

JSP에서 공부할 핵심 개념은 이렇습니다.

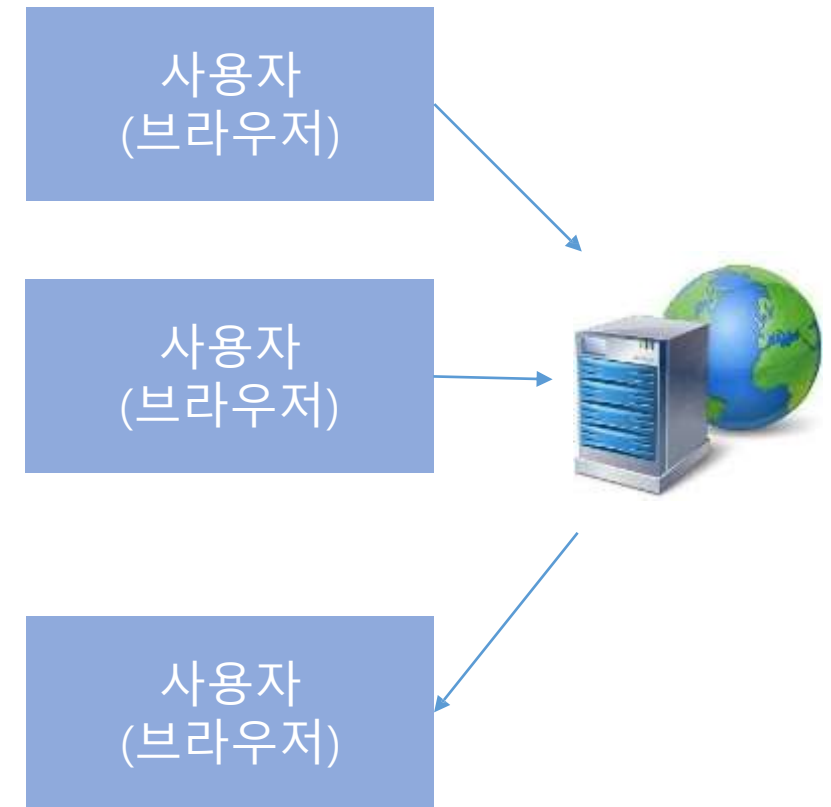
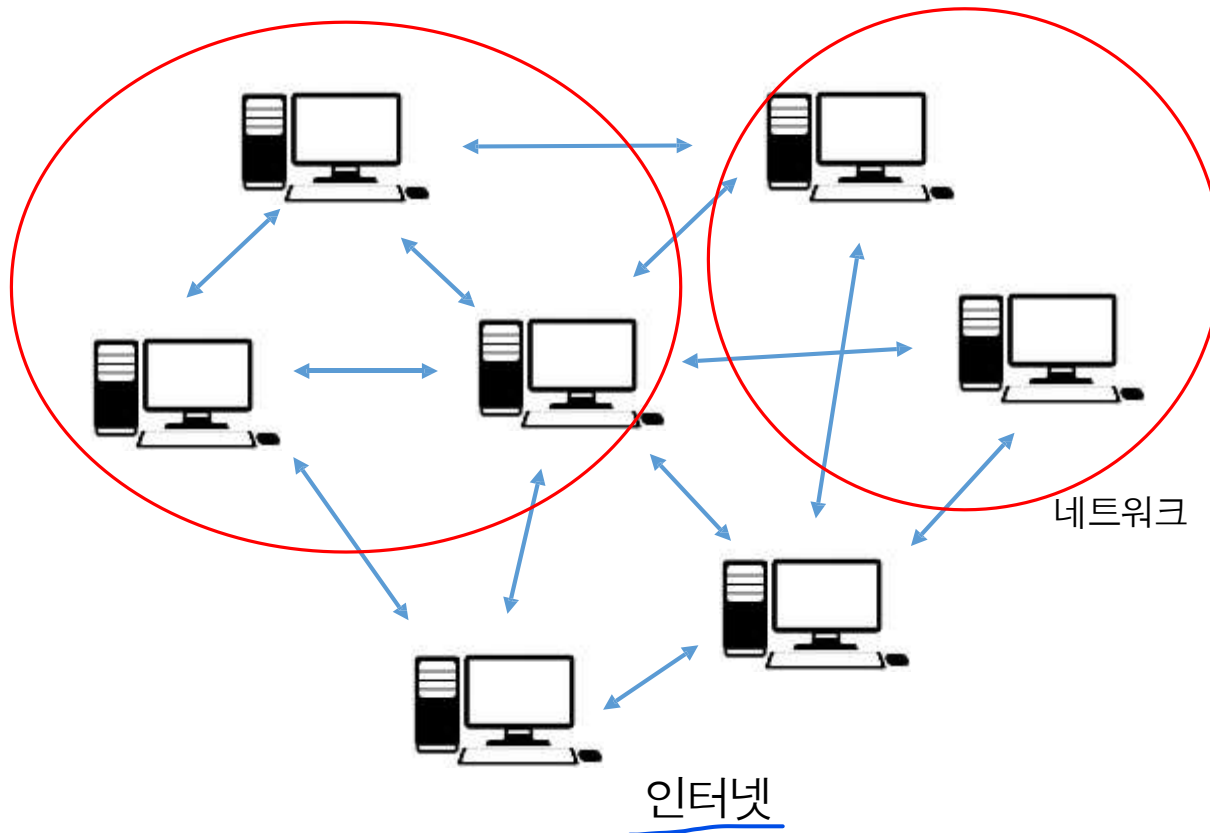
1. JAVA언어 복습
2. 화면구현 언어 HTML의 미리보기
3. 서버 VS 클라이언트
4. 서버사이드 렌더링
5. Request(요청), Response(응답), session
6. 포워딩
7. MVC2 모델
8. 게시판 맛보기

JSP

- 웹 프로그래밍이란?
- URL개념
- 웹 프로그래밍 가동 원리
- HTML학습

웹 프로그래밍이란?

- 웹 프로그래밍이란, 웹 어플리케이션을 구현하는 행위입니다.
- 웹 어플리케이션이란 웹을 기반으로 작동되는 프로그램입니다.
- 웹이란 1개 이상의 사이트가 연결되어 있는 인터넷 서비스의 한가지 형태입니다.
- 인터넷이란 1개 이상의 네트워크가 연결되어 있는 형태를 말합니다.

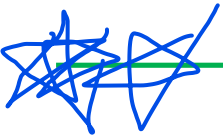


* 웹 어플리케이션의 장점

- **사용자:** 별도의 설치 없이 프로그램을 사용할 수 있기 때문에 접근성이 용이.
- **개발자:** 사용자용 프로그램을 별도로 제작하지 않고, HTML을 이용하여 사용자UI를 구축하기 때문에, 개발에 소요되는 시간과 노력이 줄어듬.

* JSP 웹어플리케이션의 장점.

1. 풍부한 java API 를 사용하여 제작됨.
2. JSP기반의 웹어플리케이션은 특정 운영체제가 아니더라도 호환성의 문제가 없음.
3. 실제 코드가 프로그램에서 노출되지 않기 때문에 보안상의 장점이 있음.



* URL

http://192.110.32.11:8080/

1. **프로토콜(protocol)**: 네트워크상에서 약속한 통신규약 (HTTP, FTP, SMTP, POP, DHCP)
2. **IP주소**: 네트워크상에서 특정 컴퓨터를 식별할 수 있는 주소
3. **Port 번호**: IP주소가 컴퓨터를 식별할 수 있게 해준다면, 포트 번호는 해당 컴퓨터의 구동되고 있는 프로그램을 구분할 수 있는 번호.
4. **DNS(도메인)**: IP주소를 인간이 쉽게 외우도록 맵핑한 문자열.

인터넷 서비스 구분

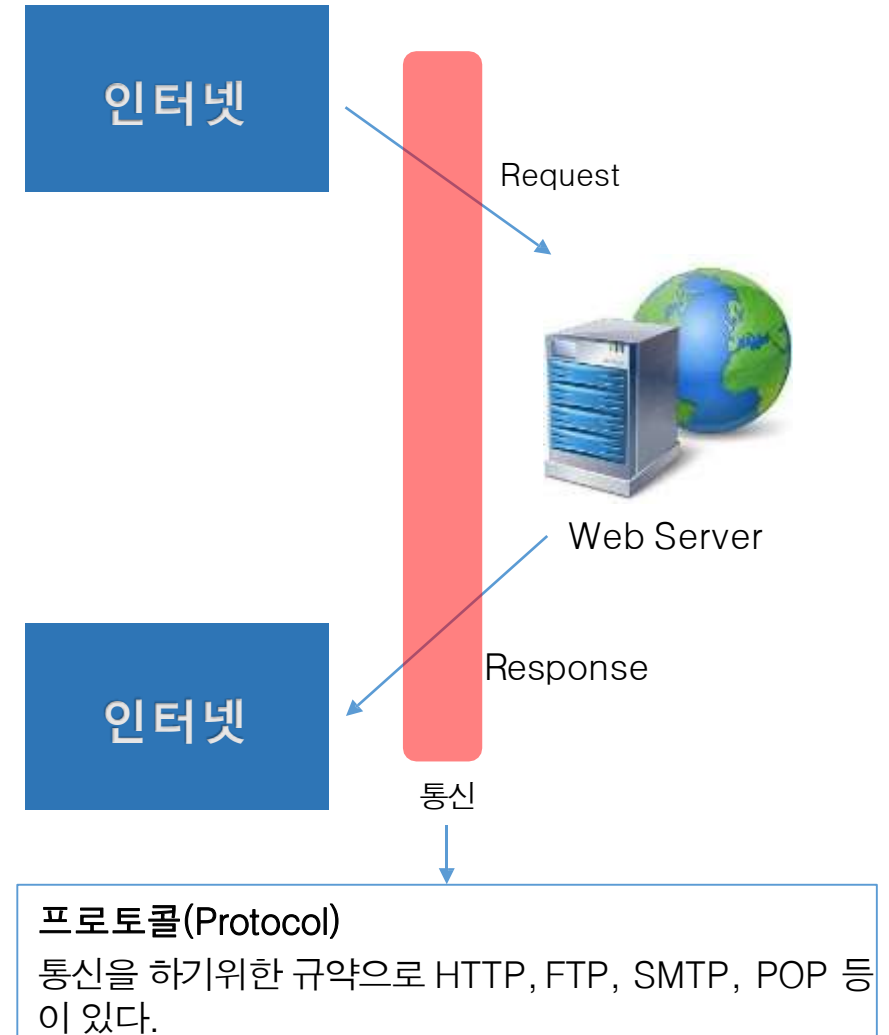
http://www.google.com80/index.html

protocol

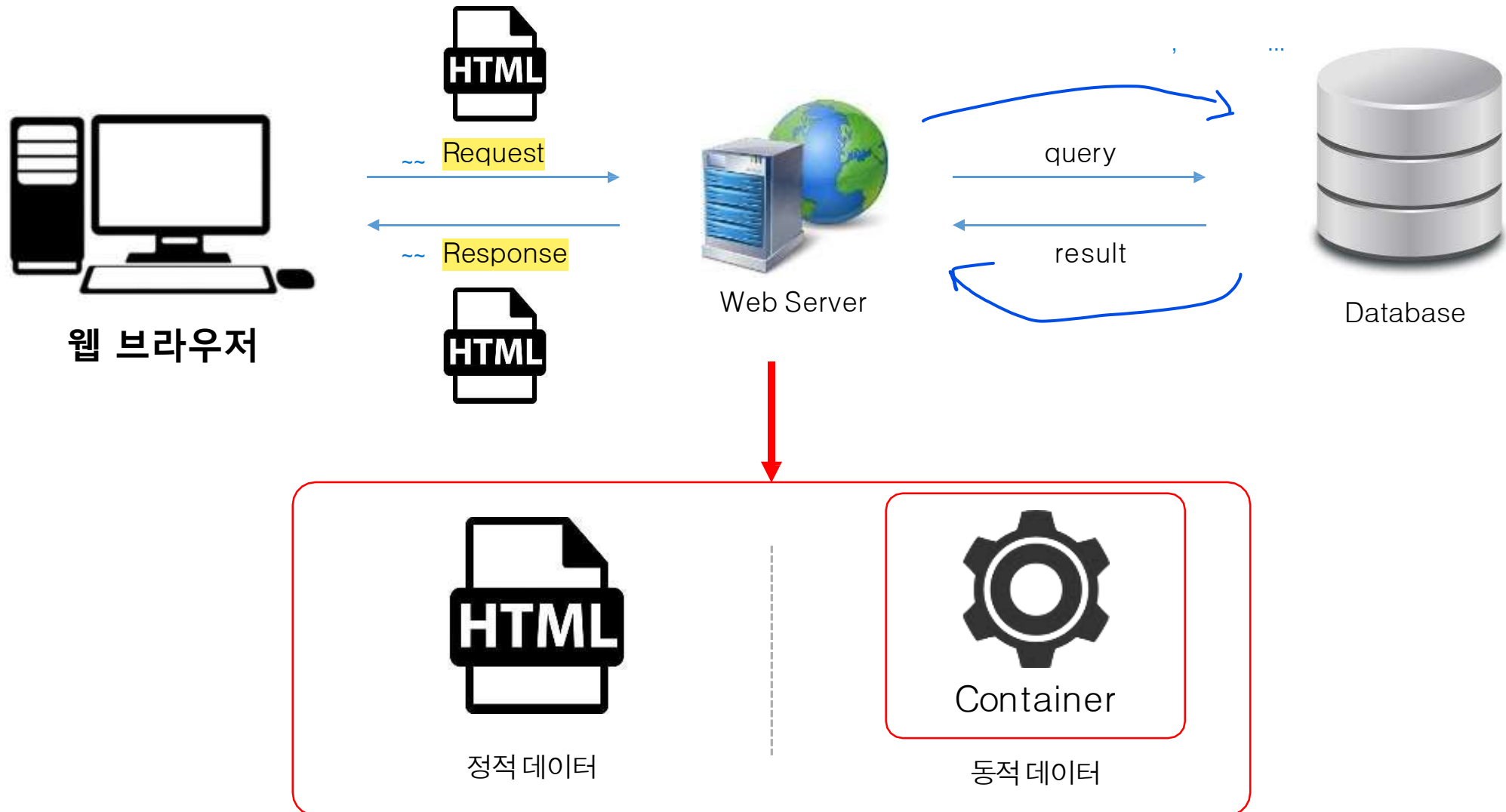
도메인

port

경로



* HTTP 요청과 응답(중요!)



* HTTP 요청과 응답

- 브라우저를 통해 인터넷의 자원을 사용할 때 대부분의 경우 HTTP 프로토콜을 사용합니다.
- HTTP 프로토콜은 hyper text문서 즉, HTML 문서를 전송하는 프로토콜입니다.
- (protocol: 네트워크상에서 약속한 통신규약)
- 브라우저가 서버의 페이지를 **요청(request)**하면 서버는 해당 파일을 찾은 다음 HTTP **응답(response)**를 통해 클라이언트에 전송하는 방식으로 사용자의 요청을 처리합니다.
- 서버로부터 전송된 페이지를 응답페이지라고 하며, 브라우저는 이 응답페이지를 해석(렌더링)하여 화면에 보여줍니다.

==

* **웹 서버**: 클라이언트의 요청(request)에 의해 정보를 제공해 주는 서버.

* **웹 브라우저**: 웹 서버에 정보를 요청하고, 웹서버로부터 정보를 받는 매개체, 이 때 HTTP 프로토콜을 사용함.

* JSP(Java Server Pages)

HTML JAVA

- JSP는 HTML 페이지를 만들기 위한 방법 중 하나입니다.
- JSP는 사용자가 코드를 작성한 다음 컴파일 과정을 거치지 않고 서버에서 실행시킬 수 있습니다.
- JSP는 웹 서버에 의해 **서블릿**으로 변환된 다음 실행됩니다.

* Java Servlets

- 서블릿은 서버에서 실행되는 **자바 프로그램**입니다.
- 서블릿이 사용자에게 의해 요청되면 서버에서는 HTTP request에 응답하기 위해 웹 서버에서 서블릿 객체를 생성하고 이를 통해 사용자의 요청을 처리합니다.



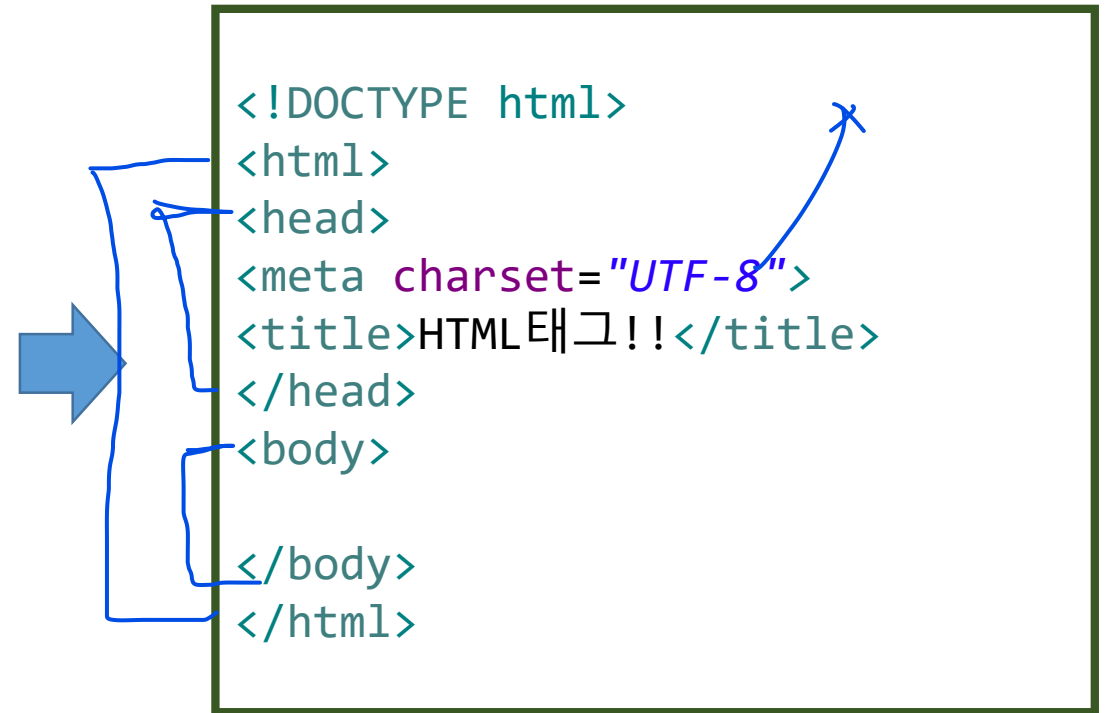
* 웹 프로그래밍을 하기 위해 필요한 학습.

1. JAVA: JAVA웹 어플리케이션을 구현하기 위한 선행학습.
- ✓ 2. **HTML**: 웹 어플리케이션을 구현하기 위한 틀.
- ✓ 3. CSS: 웹 어플리케이션의 레이아웃 및 스타일을 지정하는 언어.
- ✓ 4. JavaScript: 클라이언트의 기능/동작을 구현하기 위한 언어.

* HTML

- Hyper Text Markup Language의 약자
- 정적인 웹페이지를 만드는 언어
- 메모장, 한글, 워드, 문서작업이 가능한 프로그램에서는 모두 작성 가능하다.
- HTML 문서는 **태그**라는 명령어로 구성되어있다.
- 태그의 기본 형식

<태그이름> 문서의 내용 **</태그이름>**
여는태그 닫는태그



* HTML 태그

HTML은 웹 페이지를 만드는 언어 입니다

HTML태그들은 각각 기능을 가지고 있으며, **블럭요소**, **인라인 요소**로 크게 나뉩니다

추후 css에서 다시 설명합니다.

블럭요소

- 1.블럭요소는 블럭의 성질을 가집니다. 줄바꿈과 블럭크기를 가집니다.
- 2.블럭요소는 블럭요소를 자식으로 가질 수 있습니다

```
<h1></h1>
<h2></h2>
<p></p>
<div></div>
<div></div>
<ul></ul>
<ol></ol>
<table></table>
<form></form>
```

등등

인라인요소

1. 인라인 요소는 줄바꿈이 없습니다 (한줄로 출력)
2. 인라인 요소는 블럭요소를 자식으로 포함할 수 없습니다.

```
<span></span>
<strong></strong>
<small></small>
<a></a>

<input>
```

등등.

* HTML form 관련 태그

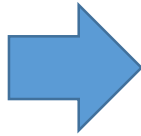
- input태그: 데이터를 입력하기 위해 사용되는 태그. 속성으로 태그의 종류를 지정합니다.
- type 속성: 태그의 종류를 지정

type 속성	기능
1. text	일반적인 text데이터를 입력하기 위해 사용.
2. password	로그인, 회원가입 페이지 등에서 비밀번호 입력을 하기 위해 사용.
3. checkbox	데이터 값을 여러 개 전송해야 할 때 사용.
4. radio	여러 개의 데이터 값 중에 한 개의 값만 전송해야 할 때 사용.
5. select	리스트 형태의 데이터를 사용.
6. textarea	여러 줄의 텍스트를 입력하기 위해 사용. 속성 cols와 rows로 행과 열의 크기를 지정.
7. button	버튼 형태를 만들기 위해 사용. 속성으로 value를 지정하여 버튼 안에 들어갈 문자열을 입력.
8. submit	form내의 데이터를 서버로 전송할 때 사용.
9. reset	form내의 데이터를 초기화할 때 사용.
7. file	파일 업로드를 위해 사용.

-개발 환경 설정 (word 파일을 참조하세요)

* JDK 설치

- java.sun.com 접속
- JAVA SE 1.8버전 다운로드
- 자바 환경변수 설정.
 1. 시스템 변수에 새로만들기 JAVA_HOME에 jdk가 설치된 경로 지정.
 2. 시스템 변수 Path에 편집을 눌러 ;%JAVA_HOME%\bin; 추가.



* Eclipse 설치

- eclipse.org 접속
- JAVA EE버전 다운로드.

* tomcat 웹 서버 설치

- tomcat.apache.org/
- 톰캣 9.0 버전

<http://localhost:8181/>

* 프로젝트 export/import 방법

1. **export:** 프로젝트 우클릭한 후, export 누르고 General탭에 Archive File 저장명 입력하고 finish
2. **import:** 이클립스 file -> import , Existing ~~ 누르고 select Archive file -> 압축파일 로드

* 자바, tomcat 버전이 개발환경과 다를 경우 import시 에러가 날 수 있습니다.

* import시 에러 발생시 처리 방법 (1, 2 수행)

1. 프로젝트 우클릭 -> properties클릭 -> project facet 검색(클릭) -> Runtimes 탭 클릭
-> 현재 설치되어있는 tomcat 체크
2. 프로젝트 우클릭 -> properties클릭 -> build path검색(클릭) -> Libraries탭 클릭 ->
-> JRE System Library항목 더블 클릭 -> alternate JRE 버전을 현재 자바버전으로 선택