**Bean 的 Scope**

Singleton,prototype, request, session @Scope("Singleton)

**spring EL 和资源调用**

@PropertySource（value=“t.properties"

,encoding=“UTF-8”）

@value（“#{demoService.author

}")

**bean的初始化和销毁**

@postconstruct，@predestroy

@bean（initMethod=“init”，destroyMethod=“destroy”）

**profile的问题**

@profile（“dev”），@profile（“prod”）

env！ context.getEnvironment().setActiveProfiles("Prod");

Jvm! Set java\_opts="-Dspring.profiles.active=test"

Web!

<init-param> <paramname>Spring.profiles.active

<param-name>

<param-value>prod</param-value>

<init-param>

注解！

@activeprofiles（"Dev"）

**Spring中的事件传递**

applicationevent，applicationlistener

**spring aware**

beanNameAware可以帮助我们获得Bean的名称

beanFactoryAware可以帮助我们获取当前Bean Factory

ResourceLoaderAware可以帮助我们获取资源加载器，可以帮助我们获取外部资源文件

**spring 多线程**

@EnableAsync 开启异步任务支持

threadpooltaskexecutor 线程池

@async 该方法是异步的

**spring 计划任务**

**@**scheduled.

@enablescheduling.开启对计划任务的支持

容器加载该类，时间方法就开始运行了

**spring 条件注解@conditional**

该注解用在config类中，多个同时用，满足哪个条件，返回哪个的结果

**spring 组合注解与元注解**

元注解是一种使用在别的注解上的注解，被注解的注解称为组合注解。

**spring中的测试问题**

@runwith（springJunit4classrunner.class）

@contextconfiguration（class=配置类）

@test