

Spring 注解

要求

- 掌握 Spring 常见注解

提示

- 注解的详细列表请参考：面试题-spring-注解.xmind
- 下面列出了视频中重点提及的注解，考虑到大部分注解同学们已经比较熟悉了，仅对个别的作简要说明

事务注解

- @EnableTransactionManagement，会额外加载 4 个 bean
 - BeanFactoryTransactionAttributeSourceAdvisor 事务切面类
 - TransactionAttributeSource 用来解析事务属性
 - TransactionInterceptor 事务拦截器
 - TransactionalEventListenerFactory 事务监听器工厂
- @Transactional

核心

- @Order

切面

- @EnableAspectJAutoProxy
 - 会加载 AnnotationAwareAspectJAutoProxyCreator，它是一个 bean 后处理器，用来创建代理
 - 如果没有配置 @EnableAspectJAutoProxy，又需要用到代理（如事务）则会使用 InfrastructureAdvisorAutoProxyCreator 这个 bean 后处理器

组件扫描与配置类

- @Component
- @Controller
- @Service
- @Repository
- @ComponentScan
- @Conditional
- @Configuration
 - 配置类其实相当于一个工厂，标注 @Bean 注解的方法相当于工厂方法
 - @Bean 不支持方法重载，如果有多个重载方法，仅有一个能入选为工厂方法
 - @Configuration 默认会为标注的类生成代理，其目的是保证 @Bean 方法相互调用时，仍然能保证其单例特性
 - @Configuration 中如果含有 BeanFactory 后处理器，则实例工厂方法会导致 MyConfig 提前创建，造成其依赖注入失败，解决方法是改用静态工厂方法或直接为 @Bean 的方法参数依赖注入，针对 Mapper 扫描可以改用注解方式

- @Bean
- @Import
 - 四种用法
 - ① 引入单个 bean
 - ② 引入一个配置类
 - ③ 通过 Selector 引入多个类
 - ④ 通过 beanDefinition 注册器
 - 解析规则
 - 同一配置类中, @Import 先解析 @Bean 后解析
 - 同名定义, 默认后面解析的会覆盖前面解析的
 - 不允许覆盖的情况下, 如何能够让 MyConfig(主配置类) 的配置优先? (虽然覆盖方式能解决)
 - 采用 DeferredImportSelector, 因为它最后工作, 可以简单认为先解析 @Bean, 再 Import
- @Lazy
 - 加在类上, 表示此类延迟实例化、初始化
 - 加在方法参数上, 此参数会以代理方式注入
- @PropertySource

依赖注入

- @Autowired
- @Qualifier
- @Value

mvc mapping

- @RequestMapping, 可以派生多个注解如 @GetMapping 等

mvc rest

- @RequestBody
- @ResponseBody, 组合 @Controller => @RestController
- @ResponseStatus

mvc 统一处理

- @ControllerAdvice, 组合 @ResponseBody => @RestControllerAdvice
- @ExceptionHandler

mvc 参数

- @PathVariable

mvc ajax

- @CrossOrigin

boot auto

- @SpringBootApplication
- @EnableAutoConfiguration
- @SpringBootConfiguration

boot condition

- @ConditionalOnClass, classpath 下存在某个 class 时, 条件才成立
- @ConditionalOnMissingBean, beanFactory 内不存在某个 bean 时, 条件才成立
- @ConditionalOnProperty, 配置文件中存在某个 property (键、值) 时, 条件才成立

boot properties

- @ConfigurationProperties, 会将当前 bean 的属性与配置文件中的键值进行绑定
- @EnableConfigurationProperties, 会添加两个较为重要的 bean
 - ConfigurationPropertiesBindingPostProcessor, bean 后处理器, 在 bean 初始化前调用下面的 binder
 - ConfigurationPropertiesBinder, 真正执行绑定操作