

Spring Professional Exam Tutorial v5.0

Question 06

Question 06 - What do you have to do to enable the detection of the `@Aspect` annotation? What does `@EnableAspectJAutoProxy` do?

To enable detection of `@Aspect` annotation you need to:

- ▶ Have `@Configuration` class with `@EnableAspectJAutoProxy`
 - ▶ Without `@EnableAspectJAutoProxy` Spring will not scan for `@Aspect`
- ▶ Have beans for `@Aspect` annotated classes created
 - ▶ Use `@ComponentScan` with `@Component` at class annotated with `@Aspect`
 - ▶ Use `@Bean` in Configuration class and create Spring Aspect Bean manually
- ▶ Have `aspectjweaver/spring-aop` on classpath
 - ▶ It is easiest to use `org.springframework:spring-aspects` dependency to have those included
 - ▶ Without required dependencies on classpath spring will fail with `ClassNotFoundException/NoClassDefFoundError` during creation of Proxy objects for Spring Beans subject to aspects

Question 06 - What do you have to do to enable the detection of the `@Aspect` annotation? What does `@EnableAspectJAutoProxy` do?

Annotation `@EnableAspectJAutoProxy` enables detection of `@Aspect` classes and creates proxy object for beans subject to aspects. Internally process of creating proxies is done by `AnnotationAwareAspectJAutoProxyCreator`. By creating a proxy for each bean subject to aspects, spring intercepts the calls and implements `Before` / `After` / `AfterReturning` / `AfterThrowing` / `Around` advices. It is important to remember that `@Aspect` will not create Spring Beans on it's own, you need to use Component Scanning or manually create beans for `@Aspect` classes.

