

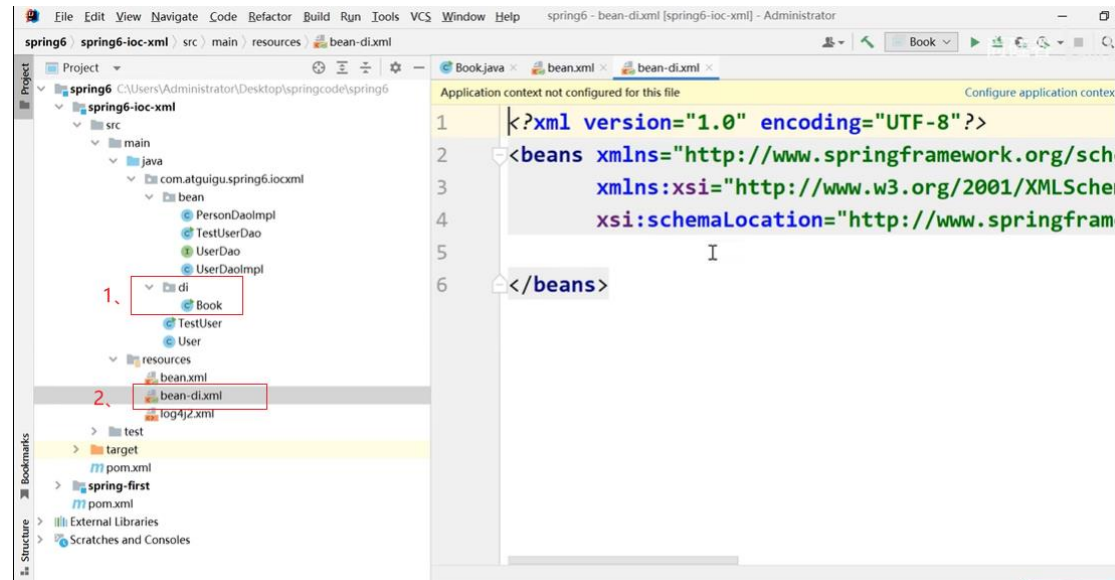
依赖注入

1、类有属性，创建对象过程中，向属性设置值

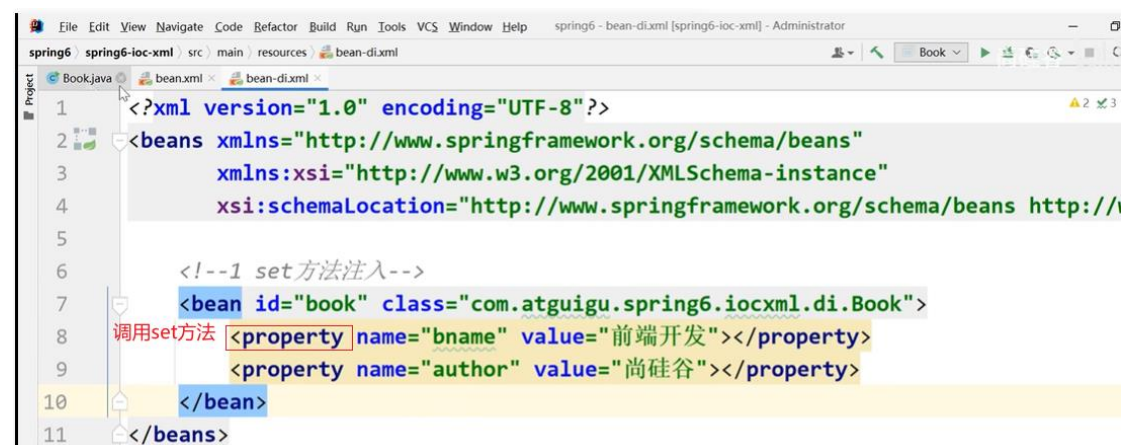
1) 基于 set 方法完成

2) 基于构造器完成

环境搭建：



一、set 方法的注入



测试：

```
spring6 > spring6-ioc.xml > src > main > java > com > atguigu > spring6 > iocxml > di > TestBook > testSetter
Book.java > bean-di.xml > TestBook.java
3 import org.junit.jupiter.api.Test;
4 import org.springframework.context.ApplicationContext;
5 import org.springframework.context.support.ClassPathXmlApplicationContext;
6
7 public class TestBook {
8
9     @Test
10    public void testSetter() {
11        ApplicationContext context =
12            new ClassPathXmlApplicationContext( configLocation: "bean-di.xml");
13        Book book = context.getBean( name: "book", Book.class);
14        System.out.println(book);
15    }
16 }
```

3) 构造方法注入

```
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help spring6 - bean-di.xml [spring6-ioc.xml] - Administrator
ring6 > spring6-ioc.xml > src > main > resources > bean-di.xml
Book.java > bean-di.xml > TestBook.java
8 <property name= bname value= 前端开发 ></property>
9 <property name="author" value="尚硅谷"></property>
10 </bean>
11
12 <!--2 构造器注入-->
13 <bean id="bookCon" class="com.atguigu.spring6.iocxml.di.Book">
14     <constructor-arg name="bname" value="java开发"></constructor-arg>
15     <constructor-arg name="author" value="尚硅谷"></constructor-arg>
16 </bean> index 根据索引定位, 不好用
17
18 </beans>
```

```
ioc.xml > src > main > java > com > atguigu > spring6 > iocxml > di > Book > Book
bean-di.xml > TestBook.java
}

//有参数构造方法
1 usage
public Book(String bname, String author) {
    System.out.println("有参数构造执行了。。。");
    this.bname = bname;
    this.author = author;
}
```

测试：

```
springio / springio-loc.xml / src / main / java / com / dygugu / springio / loc.xml / ui / testbook / testConstructor
Book.java x bean-di.xml x TestBook.java x
12      new ClassPathXmlApplicationContext( configLocation: "bean-di.xml" )
13      Book book = context.getBean( name: "book", Book.class);
14      System.out.println(book);
15  }
16
17  @Test
18  public void testConstructor() {
19      ApplicationContext context =
20          new ClassPathXmlApplicationContext( configLocation: "bean-di.xml" )
21      Book book = context.getBean( name: "bookCon", Book.class);
22      System.out.println(book);
23  }
24  }
25
```