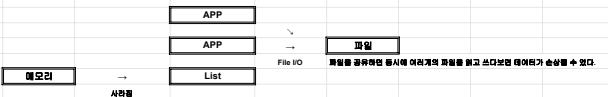
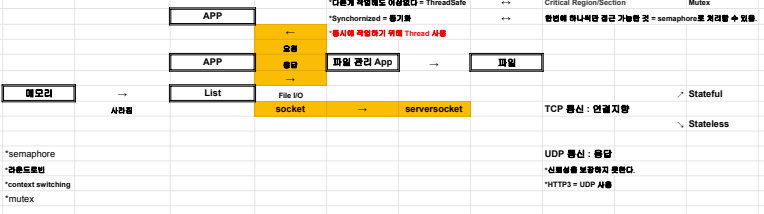


*파일 수정



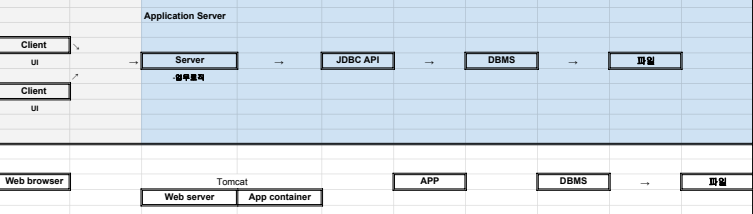
*파일 관리 App



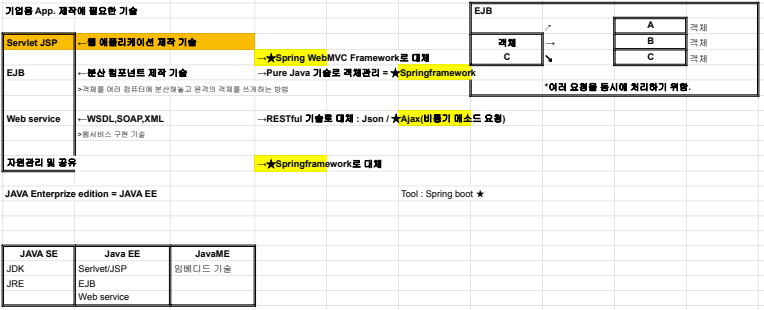
*파일 입출력에 요구조건 : DBMS



*보안 / 데이터 통제 체계 발생



*Servlet/JSP 기술 - 웹 애플리케이션 제작 자바 표준 기술

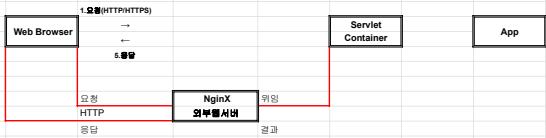


*JavaEE와 하위 기술을 버전

JavaEE-기술	Servlet	JSP	EJB
JavaEE5	2.5	2.1	3
JavaEE6	3	2.2	3.1
JavaEE7	3.1	2.3	3.2
JavaEE8	4	2.3	3.2

		<p>*JavaEE와 기술과 구현 서버(implements) ✓Java EE 기술 규격에 맞춰서 동작하도록 제작된 서버 JavaEE → JavaEE Implements(JavaEE 구현서버)</p> <p>WebLogic(Oracle) WebSphere(BM) JBoss(RedHat) JEUSt(TmaxSoft) Geronemo(Apache 재단)</p> <p>*JavaEE App 개발</p> <p>(기본 규격 및 라이브러리) JavaEE</p> <p>개발 → JavaEE 구현서버 ← 실행 ← JavaEE App ← 개발 ← 우리</p> <p>오라클개발자</p> <p>WebLogic(Oracle) → WebSphere(BM) → EJB JBoss(RedHat) → Servlet JEUSt(TmaxSoft) →</p> <p>JavaEE 기술 규격에 맞춰 제작된 App이라면 어떤 JAVAEE 서버에서도 실행할 수 있다. > Java EE 기술은 표준기술이기 때문이다.</p>
		<p>*JavaEE 버전과 App 개발</p> <p>JavaEE 1.3 WebLogic 6.1 → JavaEE 1.3 App WebLogic 7.0 → WebLogic 8.1 ↘ (X)</p> <p>JavaEE 1.4 WebLogic 9.0 → JavaEE 1.4 App JavaEE 5 WebLogic 10.0 → JavaEE 5 App WebLogic 11.x → JavaEE 6 App JavaEE 6 12cR1 → JavaEE 7 App JavaEE 7 WebLogic 12cR2 → JavaEE 8 App JavaEE 8 WebLogic 14c</p> <p>*상위버전 사용은 안됨 *서버가 어떤 버전으로 제작되어 있는지에 따라 다름.</p>
		<p>*Tomcat 서버와 JavaEE</p> <p>Servlet JSP JSF JSTL EJB WebService</p> <p>JavaEE 기술 중 Servlet/JSP 기술 규격만 지원하는 서버</p> <p>Tomcat 서버 Resin 서버 Jetty 서버</p> <p>web Container</p> <p>(자판)↗ Servlet (자판)→ str (자판)→ jst jsp+jstl ejb web service</p> <p>*웹 컨테이너 & 서버실행태어너 (자판x)→ (자판x)→</p>
		<p>*Tomcat 서버와 버전과 App 개발</p> <p>Servlet JSP JavaEE</p> <p>Tomcat 9 4 2.3 8 Tomcat 8 3.1 2.3 7 Tomcat 7 3 2.2 6 Tomcat6 2.5 2.1 5 Tomcat5 2.4 2 4 Tomcat4 2.3 1.2 3 Tomcat3 2.2 1.1 2</p>
		<p>*Web Application 개발 예</p> <p>Tomcat Server 9</p> <p>실행 → App ← 개발</p> <p>웹 브라우저의 지원 능력</p> <p>HTML5 CSS3 JavaScript</p>
		<p>Servlet/JSP = 웹 애플리케이션 제작 기술 > wrapping > Spring WebMVC Framework로 대체 EJB = 분산 컴포넌트 제작 기술 > 대체 → Pure Java 기술로 객체관리 = ★Springframework</p> <p>JavaEE 기술과 버전 = 어떤 버전이냐에 따라 호출 할 수 있는 메서드, 인터페이스가 달라짐 JavaEE(기술) → JavaEE Implements(JavaEE 구현서버)</p>
		<p>*Tomcat 서버의 핵심 부분</p> <p>1.요청(HTTP/HTTPS) Web Browser →</p> <p>2.처리 Mini Web server →</p> <p>3.실행 Servlet Container →</p> <p>4.결과 HTTP 요청처리 →</p> <p>5.출발 App ←</p> <p>Servlet/JSP 규격에 따라 동작</p> <p>Servlet 객체 (생성 → 실행 → 소멸) ↔ 관리 JSP 엔진 Servlet/JSP 규격에 따라 동작</p>

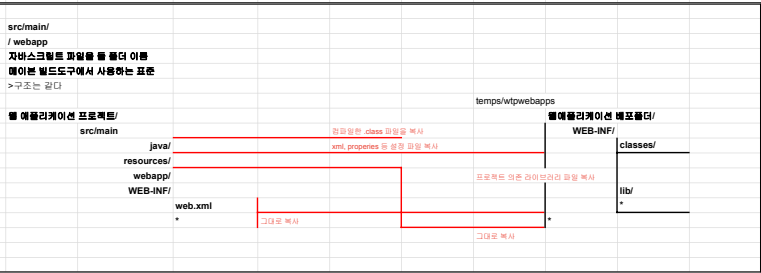
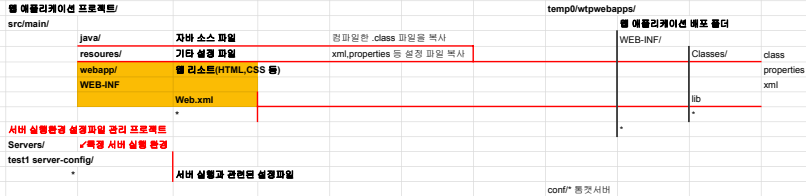
*현업에서 시스템을 구축할 때



*서버넷 컨테이너에 배포하는 웹 애플리케이션의 구조



*서버넷 컨테이너에 배포하는 웹 애플리케이션의 구조



*서버넷 컨테이너가 관리하고 실행하는 웹 애플리케이션 런타임(객체)



