

web-server

web-server = web-client의 요청을 받아서 처리 후 응답하는 컴퓨터

tomcat = web-server + web-application-server

web-server = http-요청 + html5 + http-응답

web-application-server = servlet-container + jsp-container

html + servlet + jsp

11, 12-jsp

자바클래스에 html태그 포함 형식 *.java	jsp-html파일내부 자바문장 포함 형식 *.jsp -> *.java-> *.class(tomcat서버)
class A extends HttpServlet doGet(HttpServletRequest, HttpServletResponse) doPost html태그를 출력문장 매개변수 취급	<html> <% 자바문장 ; %> jspService() 내부 문장이나 지역변수 선언

html	<html> <style> <script> 사용 html파일내용==>jsp포함가능
자바 1> <% %> 2> jsp (요청-응답 필요한 객체 미리 생성-제공) 내장객체 request.getParameter() String str = "출력"; <% out.println(str) ; %> ===> <%=str %>	<%! 멤버변수나 메소드 %>-> 선언문-태그 <%-----%>-> jsp 주석문태그 <!----> <% // %> <%= 변수명 %>-> 출력문-태그 <input type="" value=<%= 변수명 %> <%@ page--> page-지시자태그 -->tomcat 에게 jsp-서블릿 소스 변환시에 알려줌

—내장객체

요청-응답관련 객체	request response out	javax.servlet.HttpServletRequest javax.servlet.HttpServletResponse javax.servlet.jsp.JspWriter(PrintWriter 하위클래스)
오류처리관련객	exception	java.lang.Exception

서식 있음: 글꼴 색: 연한 파랑

채		
---	--	--

web server =web client의 요청을 받아서 처리 후 응답하는 컴퓨터

tomcat = web server + web application server

web server = http 요청 + html5 + http 응답

web application server = servlet container + jsp container

html + servlet + jsp

"자바기능 활용 자바문장 최소화"

12장 스크립트태그 + 내장객체

13장 액션태그

14장 el

동작 - 포함/이동/파라미터추가/

자바객체생성동작 태그

<% request.setAttribute("c", "배열/객체/ 문자열/숫자/논리값.); %>

<jsp:include page="동일서버의동일컨텍스트/포함파일명(html,jsp,서블릿url)">

==> 현재파일+incude파일 실행결과 합쳐짐

<jsp:param name="a" value="문자열/숫자/논리값" />

</jsp:include>

<% 응답 포함x (무시), 응답 내용 만들어서 처리 과정 ...%>

<% request.setAttribute("c", "배열/객체/ 문자열/숫자/논리값.); %>

<jsp:forward page="동일서버의동일컨텍스트/이동파일명(html,jsp,서블릿url)">

==> 현재파일 실행응답x +forward파일 실행결과 응답

<jsp:param name="a" value="문자열/숫자/논리값" />

</jsp:include>

- 액션태그 (자바객체생성동작)

<u><jsp:useBean /></u>	<u><% test.A a1 = new test.A();</u> <u>request.setAttribute("a1", a1);</u> <u>%></u> <u><jsp:useBean id="a1" class="test.A" /></u> <u>-->"기본생성자(매개변수 없는 생성자) 호출 변환"</u> <u>bean/ beans ---> 자바객체들</u>
<u><jsp:setProperty /></u>	<u>setter메소드호출동작태그</u> <u><% test.A a1 = new test.A();</u> <u>a1.setName("이자바");</u> <u>%></u> <u><jsp:useBean id="a1" class="test.A" /></u> <u><jsp:setProperty name="a1" property="name" value="이</u> <u>자바" /></u>
<u><jsp:getProperty /></u>	<u>getter메소드호출동작태그</u> <u><% test.A a1 = new test.A();</u> <u>a1.setName("이자바");</u> <u>%></u> <u><%=a1.getName()%></u> <u><jsp:useBean id="a1" class="test.A" /></u> <u><jsp:setProperty name="a1" property="name" value="이</u> <u>자바" /></u> <u><jsp:getProperty name="a1" property="name" /></u>

서식 있는 표

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게

서식 있음: 글꼴: 굵게 없음

MemberDTO 클래스 – data transfer object

변수들 + setter/getter

<jsp:useBean id="" class="java.lang.String" />

expression language – jsp 내부 독립적 언어

\${ i=100}

\${i*i}

\$(내장객체 통하면 jsp 태그 선언변수 사용가능)

`$(논리값 ,정수,실수,문자열, null)`
`$(empty 100)--> null이면 true`
`$(null)--> 공백(NullPointerException 전혀발생x)`
`$(null + 1 } --> 0 + 1 --> 1`

`$(param.파라미터이름)`
`$(paramValues.파라미터이름[0])`

<u>a.jsp</u>	<u>b.jsp</u>
<u><code><% session.setAttribute</code></u> <u><code>("a", "javasession"); %></code></u> <u><code><% request.setAttribute("a", "java"); %></code></u> <u><code><jsp:forward page="b.jsp" /></code></u>	<u><code><% pageContext.setAttribute</code></u> <u><code>("a", "b.jsp"); %></code></u> <u><code><%=request.getAttribute("a")%></code></u> <u><code><%=session.getAttribute("a")%></code></u> <u><code>\$(a)->b.jsp</code></u> <u><code>\$(pageScope.a)-->b.jsp</code></u> <u><code>\$(requestScope.a)-->java</code></u> <u><code>\$(sessionScope.a)-->javasession</code></u>

`$(param.파라미터이름)`
`$(paramValues.파라미터이름[0])`

`$(xxxScope.변수이름)`

`<jsp:useBean id="a" class="test.A" />`
`-----> ${a}`

`<% test.A b = new test.A();`
`pageContext.setAttirbute("b", b);`
`%>`
`-----> ${b}`

-jsp포함 태그 - 표현방법

html 파일 == HTML + CSS + JAVA SCRIPT 복합

<% %>

내장객체

<jsp:액션태그 --> 6가지>

el

#{ paramValues.파라미터이름} --> 배열

java.util.ArrayList

java.util.HashMap

1> 일반 자바 클래스

2> 서블릿이나 jsp <% %> 사용 가능

3> el

MemberDTO 5개 ArrayList 저장 --- el 출력

<u>jsp내장객체(객체도 저장 - 공유)</u>	<u>el내장객체</u>
<u>1>현재페이지의 el과 공유</u> <u>pageContext</u>	<u>1>현재페이지의 jsp 태그로부터 공유받은 정보</u> <u>pageScope</u>
<u>2> forward/include 하는 jsp 파일의 el 공유</u> <u>request</u>	<u>2> forward/include된 파일의 공유받은 정보</u> <u>requestScope</u>
<u>3> 동일 브라우저 jsp 파일의 el 공유</u> <u>session</u>	<u>2> forward/include된 파일의 공유받은 정보</u> <u>sessionScope</u>
<u>4> 동일 서버의 다른 브라우저의 실행 파일의 el 공유</u> <u>application</u>	<u>2> forward/include된 파일의 공유받은 정보</u> <u>applicationScope</u>
<u>공통</u> <u>setAttribute</u> <u>getAttribute</u> <u>removeAttribute</u>	<u>공통</u> <u>#{변수명}</u> <u>#{xxxScope.변수명}</u>
<u>el 공유받은 객체 찾는 순서</u>	<u>#{변수명}</u> <u>pageScope -> requestScope ->sessionScope</u> <u>->applicationScope</u>

14.4 custom 태그

<u>java script --> 별도 설치 파일 없다</u> <u>브라우저 실행</u>	<u>jsp el 표준 실행 가능</u> <u>--> 별도 설치 파일 없다</u>
<u>jquery--> jquery-3.6.0-min.js 설치</u> <u>자바 스크립트 라이브러리</u> <u><script src="jquery..." ></u>	<u>jsp tag library 모음 제공</u> <u>custom tag 일종 = jstl태그</u> <u>-->jstl 태그 파일 설치</u> <u>jstlxxx.jar</u> <u><%@ taglib</u> <u>uri="http://java.sun.com/jsp/jstl/core"</u> <u>prefix="c" %></u> <u>-->jsp파일 template</u> <u><c:out /></u>

<http://mvnrepository.com>

14장 표현언어와 jstl

if-else --> <c:if

switch-case --> <c:choose , <c:when, <c:otherwise

<c:choose >

<c:when test="조건1" >

true html, <% %> \${} , <c:

</c:when>

<c:when test="조건2" >

true html, <% %> \${} , <c:

</c:when>

<c:when test="조건3" >

true html, <% %> \${} , <c:

</c:when>

<c:otherwise>

조건 1, 2, 3 아닌 경우 수행

</c:otherwise>

</c:choose>

<c:forEach items="\${반복대상배열/리스트/맵}" var="반복중인값변수" >
 \${ var 변수명 }
</ c:forEach>

<c:forEach items="\${반복대상배열/리스트/맵}" var="반복중인값변수" varStatus="상태변수" >

\$(varStatus변수명.
</ c:forEach>

- varStatus

<u>index</u>	<u>정수 0~</u>
<u>count</u>	<u>정수 1~</u>
<u>first</u>	<u>boolean . 첫번째데이터여부</u>
<u>last</u>	<u>boolean . 마지막데이터여부</u>
<u>current</u>	<u>반복중인 데이터값</u>

<c:url --> url만 정의변수
<c:import --> url사용 태그
<c:set value=""

15장 파일업로드/파일다운로드==> 스프링

16장 html5

17장 모델2 방식으로 효율적으로 개발하기

모델1 = jsp만

모델2 = mvc ==> 서블릿(요청-처리선택 -jsp선택-응답)+jsp+dao,dto

servlet+jsp 활용 개발 = mvc 패턴 자유-선택

spring framework "강제" mvc

1> servlet + dto, dao 클래스 + jsp

2> 규칙

----1개 웹어플리케이션 ---

1> 기능 설계

1-1> 게시판

게시물리스트 보여주기

제목 클릭한 게시물 상세정보 보여주기

게시물작성하기

수정하기

삭제하기

==> CRUD 기능 + 추가

1-2> 흐름 설계

게시물리스트 보여주기

제목 클릭한 게시물 상세정보 보여주기

게시물작성하기

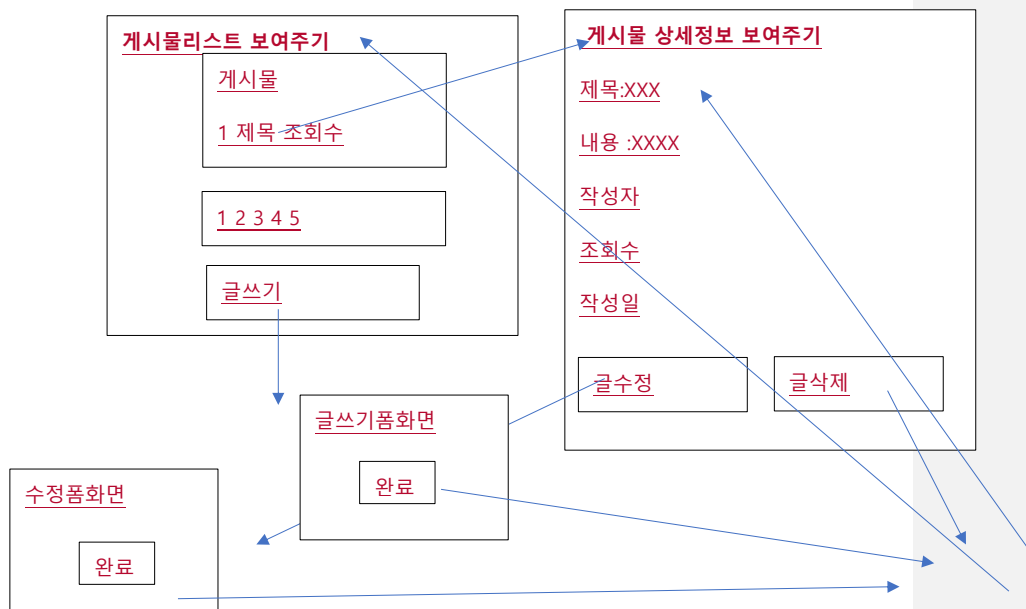
수정하기

삭제하기

<http://ip:port/컨텍스트명/myboard?menu=list>

<http://ip:port/컨텍스트명/myboard?menu=oneboard>

<http://ip:port/컨텍스트명/myboard?menu=boardwriting>

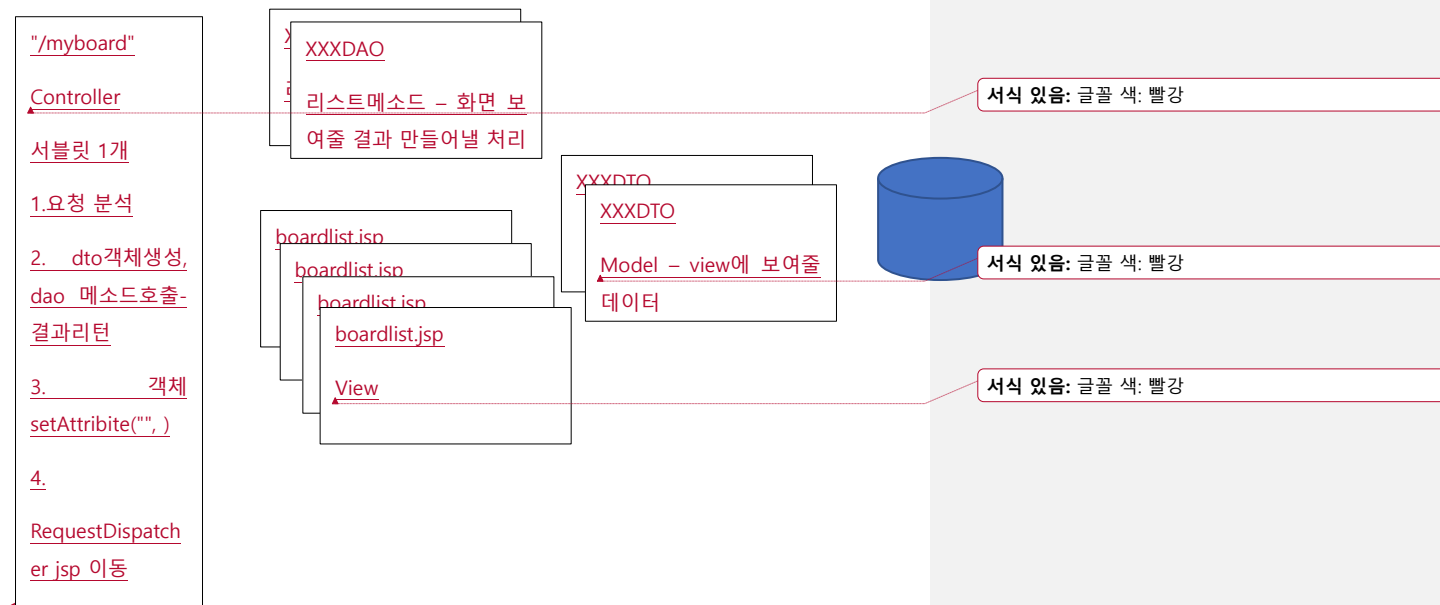


1-3> 테이블 생성

1-4> SQL

1-5> xxxxDAO , XXXDTO 클래스

<http://ip:port/컨텍스트명/myboard?menu=list>
<http://ip:port/컨텍스트명/myboard?menu=oneboard>
<http://ip:port/컨텍스트명/myboard?menu=boardwriting>



`$(document).ready(..)=>jquery`

`$(el 변수 el 연산자)`

`${100} --> 100`

`${ 백+100 } --> 공백`

`<% out.println(); %>`

`value='<%=a %>'`

`value=null`

변수

논리값/정수/실수/문자열/null

`$(true)`

`$(false)`

`$(3.14)`

`#{'aaaaa'}`
`#{ "aaaaa" }`
`#{null}`
연산자

+ 산술연산자
+= 문자열결합연산자
`#{'백' + 1 }` ---> error
`#{'백' += 1 }`
`#{null + 1 }==> null '+' 연산시 자동 0`
`#{10 / 3 }-->실수`
`#{10 % 3 } --> 나머지`
`#{empty 1 }-->?`

- el과 jsp태그

pageContext 내장객체
- jsp 파일당 1개씩
- 같은 jsp 내부 el값 공유

pageContext 내장객체
`<% int i = 10;`
`pageContext.setAttribute("jsp_i" , i);`
`%>`
`#{ jsp` pageContext 내장객체
`#{ j= 100; }==>자동`

`pageContext.setAttribute("j" , j);`
`<% int i = 요청파라미터값정수변환`
`<% pageContext.getAttribute("j")%>`

서식 있음: 글꼴 색: 빨강

<u>jsp태그</u>	<u>el</u>	
<u><code>pageContext.setAttribute("a", "b");</code></u>	<u><code>#{ a }</code></u>	<u>같은 jsp파일 내부</u>
<u><code>pageContext.getAttribute("a");</code></u>		
<u><code>request.setAttribute("a", "b");</code></u>	<u><code>#{a}</code></u>	<u>다른 jsp파일 (이동/포함)</u>

서식 있는 표

<u>request.getAttribute("a");</u>		
<u>session.setAttribute("a", "b");</u> <u>session.getAttribute("a");</u>	<u>\${a}</u>	<u>다른jsp파일(동일브라우저)</u>
<u>application.setAttribute("a", "b");</u> <u>apliation.getAttribute("a");</u>	<u>\${a}</u>	<u>서버지속-다른브라우저</u> <u>도</u>
<u>pageContext.setAttribute("a", "b1");</u> <u>request.setAttribute("a", "b2");</u>	<u>\${a} --> b1</u> <u>\${requestScope.a}--> b2</u> <u>\${sessionScope.a}-->"</u> <u>\${applicationScope.a} --></u> <u>"</u> <u>_</u>	