

加载bean顺序

singletonObjects
(一级)

从容器里面获取的bean就是从这里面拿的

A (初始化完成)

B (已经初始化完成, 执行完initializeBean ())

earlySingletonObjects
(二级缓存)

A (在注入B的时候, A的状态还是跟上一部一样)

B依赖的A在开始创建的时候已经在缓存中了, 所以直接可以注入给B, 就完成了Bean的初始化, 直接从三级缓存中移除, 添加到一级缓存中

从三级缓存中移除

创建B的过程中, 发现B又依赖于A并且A在三级缓存中, 所以就直接将A从三级缓存中拿出来, 然后放到二级缓存中

singletonFactories
(三级缓存)

A (半成品, 没有填充属性和初始化)

populateBean

B(半成品,没填充属性和初始化)

出现在三级缓存中的bean的状态是, 调用构造方法将bean创建出来, 在注入A属性注入的时候, 发现A依赖B, 所以就去缓存里面找B, 发现没找到, 然后就去三级缓存中找B, 发现找到了, 然后就去三级缓存中移除B, 添加到二级缓存中