

web server

web server = web client의 요청을 받아서 처리 후 응답하는 컴퓨터

tomcat = web server + web application server

HTML응답 + 서블릿자바코드실행 - 태그 응답

서블릿 기본 api DOGET DOPOST

HttpServlet 상속 - doGet / doPost 1개 이상 오버라이딩 필수

HttpServletRequest / HttpServletResponse - doGet / doPost 매개변수 선언

서블릿 확장 api

- DataSource 활용한 jdbc

1> jdbc 코드 반복 - jdbc driver 로드 / db 연결 / / db 연결해제

2> db 연결(driver, url, user, password) 메모리/ 시간 소요 작업 - db 연결해제 메모리 삭제 작업

3> db 외부 접속 최대 허용치 한정 -

4> tomcat 시작시에 "미리" 생성 - 반납 관리 - 재사용

==> connection pooling 구현 클래스

- javax.sql.DataSource 인터페이스를 상속받은 하위클래스--> tomcat 제공

연결 몇개, driver, url, user, password, ---> "jdbc/a" --> ServersW....Wcontext.xml 1번 설정

5> 자바 jdbc driver, url, user, password 소스 드러나지 않는다

| 필요한 라이브러리 | context.xml | dao, servlet |
|--|--|---|
| 컨텍스트 /src/main/webapp/web- inf/lib/mysql jdbc driver.jar tomcat-dbc.jar | <Resource name="jdbc/a" jdbcdriverclass= user= password url maxActive type="javax.sql.DataSource" | Context c = new InitContext(); Conetxt d = (Context)c.lookup("java:comp/env"); DataSource d = (DataSource)d.lookup("jdbc/a"); Connection con = d.getConnection();//미리생성빌려온 다 |

| | | |
|--|--|------------------|
| | | con.close();//반납 |
|--|--|------------------|

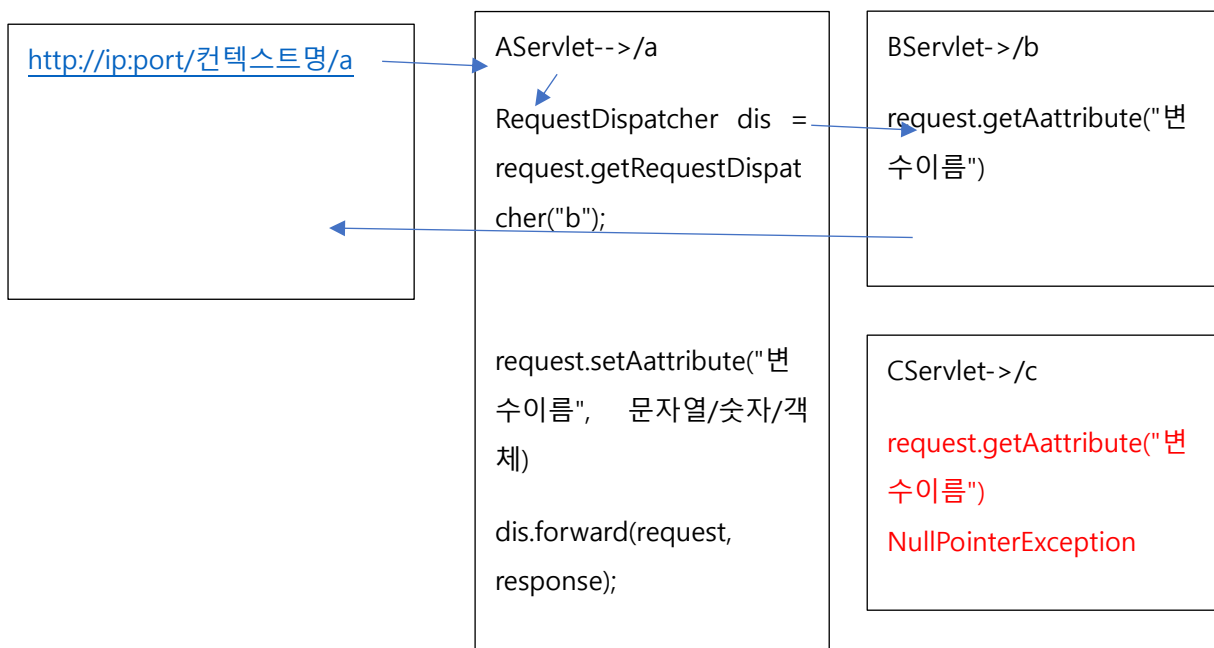
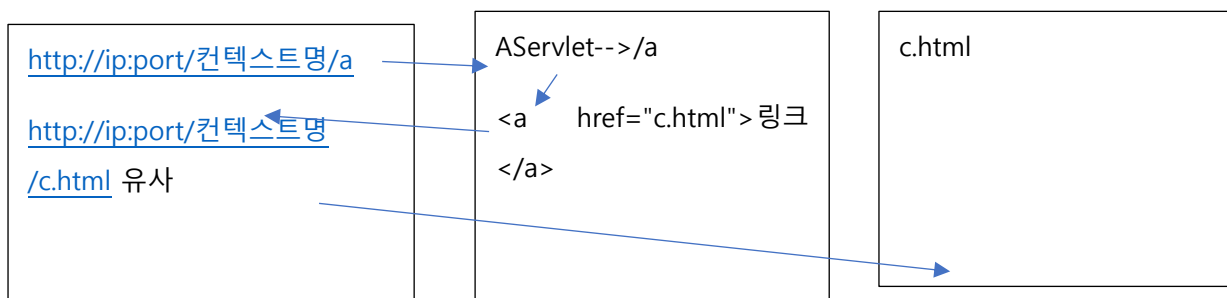
- RequestDispatcher를 이용한 서블릿간 자동 이동

다른파일이동 </ a>

1> 링크 클릭하면 이동

2> 파라미터 공유 불가

3> 자바 객체 해석 불가 / 객체 정보 공유 불가



- HttpSession을 이용한 클라이언트 상태 정보 유지

1> 동일 클라이언트 내부 모든 서블릿 정보 공유

2> 동일 ip 컴퓨터의 동일 브라우저 내부 실행 동일 서버의 동일 컨텍스트의 서블릿간

3> 브라우저 종료/30분 no action / 시간 조정 가능

4> 세션 객체 저장 정보-서버컴퓨터 저장

5> 여러개 클라이언트 / 1대 서버 물리적 메모리 크기 제한

6> 생성

```
HttpSession session = request.getSession();
```

브라우저 동일 서버 동일 컨텍스트 내 서블릿 최초 요청

서버 내부 브라우저 정보 저장 세션 객체 생성하고 세션id 부여

브라우저 동일 서버 동일 컨텍스트 내 서블릿 2번째이상 요청

세션id 받아서 해당 브라우저 확인하고 세션 객체 리턴

7> 사용

```
session.setAttribute("변수이름", 문자열/숫자/객체)
```

```
session.getAttribute("변수이름")
```

8> 삭제

```
session.removeAttribute("변수이름")
```

```
session.invalidate();
```

```
//session.setMaxInactiveInterval(60초);
```

RequestDispatcher 2개 이상 서블릿이 협업해서 1개 요청 처리 1개 응답

---> 정보 공유 대상 2개 서블릿간

10장 ServletFilter / ServletListener --- spring mvc 포함 내장 구현 기술

=====서블릿 =====

11. JSP 역할과 구조

서블릿

1> db 이용

2> 변수 공유, 객체 사용

3> 요청 - 처리 - 응답 기본 구조

4> 응답 - 브라우저 언어 html 태그 전달 불편

```
response.setContentType("text/html;charset=utf-8");
PrintWriter out = response.getWriter();
out.println("<h1>" + sessionid + " 회원님 로그아웃되었습니다.</h1>");
```

| | |
|---|---|
| 서블릿 자바문장 - A.java - A.class src/main/java 폴더 패키지 저장 | jsp 자바문장 - *.jsp src/main/webapp/*.html, *.jsp |
| <pre>@WebServlet("/a") class A extends HttpServlet doGet(HttpServletRequest req, HttpServletResponse res){ 요청 - 처리 - 응답 } }</pre> | <pre><html> <style> <script> <% request.getParameter("id"); System.out.println(); %></pre> |
| 클래스 내부에 html 태그 포함 형태 | html 파일 내부 자바문장 포함 형태 |
| | <p>jsp 실행과정(tomcat 자동)</p> <p>http://ip:Port/컨테스트/*.jsp 실행</p> <p>1> *.jsp 파일 내부의 모든 내용을 서블릿 소스로 변환한다. ---> *.java</p> <p>2> 컴파일한다 --> *.class</p> <p>3> 실행한다 결과 브라우저 응답</p> |

tomcat = html/css/java script (파일 찾아서 응답) + servlet + jsp(서버 실행+응답)

= web server + web application server

servlet container + jsp container

11.3 jsp 구성요소

| | |
|--|--|
| html 요소 | html css java script jquery 태그나 문장 |
| <jsp:directive> | 디렉티브태그 |
| <% 자바문장 그대로; %> | 스크립트태그 |
| \${param.type} | 표현언어(쉽고 간결하게) |
| session.setAttribute("", 객체); | 내장객체 |
| <% RequestDispatcher d = request.getXXXX(""); d.forward(요청, 응답); %> <jsp:forward page="" /> | 액션태그 |
| <c:forEach /> | 커스텀태그 내가 만들어서 jstl library 개발 태그- *.jar 설치 |

- 디렉티브태그=지시자태그=tomcat 지시 내용 정의

| | |
|-----------------------------------|------------------------------|
| <%@ page import contentType | 현재 jsp 페이지 여러 설정 |
| <%@ include | 다른 파일 포함 실행 <jsp:include |
| <%@ taglib | 표준 jsp 태그 외 다른 단체 개발 태그모음 사용 |

- 스크립트태그

| | |
|-----------------------|--|
| <% 자바 문장 종류 상관없이 ; %> | 자바언어=스크립트언어설정 |
| <%-- --%> | jsp 주석문(html 주석문 <!-- -->) |
| <%= %> | 표현문태그(출력) |
| <%! 멤버변수나 메소드 %> | 선언문태그 _jspInit() _jspDestroy() _jspService() 자동 생성 jsp 모든 코드들은 _jspService() 내부 실행 |

| | |
|----------|-----------|
| | 멤버변수나 메소드 |
| <%@ page | |

- 주석문

1> 자바문장 – 자바 주석문

```
// , /* */
```

2> html태그 – html 주석태그

```
<!-- -->
```

3> jsp 주석문

```
<%-- --%>
```

| | jsp | 서블릿 소스 변환 | 브라우저 전송 |
|--|------|-------------------|------------------------|
| <%-- jsp주석문 --%> | 포함 o | 주석으로 인식 시점 포함x | 포함x |
| <% // 자바 주석문 %> | 포함 o | 포함 o | 주석으로 인식 시점 포함x |
| <!--html 주석문 이 파일 저작 버전 최적 --> | 포함o | 포함o | 브라우저 해석 시점 소스보기 포함o |

login.html-->LoginServlet (id, pw + db + session 저장)

input.jsp (2-9단까지)

시작단 : 3

종료단 : 7

구구단출력버튼

| | |
|--|---|
| input.jsp (2-9단까지) | output.jsp?start=3&end=7 |
| <p>시작단 : 3</p> <p>종료단 : 7</p> <p>구구단출력버튼-output.jsp 이동</p> <p>--></p> | <p>테이블태그로 + for문</p> <p>3*1=3 4*1=4 7*1=7</p> <p>3*2=6 4*2=8 7*2=14</p> <p>....</p> <p>3*9=27</p> <p>구구단 입력하기 링크</p> <p>테이블 가운데 정렬</p> <p>선 색상 파랑색.</p> <p>홀수줄 – 바탕색상 pink</p> <p>짝수줄 – 바탕색상 silver</p> |

18. 스프링 legacy

32. 스프링 boot