

***서블릿 객체 생성과 Init()**

Tomcat server(서블릿 컨테이너) → 시작

1. 웹서버 관련 객체 생성 → 2. 웹 애플리케이션 관련 객체 생성

2-1) ServletContext 객체 생성
↓
2-2) 서블릿 객체 준비 -- 각각의 서블릿 객체에 대해 수행
✓ 단, loadOnStartup 설정이 되어 있을 경우

1.loadOnStartup 설정된 경우
(웹 애플리케이션이 시작될 때)
↓
2.요청이 들어 왔을 때
(생성되어 있지 않았을 때)

▼ ServletConfig 객체 생성
▼ 서블릿객체 생성 → 생성자 호출 (여기서 ServletConfig 객체 주입 권)
▼ Init() 호출 : ServletConfig 객체 주입

***서블릿/JSP가 사용할 수 있는 공식 보관소**

*** **

- ServletContext → 각 웹 앱 당 하나씩 생성된다.
DOA, Service, Connection
- HttpSession → 로그인 사용자 정보, 여러페이지를 걸쳐서 기억하는 데이터
- ServletRequest → 요청을 처리하는 동안 서블릿에서만 공유하는 데이터
- JSPContext → JSP를 실행하는 동안 사용할 곳

Interface ServletContextListener

Interface for receiving notification events about ServletContext lifecycle changes.

In order to receive these notification events, the implementation class must be either declared in the deployment descriptor of the web application, annotated with WebListener, or registered via one of the addListener methods defined on ServletContext.

Implementations of this interface are invoked at their contextInitialized(javax.servlet.ServletContextEvent) method in the order in which they have been declared, and at their contextDestroyed(javax.servlet.ServletContextEvent) method in reverse order.

ServletRequestListener

Interface for receiving notification events about requests coming into and going out of scope of a web application.

```

Tomcat 서버 call <interface> ServletContextListener ServletContext가 생성될때 소멸될 때
call <interface> ServletRequestListener ServletRequest가 생성될 때 (요청이 들어올 때) 소멸될 때(응답이 완료될 때)
call <interface> HttpSessionListener HttpSession이 생성될 때 소멸될 때
  
```

*ServletContextListener

```

<interface>
ServletContextListener
^
  
```


directive element	변환	
3 <%@ 태그 및 속성 %>	→	자바코드
declarattation element	복사	
4 <%!필드 및 메서드 선언%>	→	필드 및 메서드 선언
expression element	변환	
5 <%= 표현식(expression) % >	→	out.print(표현식);
JSP Action tag(JSP 전용태그)	변환	
6 <jsp:태그 속성 = " ... />	→	자바코드

JSP를 이용하여 유틸리티를 자동으로 생성하기 : MVC 모델1

*MVC 모델 1과 모델 2
Model View Controller

1.MVC 모델 1



1.MVC 모델 2



<%! %> = 매크로/태그
값을 리턴하는 문장 = expression

Model
View
Controller

Model
업무로직 수행
UI 생성
요청제어

