

- 웹 서버 프로그래밍

http client

=web client

1. 브라우저

url 입력 - 내용 실행 표현 - html

http://127.0.0.1:9090/경로/*.html

4. 브라우저 표현 출력한다

(db 데이터 , 서버내부 파일 정보....

http server

=web server

2. 요청 분석한다

3. html 파일을 찾아서 클라이언트에게 전송한다

- 웹사이트 영구적 저장 데이터 쌓여간다--> db

http client

=web client

1. 브라우저

url 입력 - 내용 실행 표현 - html

<http://127.0.0.1:9090/경로/servlet.jsp>

자바소스실행

5. 브라우저 표현 출력한다

(db 데이터 , 서버내부 파일 정보....

http server

=web server

2. 요청 분석한다

3. 자바소스 실행한다.(servlet, jsp)

4. 실행 결과를 전송한다

-> css, html태그, js

웹클라이언트실행프로그램 =웹브라우저 실행	웹서버실행프로그램 = 웹요청(url)+자바실행 =tomcat
html css javascript jquery	asp asp.net php cgi 자바작성 웹서버실행 servlet , jsp

요청 - 서비스 처리 - 응답

웹서버 실행 / 결과 html 파일 넣어서 표현 프로그램

= 자바 실행 웹어플리케이션= servlet , jsp

tomcat = 웹서버(요청 - 응답) + 웹어플리케이션서버(servlet,jsp실행)

- 목차

1 2장 개념 웹어프리케이션 요청 응답

3장 jdk tomcat eclipse ORACLE 설치

4장 이클립스에서 웹어플리케이션 설정-98P

웹어플리케이션 = HTML 태그 + CSS + JAVASCIPT + SERVLET + JSP
 + 자바 클래스 + XML (설정)

이클립스(자동) => **DYNAMIC WEB PROJECT** = 웹어플리케이션 저장

묶음

DYNAMIC WEB PROJECT = 웹어플리케이션 이름

src\W*.java

library\Wtomcat library + jdk library

webcontent\W*.html *.css *.js *.gif *.mp3

*.jsp

WEB-INF\lib\W*.jar

web.xml(서블릿과 jsp 설정)

tomcat = 웹서버(요청 - --응답)

웹어플리케이션서버 = 서버실행 = container 컨테이너

*.html 파일 요청 - 응답

servlet, jsp 요청 - 웹어플리케이션서버가 실행 - 응답

5 6 7 8 9 장 서블릿 프로그래밍

4/7. 8

class A extends HttpServlet

--> http 요청 특화 처리 doGet/doPost 선택적 오버라이딩

class B extends GenericServlet

--> Servlet 메소드 일부 구현/ 나머지 오버라이딩

class C implements Servlet ---> 5개 메소드 모두 오버라이딩

- 서블릿 요청 - 실행 - 응답 과정

1. 클라이언트1는 브라우저에서

[HTTP://ip:port/프로젝트명/서블릿url](http://ip:port/프로젝트명/서블릿url) 요청한다

2. 서버는 웹서버가 웹어플리케이션 서버로 위임한다.

3. 위임받은 웹어플리케이션 서버가 메모리 서블릿 객체 로딩되어 있는 않으면 서블릿 객체 메모리 로딩한다.

4. 서블릿 객체 멀티스레드 실행한다.

5. **init** 호출한다.-서블릿 처리 이전 초기화 구현

6. **doGet** 호출하여 응답한다.

7. 클라이언트2는 브라우저에서

[HTTP://ip:port/프로젝트명/서블릿url](http://ip:port/프로젝트명/서블릿url) 요청한다

8. 서버는 웹서버가 웹어플리케이션 서버로 위임한다.

9. **doGet** 호출하여 응답한다.

10. 서버 종료나 서블릿 소스 변경 - 재컴파일시에는 서블릿 객체 메모리 삭제한다.

11. **destroy()** 호출한다.- **init** 변수 메모리 삭제

내컴퓨터 웹클라이언트

내컴퓨터 웹서버(tomcat)

전세계 모든 http 컴퓨터 웹서버 내부 구조 잘 모르도록 보안 유지

<http://ip:9090/servlettest/test> 입력

<http://127.0.0.1:9090/servlettest/servlet/servlettest.TestServlet> 입력

--> servlet 2.x 기본

1> web.xml

2> servlet 4.0 기본

@WebServlet("/test")

class TestServlet

tomcat -> servlettest 웹어플리케이션 찾는다

servlet/servlettest.TestServlet 찾는다

6장

html파일

1> db연동 x

2> 다른 서버 내부 파일 접근 x

- servlet

1> 자바

2> jdbc, io

@WebServlet("/a")

class A extends HttpServlet {

doGet(HttpServletRequest req , HttpServletResponse res){

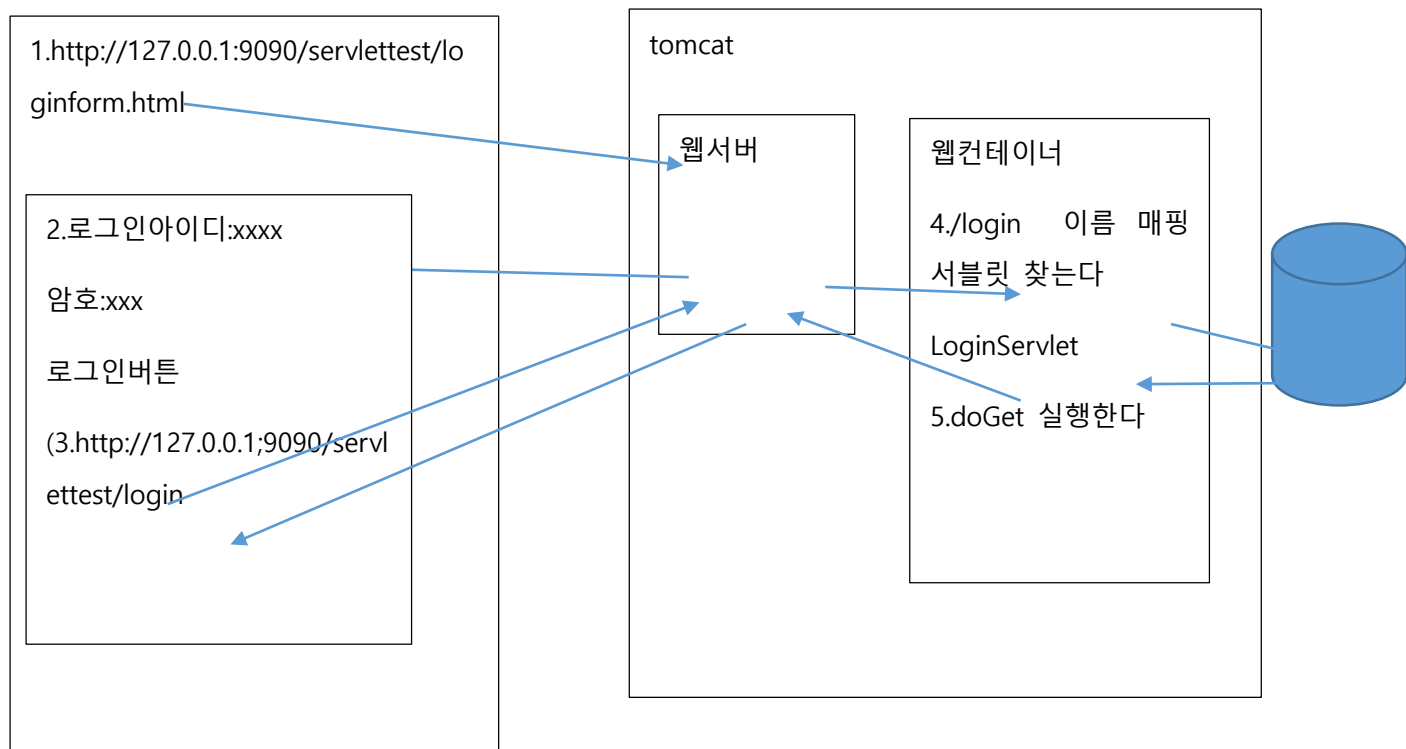
요청(클라이언트ip, id, pw form 태그 전송)

- 처리 (id, pw - member 테이블 id 존재 / pw 맞는지 확인)

- 응답(id, pw 존재 -정상 응답, 비정상 응답- 브라우저 출력-html 파일 전송)

}

}



http 규칙

1> 요청 url

2> http://ip:port/경로/*.html

2> <http://ip:port/경로/>서블릿매핑url

3> 클라이언트는 서버에게 처리 필요 데이터 전송 가능

4> get | post

get	post
url뒤에 데이터 전송 url 에서 ? 뒷부분 name1=value1 & name2=value2... 전송 데이터 길이 250자 1.<form action="login" > 2.<form action="login" method=get > 3.주소 입력 : http:..... 4. 서블릿 이 동 get - 기본 방식 doGet(...)	url 과 별도 분리 데이터 전송 url 뒤 안보인다 (url + ip + post데이터들) 보안 강하다 파일 업로드 길이 긴 데이터 전 송 유리하다 서블릿에서는 속도 느리다 1.<form action="login" method=post> doPost(...)

http 규칙

500번 - 서블릿이나 jsp 예외발생

405 -

get방식 - doPost()

post방식 - doGet()

gugudan.html

출력단 [몇단]

2-20eks

gugu – servlettest.GugudanServlet 작성

2-9단까지만 출력

1> 사용자입력단만 (2-9단 사이면)출력-테이블

2>1, 0, -1, 20(2-9단 사이가 아니면) : 2-9단까지 모든 출력 테이블

3> 입력값 없음 기본 2단만 출력

get + post

BoardMain	BoardInaertView	BoardDAO	BoardDTO
메뉴	input()	insertBoard(BoardDTO)	
글쓰기	제목	ArrayList<BoardDTO>	
게시물리스	내용	getBoardList()	
트	작성자		

11 12 13 14 15 JSP 프로그래밍

4/9, 4/12

16장 HTML5 – JQUERY

17장 이후 스프링

4/13, 14, 15, 16, 19

4/20 특강

4/21 - 23 간단 세미 프로젝트 / 1인 프로젝트