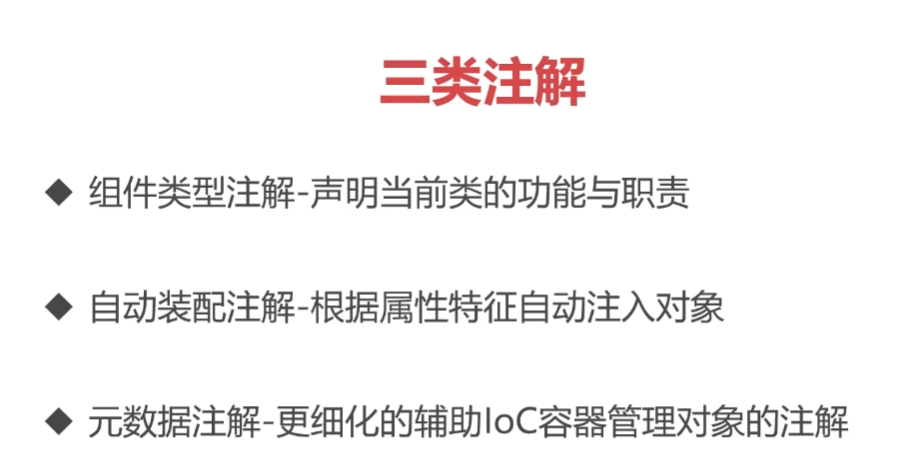
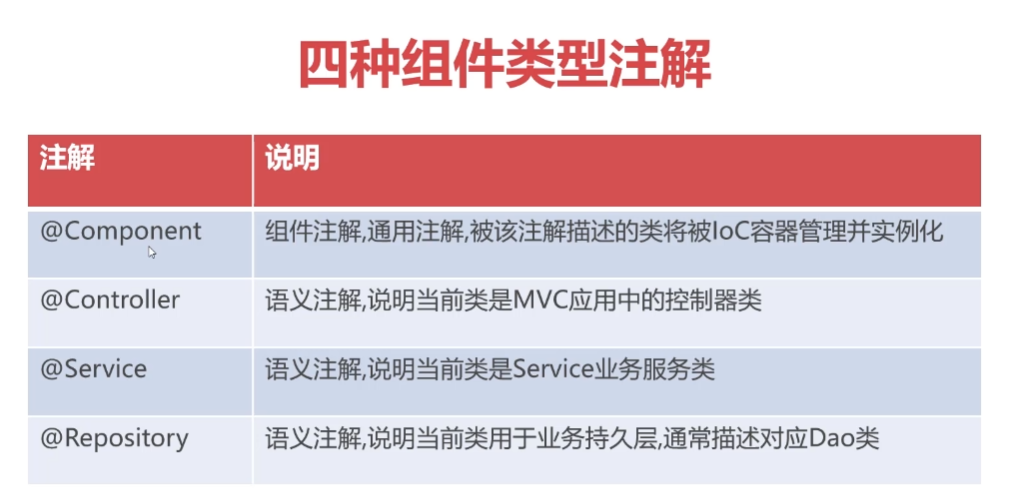
使用註解方式實現SpringIoC

**一、三類註解**



1.第一類:組件類型註解



為了要讓註解生效，要開啟組件掃描，透過在XML中配置。



base-package:代表要掃描的資料夾，會一層層往下掃描資料夾內的所有資料。

exclude-filter type代表:要排除的檔案，透過正規表達式排除檔案。

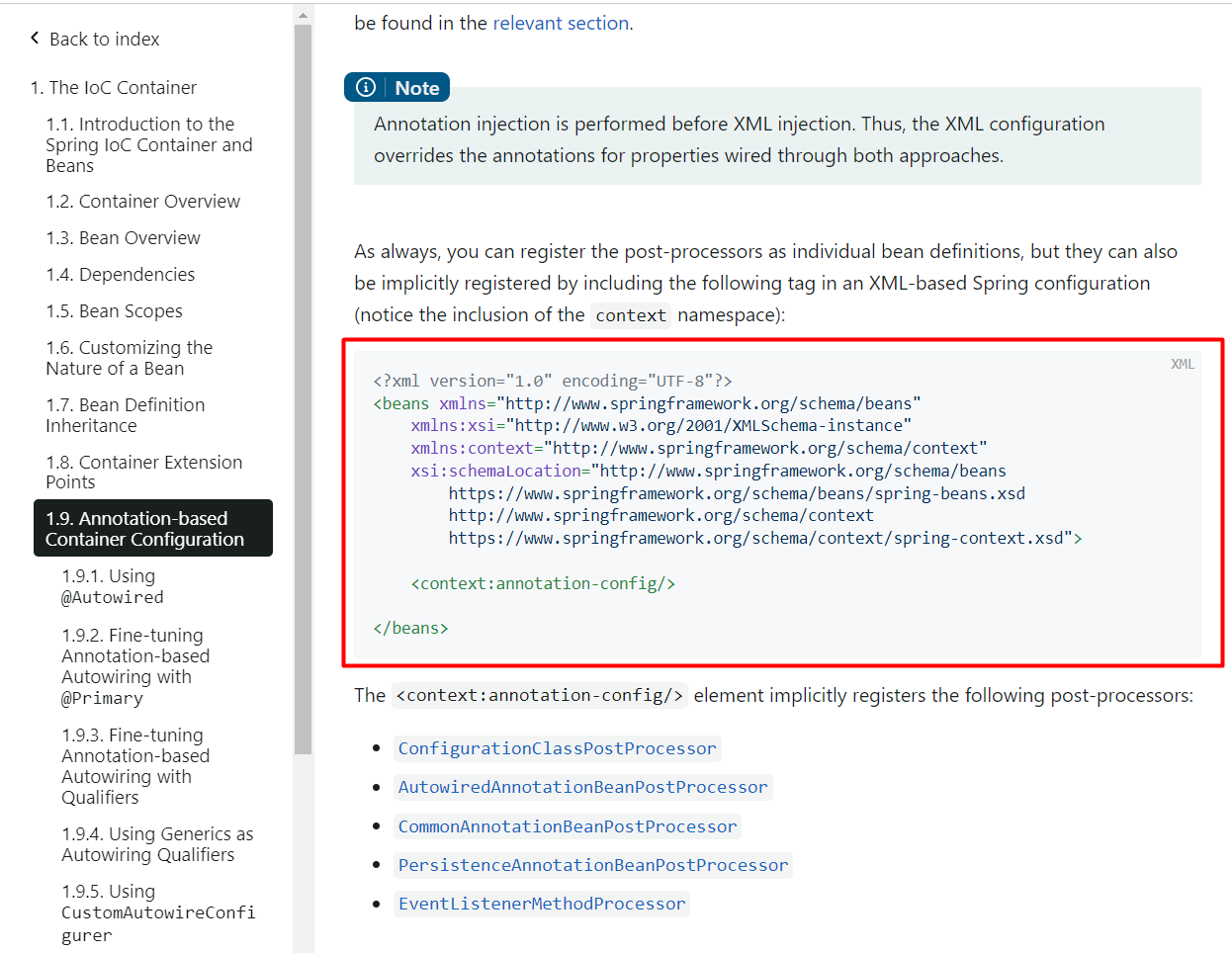
**二、使用註解開發方式(程式碼Spring\_07)**

1.在Maven中引用Spring依賴

2.創建applicationContext.xml

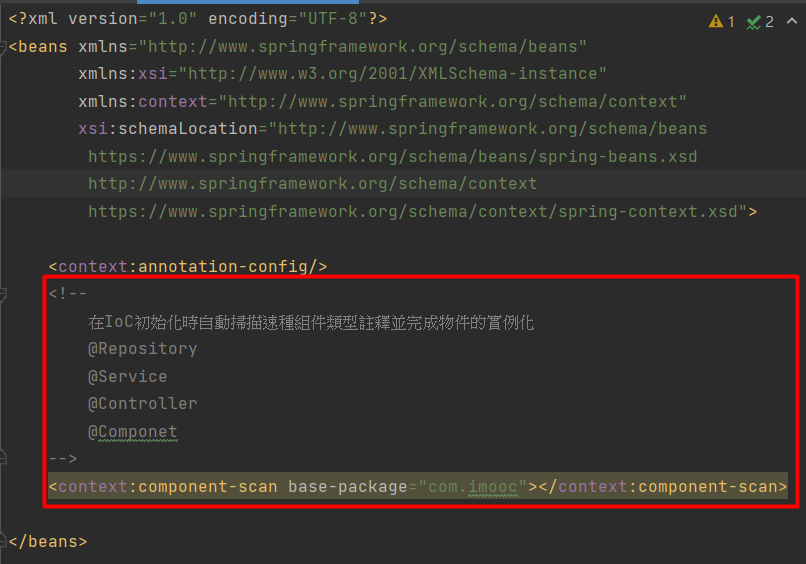
3.填入註解開發方式的DTD

DTD位置: <https://docs.spring.io/spring-framework/docs/current/reference/html/core.html#beans-annotation-config>

找到Annotation-based Container Configuration的xml header

值得注意的是context其實是一個命名空間，如果要使用context的子標籤則語法為context:子標籤

4.開啟組件掃描功能



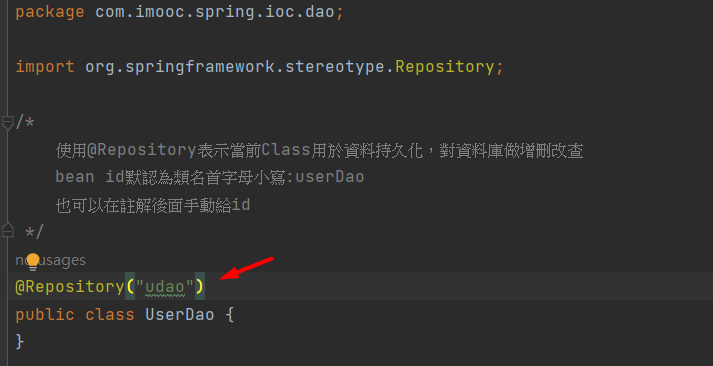
使用context:component-scan標籤，可以開啟註解型式的設計方法，IoC容器會在初始化的時候自動掃描com.imooc資料夾下的所有Class ，並完成物件的創建。

**三、規則**

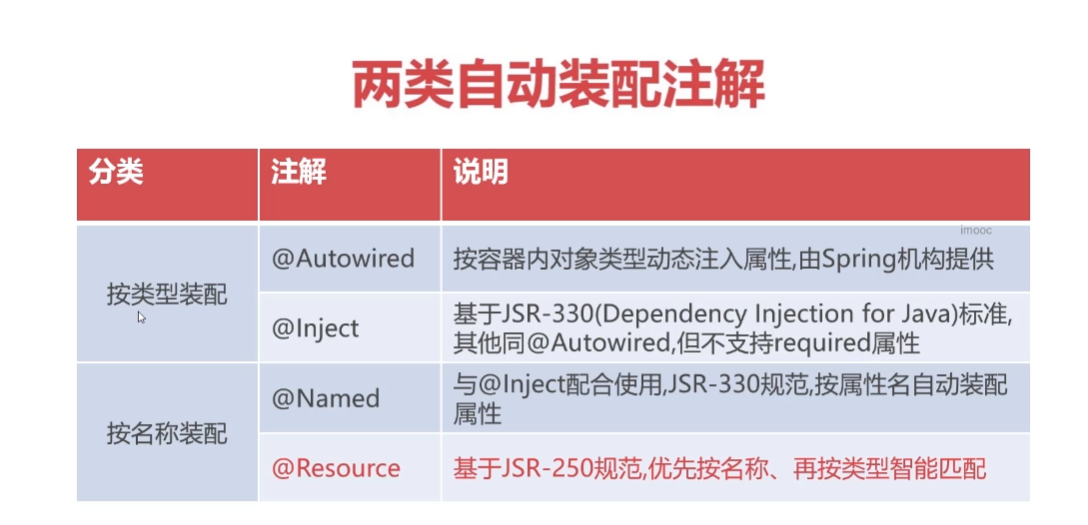
(一)bean id

使用註解開發時，bean的id默認會是類名首字母小寫，

也可以手動在註解後面設定id。



**四、自動裝配註解**

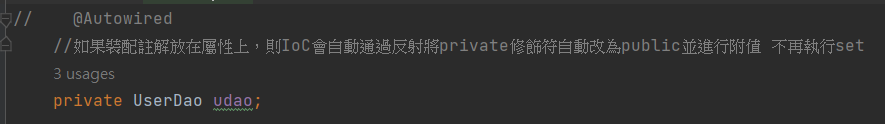


(一)依照類型自動裝配(動態注入)

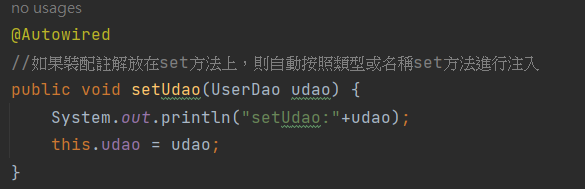
1.Autowired的用法差異

在使用Autowired時，放在屬性上與setter上，執行原理不一樣。

(1)放在屬性上:通常使用這種方法，這樣就不用寫setter方法。



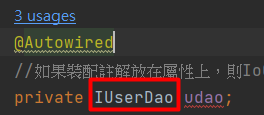
(2)放在setter上面。



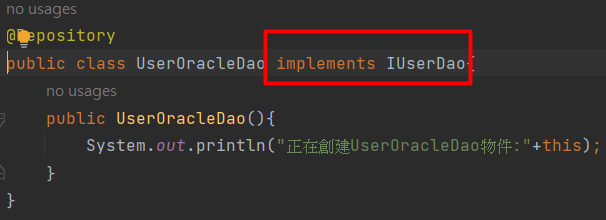
[依照類型動態裝配的問題]

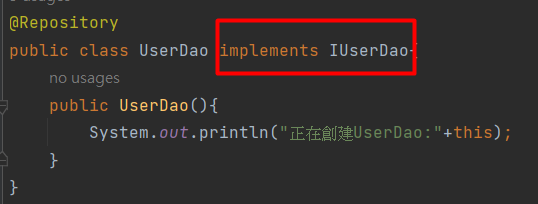
但如果屬性為一個接口，且有多個類實現該接口，則Autowired會不知道要動態注入哪一個類。

UserService中的屬性為一個IUserDao接口



有一個UserOracleDao實作了IUserDao

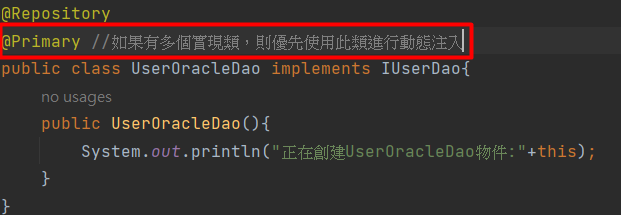


同時，UserDao也實作了IUserDao

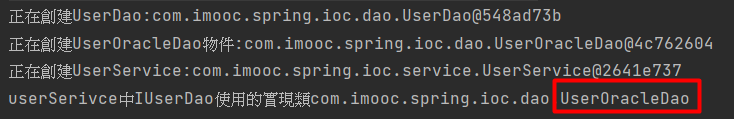
Spring IoC容器無法決定要將userDao或userOracleDao這兩個實現類，注入UserService類中的IUserDao屬性。



因此需要使用@Primary用來告訴Spring IoC容器，如果發生多個物件都有實作接口，則優先使用@Primary類。



最後就會採用使用@Primary的類，自動裝配。



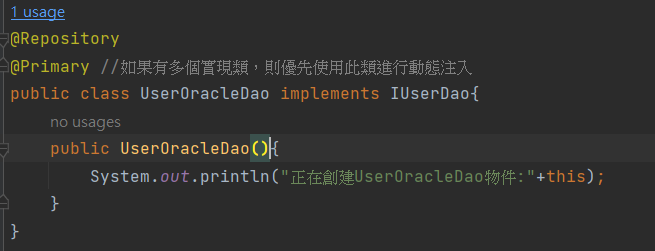
(二)依照名稱自動裝配(業界常用的方法)

使用@Resource可以優先依照指定的名稱進行動態注入，並且在沒有找到指定名稱的情況下，智能的依照類型匹配進行注入。這樣就可以避免使用類型自動裝配時，產生的類型衝突。



使用方法1:使用name來指定要綁定的物件bean id

由於UserOracleDao在@Repository中沒有手動添加id，因此id會默認為userOracleDao。



最終結果就是Resource成功匹配到bean id為userOracleDao類，自動綁定到IUserDao屬性中。

使用方法2:使用屬性名稱來指定要綁定的物件bean id



此時，@Resource就會使用userOracleDao這個屬性名稱進行bean id的匹配。

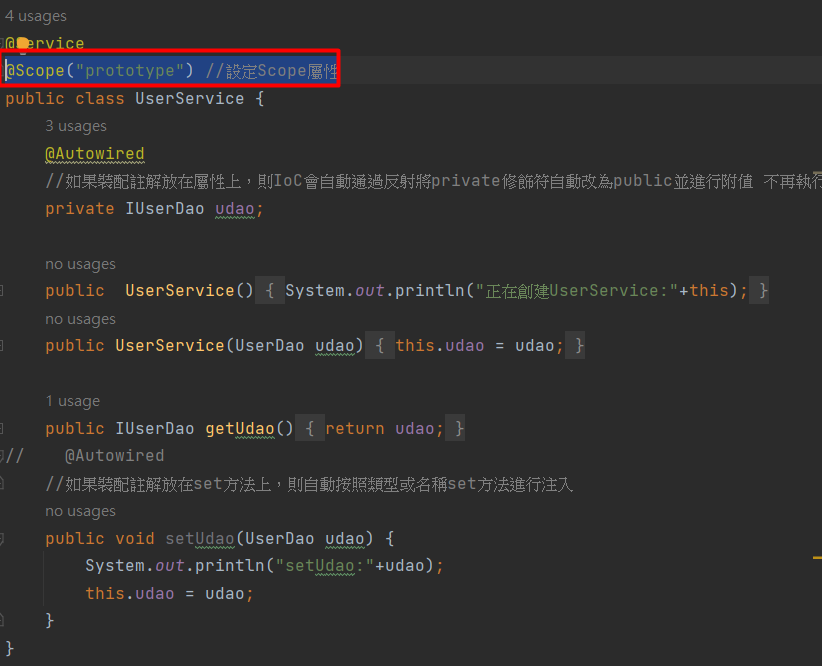
**四、元數據註解**



簡單來說就是XML中標籤能設置的屬性，這些語法的註解版本。

(一)使用方法:

1.Scope



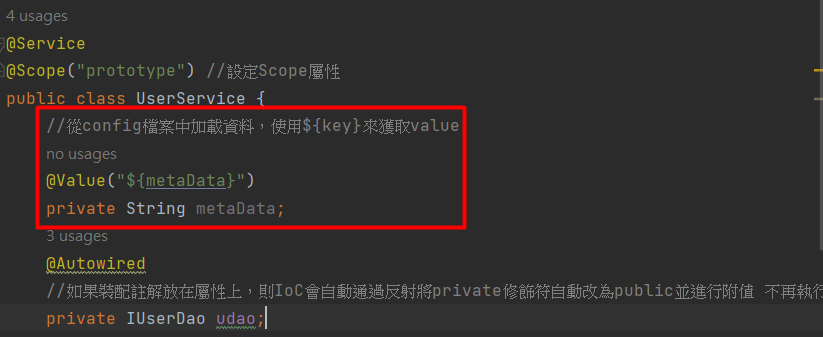
[2.PostConstruct](mailto:2.@PostConstruct)



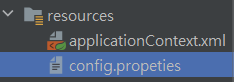
3.Value

使用步驟

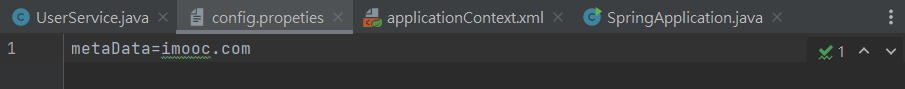
(1)先加上Value註解



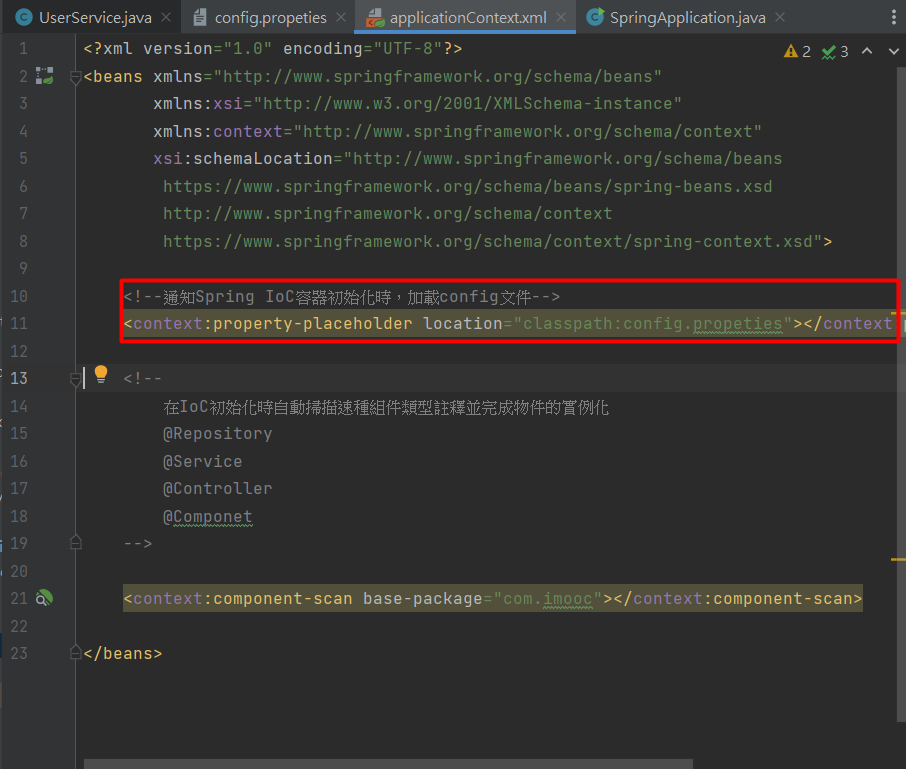
(2)在Resource資料夾中創建config.properties文件



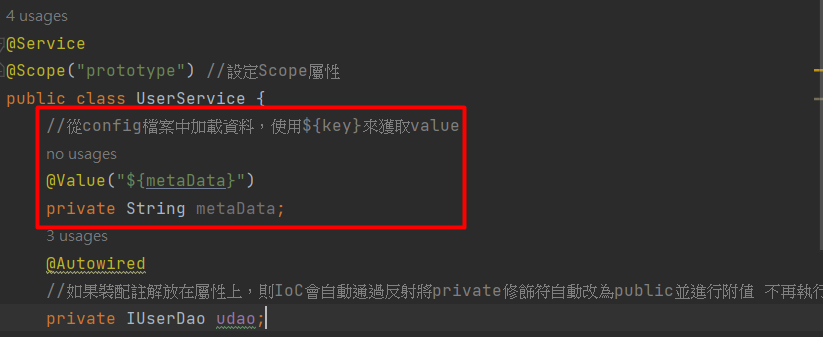
(3)在config中加入key-value pair



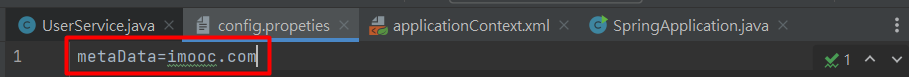
(4)在applicationContext.xml中加入加載config.proerties的指令



(5)使用Value加載對應的資料

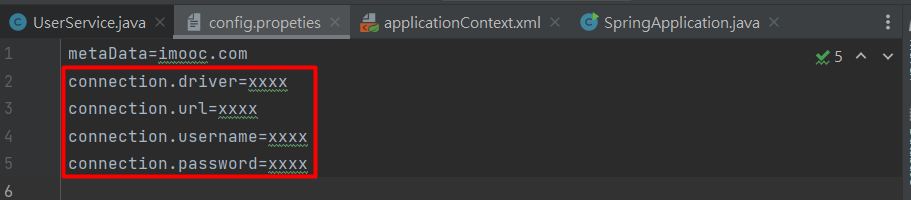


使用${key}的方式對應到config檔案中的key-value pair



[補充]properties檔案的命名規則

通常會在屬性的名稱前面加上xxx.代表xxx功能的某個屬性，例如使用connection.xxx來代表與資料庫連接的一系列屬性值。



而在程式中，只需要使用key的值來獲取value。

