**2. AOP(Aspect Oriented Programming)**

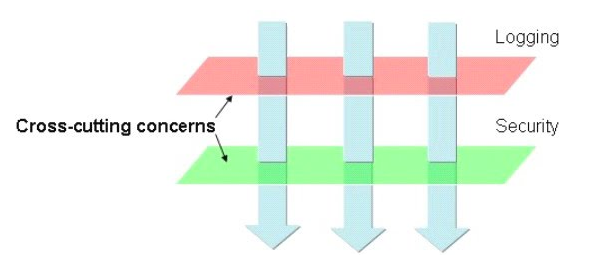
1. **AOP란**

AOP는 Aspect Oriented Programming의 약자로 관점 지향 프로그래밍으로 해석되며 관점 지향은 쉽게 말해

어떤 로직을 기준으로 핵심적인 관점, 부가적인 관점으로 나누어서 보고 그 관점을 기준으로 각각 ‘모듈화’하는

프로그래밍 기법인 것이다. 여기서 ‘모듈화’란 어떤 공통된 로직이나 기능을 하나의 단위로 묶는 것을 말한다.

AOP방법은 핵심 기능과 공통 기능을 분리 시켜놓고, 공통 기능을 필요로 하는 핵심 기능들에서 사용하는 방식 입니다.

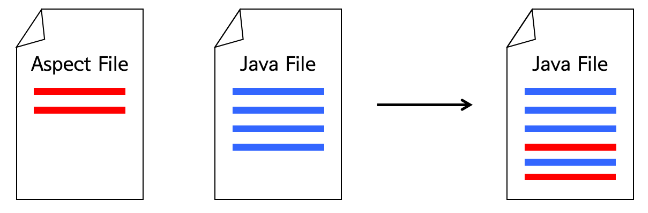


핵심 관점(Core Concerns / ex)업무 로직) 로직의 특정 실행 시점에 부가적인 관점(**Crosscutting Concerns** /

ex) 트랜잭션 / 로그 / 보안 / 인증 처리등)의 로직이 필요에 따라 실행된다.

DI는 낮은 결합도를 위한 것

AOP는 높은 응집도를 위한 것



1. **AOP의 주요 개념**
2. Aspect : 공통 기능. 예를 들어 로깅같은 기능 자체에 대한 용어
3. Advice : 공통기능을 구현한 객체 메
4. Join Point : 핵심기능. 공통 기능을 적용할 수 있는 대상.

PointCuts : Join Point의 부분으로 실제 Advice를 적용해야 되는 부분

* Proxy : Advice가 적용되었을 때 만들어지는 객체

Weaving : Advice와 target이 결합되어서 프록시 객체를 만드는 과정(advice를 핵심기능에 적용하는

AOP의 구현은 proxy를 이용한다.

Weaving 방식은 두가지

1. XML을 이용

②@Aspect 어노테이션 이용

포인트컷에는 다양한 명시자를 이용할 수 있습니다.

execution Advice를 적용할 메서드를 명시할 때 사용합니다. == 메소드 명시

within 특정 타입에 속하는 메서드를 JoinPoint로 설정되도록 명시할 때 사용합니다. ==클래스 명시

bean 스프링 버전 2.5 버전부터 지원하기 시작했으며, 스프링 빈을 이용하여 JoinPoint를 설정합니다.

//bean 명시자 위치 == 패키지 내 모든 해당 bean 객체 xml만 ? jsp도?

// execution과 within 둘다 클래스명시? 그런데 왜 명시자 작성이 다름

== 명시 범위 다름 하나는 클래스 하나는 매서드

스프링 mvc

스프링의 기본 흐름과 주요 컴포넌트

스프링 작동 과정

(1) 웹브라우저를 통해 클라이언트의 요청이 들어오면 무조건 맨 처음 DisptcherServlet에서 요청을 받는다.

(2) 요청된 내용을 HandlerMapping에게 작업을 보내 요청 URL과 매칭되는 컨트롤러를 검색해 준다.

(3) 요청된 내용의 처리를 컨트롤러에게 요청한다

(4) 요청된 내용을 Controller에게도 보내고 요청된 내용에 따라 다른 일을 하고, 컨트롤러 샐행결과를 ModelAndView로 변환해서 리턴한다

(5) 컨트롤러의 실행결과를 보여줄 View를 검색하기 위해 ViewResolver에게도 보낸다.

(6) 요청한 결과의 View로 응답 생성을 한다.

우리가 작업을 많이 할 부분은 Controller, View, Model이고 나머지 부분은 Spring에서 거의 자동으로 다루어준다.

요청이 들어오면 DispatcherServlet 이 먼저 받고, Controller에게 요청을 한다. Controller에서 다시 DispatcherServlet으로 뭔가 응답을 준다. 그러면 DispatcherServlet에서 ViewResolver를 통해 View에 호출이 된다. 그러면 사용자에게 응답이 이루어 진다.

웹 빈설정 가능 xml

<param-value>/WEB-INF/spring/root-context.xml</param-value> //db관련 bean

<param-value>/WEB-INF/spring/appServlet/servlet-context.xml</param-value> // 웹 관련 bean

<context:component-scan base-package="com.lec.ch06" />의 어노테이션 해석?

1. Web.xml스캔
2. /요청을 스캔함

<servlet-mapping>

<servlet-name>appServlet</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

새로운 요청이 있을떄까지 계속 유효한 model값???

Addattribute로 추가

Servlet-context.xml에

<resources mapping="/css/\*\*" location="/css/" />

Href는 요청?? dispatcher요청으로 처리 안하게

Modelandview 를 자동으로 만들어줌

<context:component-scan base-package="com.lec.ch06" />는 해당 패키지의 모든 클래ㅐ스를 스캔하기 때문에

로직시간 줄이기 위해 컨트롤러만 명시할 수 도있다.

<context:component-scan base-package="com.lec.ch06.controller" />

model

////Method로 요청 구분시 여러 개면 ?????