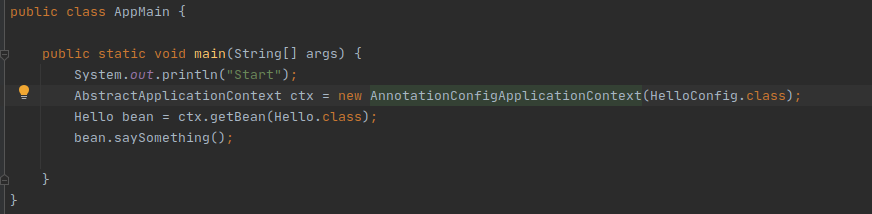
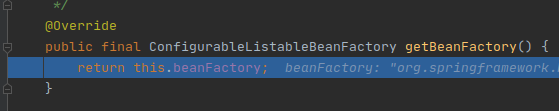
AbstractApplicationConext.getBean(Class<T>)

getBeanFactory().getBean()

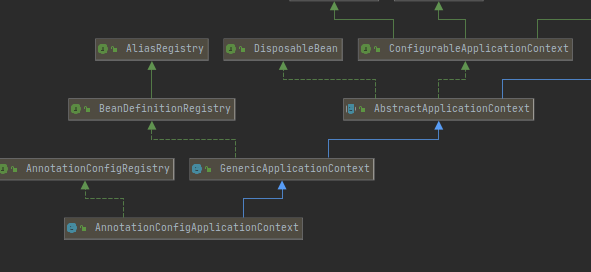
下图中，ctx对应的BeanFactory是如何创建出来的？



Debug, 看到getBeanFactory实际上调用的是GenericApplicationContext.



AnnotationConfigApplicationContext的父类是GenericApplicationContext.静态代码的时候，你就应该看到出BeanFactory源自哪里。源自GenericApplicationConext的成员变量。



下面就看这个成员变量是哪里来的？如何构造出来的。

new AnnotationConfigApplicationContext(HelloConfig.class);这个构造函数，其实调用了AnnotationConfigApplicationContext的默认构造函数，默认构造函数，虽然没明确写，肯定是调用了父类的默认构造函数，而父类的构造函数中就直接创建了DefaultListableBeanFactory。从下图的stack中可以证明这一点：

