Servlet에서 스프링 ApplicationContext 사용하기

pom.xml

<**dependency**>

<**groupId**>org.springframework</**groupId**>

<**artifactId**>spring-webmvc</**artifactId**>

<**version**>5.1.3.RELEASE</**version**>

</**dependency**>

스프링 ApplicationContext, 즉 스프링 IoC를 사용하려면 관련 의존성을 추가해야 한다.

spring-context 라이브러리를 가져올 수 있도록 의존성을 추가한다.

위 예제에서는 spring-webmvc를 가져오도록 했지만 spring-context만 가져와도 된다.

web.xml

<?xml version="1.0" encoding="UTF-8"?>

<**web-app** xmlns="http://xmlns.jcp.org/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_4\_0.xsd"

version="4.0">

<**listener**>

<**listener-class**>org.springframework.web.context.ContextLoaderListener</**listener-class**>

</**listener**>

<**context-param**>

<**param-name**>contextClass</**param-name**>

<**param-value**>org.springframework.web.context.support.AnnotationConfigWebApplicationContext</**param-value**>

</**context-param**>

<**context-param**>

<**param-name**>contextConfigLocation</**param-name**>

<**param-value**>com.atoz\_develop.AppConfig</**param-value**>

</**context-param**>

...

스프링이 제공하는 리스너인 ContextLoaderListener를 사용해서 ApplicationContext를 만들어 ServletContext에 저장할 수 있다.

방법은 web.xml에 위와 같이 설정하는 것이다.

<context-param>에 contextClass 파라미터에는 ApplicationContext의 타입을, contextConfigLocation 파라미터에는 빈 설정파일을 설정한다.

contextClass, contextConfigLocation 파라미터명은 저 이름으로 ContextLoader가 사용하기 때문에 반드시 지켜야 한다.

서블릿

public class HelloServlet extends HttpServlet {

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

ApplicationContext context = (ApplicationContext) getServletContext().getAttribute(WebApplicationContext.ROOT\_WEB\_APPLICATION\_CONTEXT\_ATTRIBUTE);

}

}

다음으로 서블릿에서 ServletRequest를 통해 ServletContext를 가져오고, 가져온 ServletContext에서 attribute name 'WebApplicationContext.ROOT\_WEB\_APPLICATION\_CONTEXT\_ATTRIBUTE'을 사용해서 ContextLoaderListener를 통해 만들어 저장한 ApplicationContext를 가져와 사용할 수 있다.