

项目中配置多个Spring配置文件

优势：

- 1、相对于把所有信息写到一个配置文件中，多个配置文件的大小要小很多，效率高
- 2、多人开发时避免出现错误

当一个项目中需要多个模块共同完成时，每一个开发人员分别完成一个模块，每个模块配置自己的配置文件，这样分工有序，开发有条。

多文件分配方式：

- 1、按功能分，一个模块一个配置文件
- 2、按类的功能分，如数据库配置的类一个Spring配置文件，做事务的类一个配置文件，做service的类一个配置文件

在项目模块b-spring-di-fields-02中添加包bao04，复制bao03张的类到该包下，在resources目录新建bao04,。

这里我们为Student04类和School类分别配置一个Spring配置文件，在其中声明对象

spring-student04.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">

    <bean id="student03" class="com.studymyself.bao04.Student04"
        autowire="byName">
        <property name="name" value="智控3"/>
        <property name="age" value="2500"/>
        <!--<property name="school" ref="school"/>-->
    </bean>

</beans>
```

spring-school.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">

    <bean id="school" class="com.studymyself.bao04.School">
        <property name="name" value="北京大学4"/>
        <property name="addr" value="北京4"/>
    </bean>

</beans>
```

然后在定义一个total的包含关系配置文件，用来包含其他配置文件的

total.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">

    <!--
        total.xml表示主配置文件：用来包含其他配置文件，一般定义对象
        包含语法：
        <import resource="classpath:其他配置文件路径"/>
        关键字classpath解析：
        加上该关键字表示让Spring从项目的类根路径中开始扫描，而其他配置文件
        路径应该放在类根路径之下
    -->

    <!-- 第一种方式加载所有配置文件：文件列表方式，一个一个写上去-->
    <!--    <import resource="classpath:bao04/spring-school.xml"/>-->
    <!--    <import resource="classpath:bao04/spring-student04.xml"/>-->

    <!-- 第二种用通配符*的方式，*代表任意字符
        这种方式要求多个配置文件需要放到同一个目录下才能发挥大效用
        而且不能写*.xml，因为这样包含了主配置文件，会让Spring进入
        扫描文件的死循环-->
    <import resource="classpath:bao04/spring-*.xml"/>
</beans>
```

测试程序MyTest04.java如下

```
package com.studymyself;

import com.studymyself.bao04.Student04;
import org.junit.Test;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
```

```
public class MyTest03 {  
  
    /**  
     * 测试多个配置文件  
     */  
    @Test  
    public void testStudent01(){  
  
        //填写主配置文件路径  
        String config = "bao04/total.xml";  
  
        //获取容器对象  
        ApplicationContext ac = new ClassPathXmlApplicationContext(config);  
  
        //获取容器中的对象，强制转换Student01类型  
        Student04 student04 = (Student04) ac.getBean("student04");  
        //输出对象  
        System.out.println(student04);  
  
    }  
  
}
```