

# 第八章：Spring Bean 作用域

小马哥 · mercyblitz



扫码试看/订阅

《小马哥讲 Spring 核心编程思想》视频课程

# Spring Bean 作用域

---

1. Spring Bean 作用域
2. "singleton" Bean 作用域
3. "prototype" Bean 作用域
4. "request" Bean 作用域
5. "session" Bean 作用域
6. "application" Bean 作用域
7. 自定义 Bean 作用域
8. 课外资料
9. 面试题精选



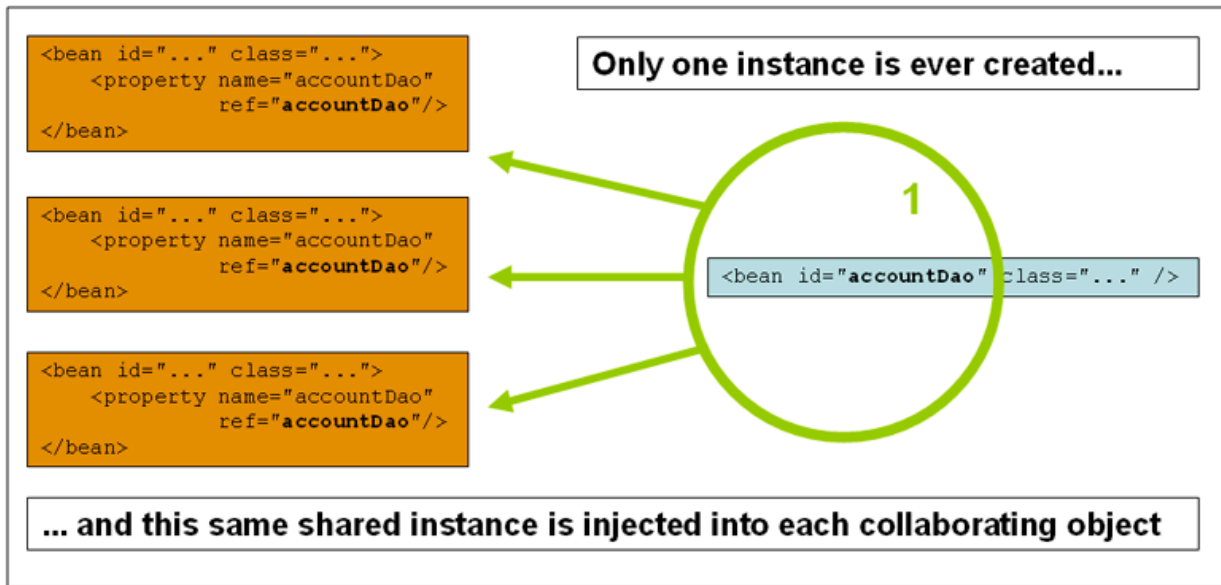
# Spring Bean 作用域

- 作用域

来源	说明
singleton	默认 Spring Bean 作用域， <u>一个 BeanFactory</u> 有且仅有一个实例 注意不是进程唯一
prototype	原型作用域，每次依赖查找和依赖注入生成新 Bean 对象
request	将 Spring Bean 存储在 ServletRequest 上下文中
session	将 Spring Bean 存储在 HttpSession 中 后三种用于视图渲染
application	将 Spring Bean 存储在 ServletContext 中

## "singleton" Bean 作用域

- 配置

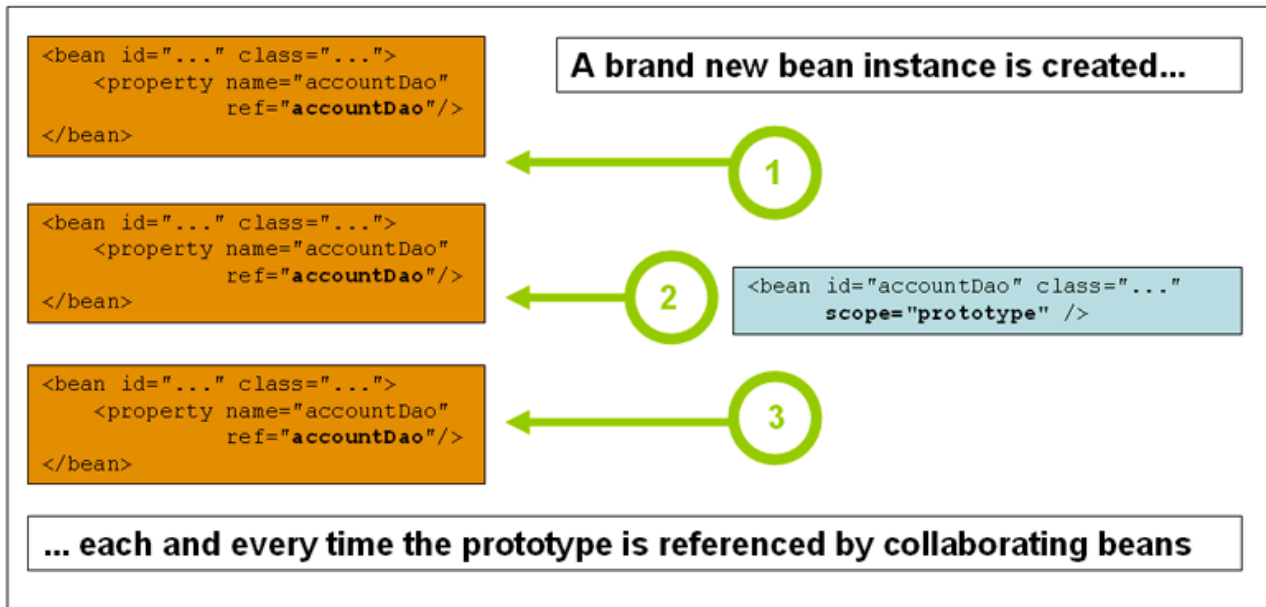


jvm 中静态变量对应一个 ClassLoader(保证唯一)

spring 中单例对应一个 BeanFactory(保证唯一)

## "prototype" Bean 作用域

- 配置



## "prototype" Bean 作用域

- 注意事项
  - Spring 容器没有办法管理 prototype Bean 的完整生命周期，也没有办法记录示例的存在。销毁回调方法将不会执行，可以利用 BeanPostProcessor 进行清扫工作。  
但具有很大的局限性

# "request" Bean 作用域

- 配置
  - XML - `<bean class= "..." scope = "request" />`
  - Java 注解 - `@RequestScope` 或 `@Scope(WebApplicationContext.SCOPE_REQUEST)`
- 实现
  - API - `RequestScope`



# "session" Bean 作用域

- 配置
  - XML - `<bean class= "..." scope = "session" />`
  - Java 注解 - `@RequestScope` 或 `@Scope(WebApplicationContext.SCOPE_REQUEST)`
- 实现
  - API - `SessionScope`

# "application" Bean 作用域

- 配置
  - XML - `<bean class= "..." scope = "application" />`
  - Java 注解 - `@ApplicationScope` 或 `@Scope(WebApplicationContext.SCOPE_APPLICATION)`
- 实现
  - API - `ApplicationScope`

# 自定义 Bean 作用域

- 实现 Scope

- org.springframework.beans.factory.config.Scope

- 注册 Scope

- API - org.springframework.beans.factory.config.ConfigurableBeanFactory#registerScope

- 配置

```
<bean class="org.springframework.beans.factory.config.CustomScopeConfigurer">
  <property name="scopes">
    <map>
      <entry key="...">
      </entry>
    </map>
  </property>
</bean>
```

## 课外资料

- Spring Cloud RefreshScope 是如何控制 Bean 的动态刷新?

## 面试题

**沙雕面试题** - Spring 内建的 Bean 作用域有几种？

答：singleton、prototype、request、session、application 以及 websocket



我真的没笑

## 面试题

996 面试题 - singleton Bean 是否在一个应用是唯一的?



答：否，singleton bean 仅在当前 Spring IoC 容器（BeanFactory）中是单例对象。

## 面试题

**劝退面试题** - “application” Bean 是否被其他方案替代



答：可以的，实际上，“application” Bean 与 “singleton” Bean 没有本质区别



扫码试看/订阅

《小马哥讲 Spring 核心编程思想》视频课程