Spring主要是IOC和AOPSpringIOC平时用来管理对象，像以前的很经典的MVC模式，MVC三层，各层的耦合度都很高，通过new每一层对象才能调用。而我们使用ioc的话能够对MVC三层进行解耦，降低耦合度。具体做法的话一般是在Spring的配置文件中取一个bean标签，并在bean标签里设置class属性，这样来让对象加入到ioc容器中。也要加一个ID属性，方便对对象的取用。我们也可以用注解配置的方式，自定义一个类，并在类上添加@Configuration注解，在类的方法上加上@Bean注解，方法里返回这个对象，方法名是默认的id属性，也可以在@Bean注解里配置id属性。我们要在其他地方使用这些对象的时候，只需在变量上添加@Autowired或者@Resource，@Autowired按类型来取的，@Resource是按id名取。这个ioc的底层的话是用map结构来区分不同Bean的。接着SpringAOP，首先想到的是OOP编程思想，OOP是自上而下的编程思想，而AOP是一个横切性的编程思想，SpringAOP借助AspectJ的语法实现的编程思想。配置SpringAOP需要在Spring的配置文件中配上切入面，增强类型以及相应的切入点。当然也可以采用注解的方式配置，自定义一个类，在类名上加入@Aspect，在类里面起方法，方法里面的内容就是相应的你要植入到目标类的逻辑代码，并在方法上加上增强类型的注解，比如@Before、@After。用@Pointcut指定这个目标类的哪个方法执行。平时使用比较多的场景，监控日志、权限管理、事务控制。AOP底层使用JDK的动态代理和cglib动态代理。在@Aspect