 설명하고, 그 문제 해법의 핵심을 설명하는 것이다. 이렇게 하면 같은 방법을 두 번 반복하지 않고, 이 해법을 백만 번 이상 재사용할 수 있다.

-크리스토퍼알렉산더

객체 지향 소프트웨어 개발에 필요한 기법을 정리  
소프트웨어 유지보수에 유리

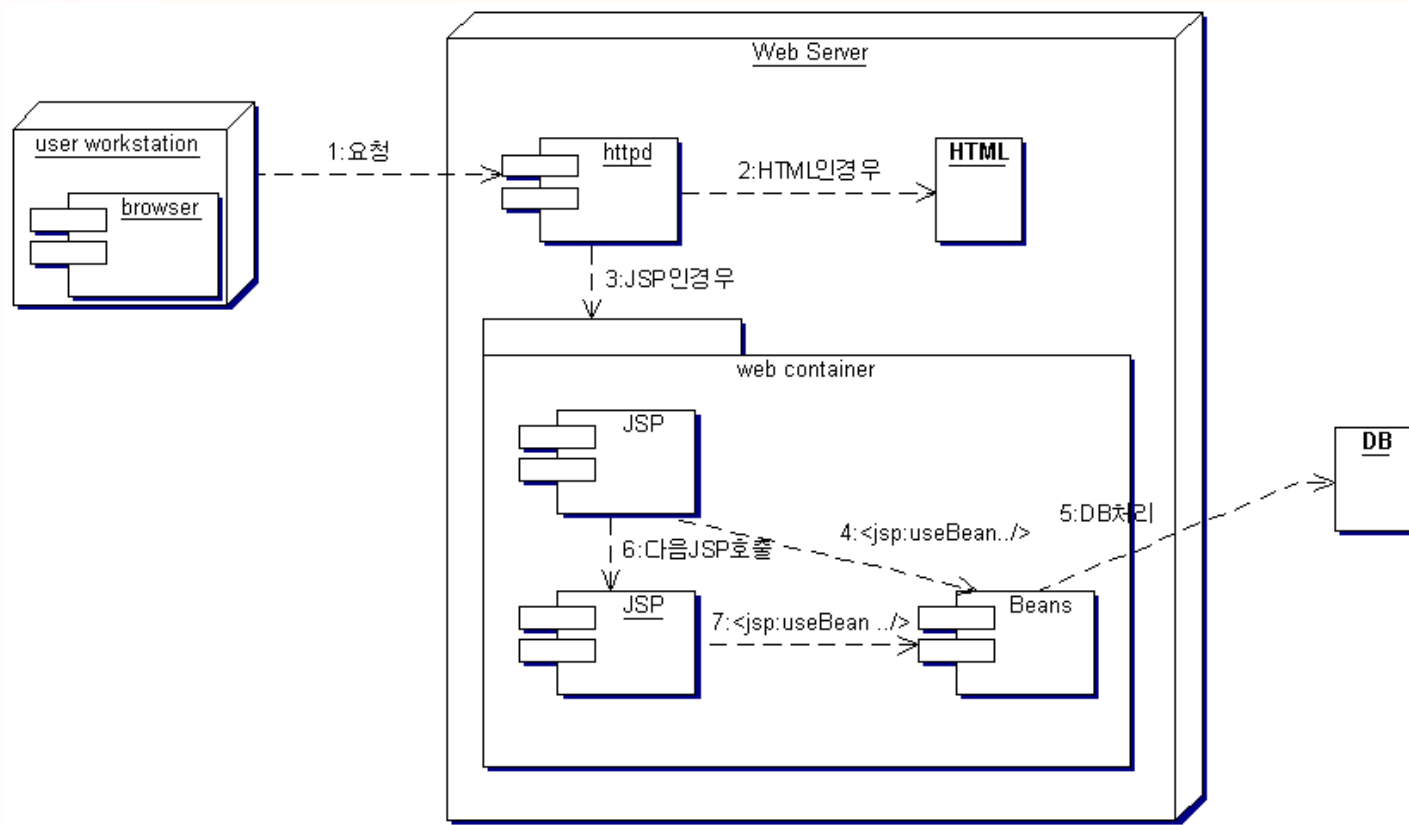
개발 표준화와 유지보수의 효율성

디자인 패턴 학습을 위한 준비

- ☀ 객체지향 개념 이해
- ☀ C++, 자바와 같은 객체지향 언어를 사용한 경험
- ☀ 실제 개발경험

# JSP 모델 1

- 초기 JSP개발에 사용된 모델.
- 일반적으로 사용되는 JSP개발 방법.
- 유지보수의 어려움으로 인해 최근엔 JSP모델-2 가 권장됨.



# JSP 모델 1

## 모델1 소스 예

```
<%  
    String user_passwd = request.getParameter("passwd");  
    if(user_passwd.equals("123456")) {  
    %>  
    <H2> 관리자 메뉴 </H2>  
    <HR>  
    1. <a href=>사용자 추가</a><p>  
    2. <a href=>비밀번호 변경</a><p>  
    3. <a href=>리포트 보기</a><p>  
    <%  
    }  
    else {  
        out.println("<H2>비밀번호가 잘못되었습니다.</H2>");  
        out.println("<a href=check_form.jsp>돌아가기</a>");  
    }  
    %>
```

# JSP 모델 1

## 모델 1 장단점

---

### 모델 1 장점

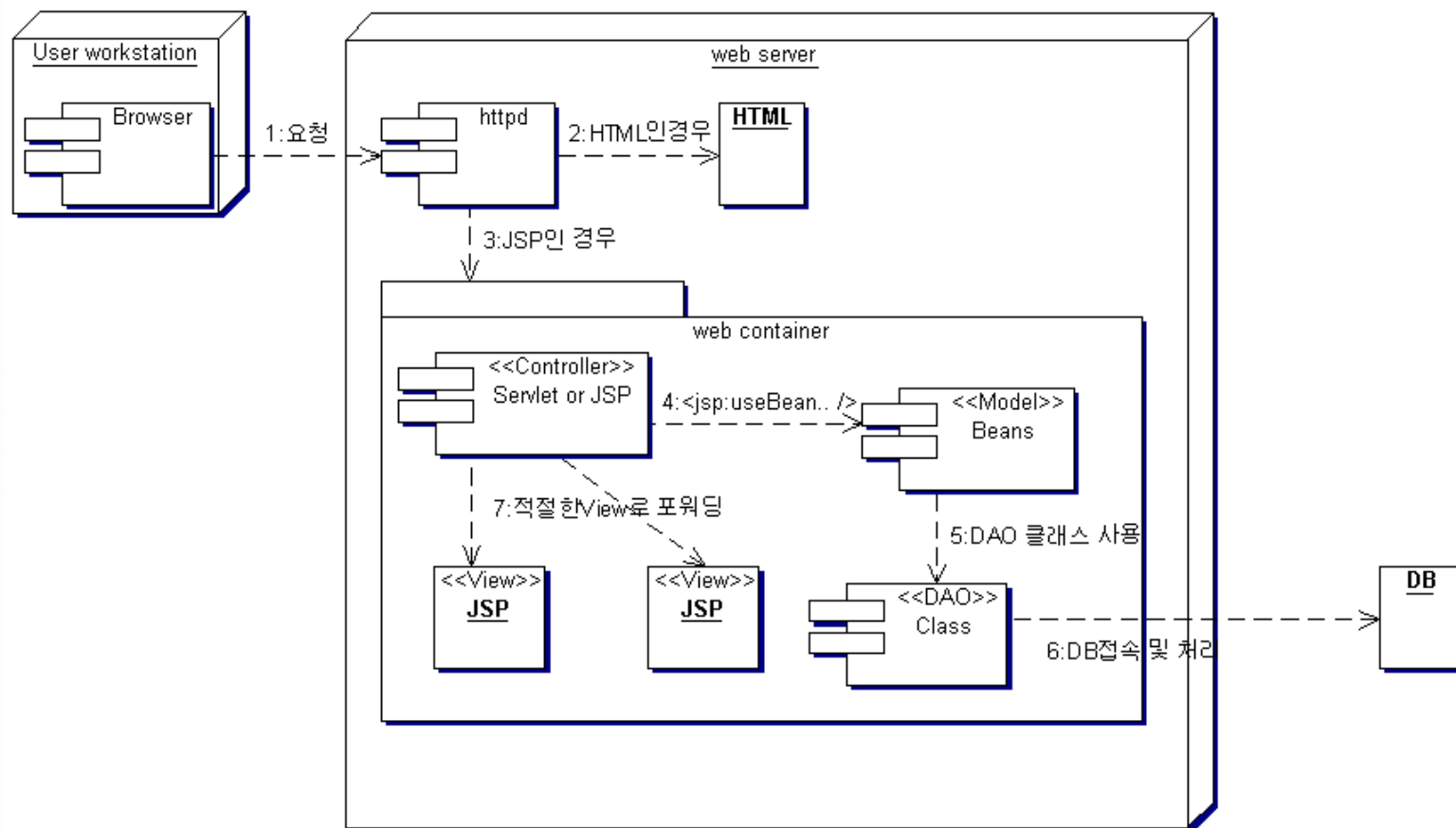
- 페이지 중심 구현 방법으로 초기 개발 속도가 빠르다.
- 적은 인원으로도 개발이 가능하며 컨테이너의 재시동이 필요없다.
- 매우 작은 프로그램 개발에 적합하다.

### 모델 1 단점

- 각종 변화에 대응하기가 어렵다.
- 소스 가독성이 떨어진다.
- 표준화되지 않은 프로그램 구조로 인해 유지보수가 어렵다.

## JSP 모델 2

- MVC(Model-View-Controller) 패턴에 기반.
- 화면(View), 프로그램로직(Controller), 데이터(Model)을 분리.



## JSP 모델 2

### 모델-2 구성

---

#### 모델 (Model)

- 데이터 처리부분 담당.
- Value Object, Data Access Object 으로 분리 하기도 함.
- 데이터베이스와 연동
- 빈즈, 자바 클래스로 구현

## JSP 모델 2

### 모델-2 구성

---

#### 뷰(View)

- 프리젠테이션 영역 즉 사용자에게 보여지는 화면을 담당.
- 스크립트릿을 통한 로직제어(for, if, while) 에 따라 화면을 구성하면 안됨.
- HTML, JSP로 구현
- 커스텀 태그 라이브러리 사용 권장(JSTL 등)



## JSP 모델 2

### 모델-2 구성

---

#### 컨트롤러(Controller)

- 뷰와 뷰를 연결하고 데이터처리를 위해 모델 영역과 연동
- 비즈니스 로직이 복잡하면 BO(Business Object) 클래스를 두어 처리
- 서블릿을 권장하며 간단한 작업에는 JSP로도 가능

## JSP 모델 2

### 모델-2 소스 예

```
<%  
    String user_passwd = request.getParameter("passwd");  
    if(user_passwd.equals("123456")) {  
        response.sendRedirect("admin_menu.jsp");  
    }  
    else {  
        response.sendRedirect("error.jsp");  
    }  
%>
```

- 서블릿 컨트롤러에서는 일반적으로 forward 를 이용한다.
- RequestDispatcher view = request.getRequestDispatcher("admin\_menu.jsp");  
 ;
- view.forward(request,response);

## JSP 모델 2

### 모델 2 장단점

---

#### 모델 2 장점

- 표준에 맞는 개발이 이루어지므로 확장성이 뛰어나다.
- 모듈별 검색이 쉽다.
- 표준화된 코드를 이용하기 때문에 공동 작업이 용이하고 유지보수가 쉽다.

#### 모델 2 단점

- 개발 과정이 복잡해 초기 개발 속도가 늦다.
- 프로그램 로직을 별도의 자바 클래스나 서블릿을 이용해 처리  
→ 소스 변경에 따른 재컴파일, 컨테이너 재시동 등 불편
- 초보자가 이해하고 프로그래밍하기에는 다소 어렵다