

Deployment Descriptor(DD) 설정

초기값 설정

ServletConfig

web.xml에 대한 정보를 가져오는 객체(interface)로 저장할 객체를 만들어 저장하고 `getServlet` 메소드로 정보가 저장된 객체 호출가능, `getInitParam("저장이름")`으로 초기화된 값을 가져올 수 있음

** 지정된 servlet에서만 활용 가능한 초기값임 / 동적 수정이 불가하여 상수값으로 보면 됨.

** servlet이 초기화된 이후에 활용가능

```
<web-app>
```

```
  <servlet>
```

```
    <servlet-name>mapping명칭</servlet-name>
```

```
    <servlet-class>설정 클래스명칭</servlet-class>
```

```
    <init-param>
```

```
      <param-name>저장 이름</param-name>
```

```
      <param-value>저장값</param-value>
```

```
    </init-param>
```

```
  </servlet>
```

```
</web-app>
```

ServletContext

ServletConfig의 초기값은 지정된 Servlet에서만 사용이 가능하나 ServletContext는 모든 어플리케이션이 공용으로 사용하는 초기값을 설정, 값은 `getServletContext().getInitParam("저장이름")`으로 호출

- ** <servlet> 태그 안이 아니라 <web-app> 내부에 설정
- ** getServletContext앞에는 getConfig() / this가 생략되어 있음.
- ** server에 대한 각종정보도 가지고 있는 객체임 (api 참조)

<web-app>

 <context-param>

 <param-name>저장 이름</param-name>

 <param-value>설정값</param-value>

 </context-param>

</web-app>

리스너 설정

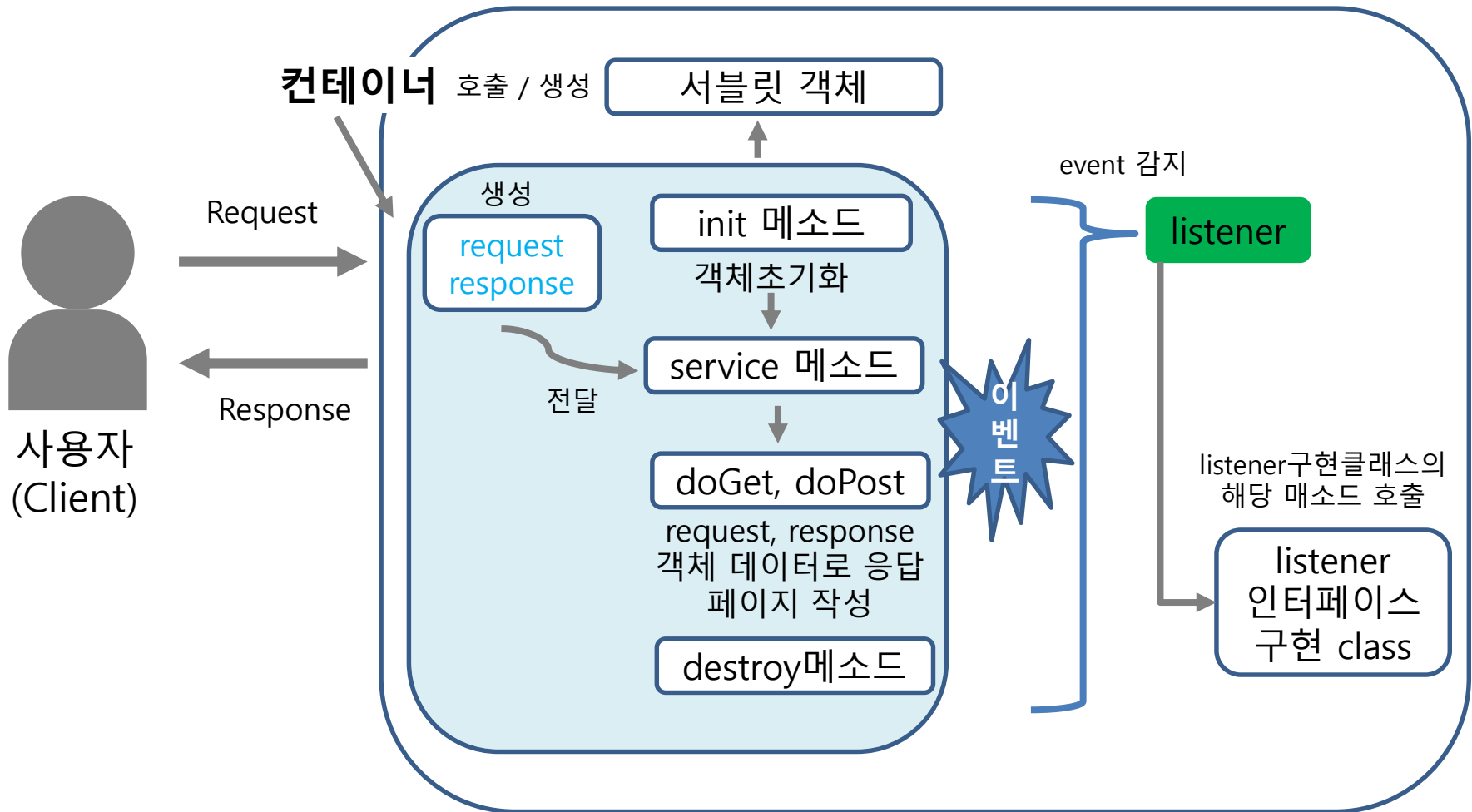
서블릿 리스너란?

서블릿 리스너란, 웹 컨테이너가 관리하는 라이프 사이클 사이에 발생하는 이벤트를 감지하여 해당 이벤트가 발생 시 해당 이벤트에 대한 일련의 로직을 처리하는 인터페이스를 말한다.

서블릿 이벤트 사용 용도

1. 웹 컨텍스트가 초기화되는 경우
2. 세션이 생기거나 소멸되는 경우, session바인딩,
3. 요청 정보의 속성이 바뀌는 경우
4. 속성값이 추가,삭제, 수정되는 경우(context, session, request)

서블릿 리스너 동작 구조



DD설정(web.xml)

```
<web-app>
```

```
  <listener>
```

```
    <listener-class>
```

listener를 구현하고 있는 클래스(패키지명까지)

```
    </listener-class>
```

```
  </listener>
```

```
</web-app>
```


클래스 설정(java코드)

```
public class 클래스명 implements 구현할 리스너인터페이스
{
    해당하는 인터페이스 메소드 오버라이딩 처리
}
```

예) public class TestContext implements ServletContextListener

```
{
    @Override
    public void contextInitalized(ServletContextEvent event)
    {
        ServletContext sc=event.getServletContext();
        sc.setAttribute("name", "자기이름");
    }
}
```

서블릿 이벤트 리스너 종류

1. **ServletContextListener / ServletContextAttributeListener**
 - 웹 애플리케이션의 ServletContext 생성과 종료 시 발생
 - ServletContext에 속성이 추가, 제거, 수정시 발생
2. **HttpSessionListener / HttpSessionAttributeListener**
 - 동시에 접속시 발생, 접속중이 세션정보 확인
 - 세션속성이 추가, 제거, 수정시 발생
3. **HttpSessionBindingListener**
 - HTTP 세션속성의 객체에 Binding되거나 해제되면 발생
4. **HttpSessionActivationListener**
 - HTTP 세션이 활성화 또는 비활성화가 될때 발생
5. **ServletRequestListener / ServletRequestAttributeListener**
 - 요청이 발생하면 발생
 - HttpServletRequest객체에 속성이 추가, 제거, 수정되면 발생

ServletContextListener

발생 시점	ServletContext가 생성, 소멸되었을때 발생 (어플리케이션 생성, 소멸시)
메소드 종류	contextInitialized(ServletContextEvent e) : void 웹 컨테이너가 처음 구동되어 ServletContext가 생성되었을때 작동하는 메소드
	contextDestroyed(ServletContextEvent e) : void 웹 컨테이너가 종료될 때 실행되는 메소드 ServletContext가 소멸 되었을 때 작동하는 메소드

ServletContextAttributeListener

발생 시점	ServletContext 객체에 속성이 추가, 삭제, 수정될 때 발생
메소드 종류	attributeAdded(ServletContextAttributeEvent e) : void 새로운 속성 값이 추가될 때 실행되는 메소드
	attributeRemoved(ServletContextAttributeEvent e) : void 속성 값이 제거될 때 실행되는 메소드
	attributeReplaced(ServletContextAttributeEvent e) : void 속성 값이 변경될 때 실행되는 메소드

HttpSessionListener

발생 시점	HttpSession 객체가 생성되거나 소멸될 때 발생
메소드 종류	<code>sessionCreated(HttpSession e) : void</code> 세션이 생성되었을 경우 실행되는 메소드
	<code>sessionDestroyed(HttpSession e) : void</code> 세션이 무효화 되었을 경우 실행되는 메소드

HttpSessionAttributeListener

발생 시점	HttpSession에 대한 속성 값이 변경될 경우 발생
메소드 종류	attributeAdded(HttpSessionBindingEvent e) : void 세션에 새로운 속성 값이 추가될 때 실행되는 메소드
	attributeRemoved(HttpSessionBindingEvent e) : void 세션에 속성 값이 제거될 때 실행되는 메소드
	attributeReplaced(HttpSessionBindingEvent e) : void 세션에 속성 값이 변경될 때 실행되는 메소드

HttpSessionActivationListener

발생 시점	HttpSession이 새로 생성되어 활성화 될 경우 발생
메소드 종류	sessionDidActivate(HttpSessionEvent e) : void 세션이 활성화 될 때 실행
	sessionWillPassivate(HttpSessionEvent e) : void 세션이 비활성화 되려고 할 때 실행

HttpSessionBindingListener

발생 시점	사용자의 현재 세션에 바인딩 되거나 해제될 객체가 발생할 경우 감지
메소드 종류	<code>valueBound(HttpSessionBindingEvent e) : void</code> 객체가 세션에 연결될 때 발생하는 이벤트를 실행
	<code>valueUnBound(HttpSessionBindingEvent e) : void</code> 객체가 세션으로부터 연결이 해제될 때 발생하는 이벤트를 실행

ServletRequestListener

발생 시점	request객체가 생성, 소멸되었을때 발생
메소드 종류	requestInitialized(ServletRequestEvent e) : void request객체가 생성되었을때 처리하는 메소드
	requestDestroyed(ServletRequestEvent e) : void request객체가 소멸 되었을때 처리하는 메소드

ServletRequestAttributeListener

발생 시점	request객체에 속성이 추가, 제거, 수정됐을때 발생
메소드 종류	attributeAdded(ServletRequestEvent e) : void 객체에 속성이 추가됐을때 호출되는 메소드
	attributeReplaced(ServletRequestEvent e) : void 객체의 속성이 수정되었을때 호출되는 메소드
	attributeRemoved(ServletRequestEvent e) : void 객체에서 속성이 삭제 되었을때 호출되는 메소드