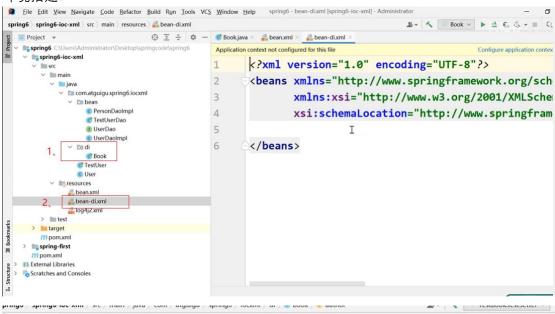
依赖注入

- 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:



```
🎯 Book.java × 🊜 bean-di.xml × 🍏 TestBook.java
10
              1 usage
11
              public Book(String bname, String author) {
12
                   this.bname = bname;
13
                   this.author = author;
14
                                 无参、有参构造器、get和set方法、toString()
15
              //生成set方法
16
17
              public String getBname() {
18
                   return bname;
19
   TestBook.testSetter
```

一、set 方法的注入

```
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help spring6 - bean-dixml [spring6-ioc-xml] - Administrator
                                                                                                                                                                                                                                                                                                        \mathbf{spring6} \ \rangle \ \mathbf{spring6\text{-}ioc\text{-}xml} \ \rangle \ \mathsf{src} \ \rangle \ \mathsf{main} \ \rangle \ \mathsf{resources} \ \rangle \ \underset{}{\not \leqslant} \ \mathsf{bean\text{-}di.xml}
<?xml version="1.0" encoding="UTF-8"?>
                                                                                                                                                                                                                                                                                                                                                                                                A2 x3
        1
          2
                                <beans xmlns="http://www.springframework.org/schema/beans"</pre>
                                                                     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           3
          4
                                                                     xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans ht
          5
           6
                                                        <!--1 set方法注入-->
           7
                                                         <bean id="book" class="com.atguigu.spring6.iocxml.di.Book">
                                调用set方法 <property | name="bname" value="前端开发"></property>
          8
          9
                                                                            cproperty name="author" value="尚硅谷"></property>
      10
                                                         </bean>
      11
                                      </beans>
```

测试:

```
spring6 | spring6-ioc-xml | src | main | java | com | atguigu | spring6 | iocxml | di | © TestBook | @ i testSetter
                                                                          &- < Book ∨ ▶ ± € € - □ O
g s Bookjava × bean-di.xml × s TestBookjava ×
       import org.junit.jupiter.api.Test;
         import org.springframework.context.ApplicationContext;
         import org.springframework.context.support.ClassPathXmlApplicationContext;
  7 N
         public class TestBook {
  8
  9
              @Test
 10
              public void testSetter() {
                  ApplicationContext context =
 11
                            new ClassPathXmlApplicationContext( configLocation: "bean-di.xml")
 12
                  Book book = context.getBean( name: "book", Book.class);
 13
14
                  System.out.println(book);
15
g 16
```

3) 构造方法注入

```
Eile Edit View Navigate Code Refactor Build Run Iools VCS Window Help spring6 - bean-dixml [spring6-ioc-xml] - Administrator
                                                              B → 

    TestBook.testSetter 

    S ← 

    TestBook.testSetter 

    TestBook.testSetter 
ring6 | spring6-ioc-xml | src | main | resources | 👼 bean-di.xml
                9
                cproperty name="author" value="尚硅谷"></property>
10
           </bean>
11
       * <!--2 构造器注入-->
12
            <bean id="bookCon" class="com.atguigu.spring6.iocxml.di.Book">
13
14
                <constructor-arg name="bname" value="java开发"></constructor-arg>
                <constructor-arg name="author" value="尚硅谷"></constructor-arg>
15
16
           </bean>
                                   idex 根据索引定位,不好用
17
       </beans>
18
oc-xml | src | main | java | com | atguigu | spring6 | iocxml | di | @ Book | @ Book
                                                                          # TestBoo
🍰 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:

```
Springe / Springe-to-sam / St./ Inditi / Java / Com / diguigu / Springe / tocam / ut / 🐷 restoors / 🐷 = testconstructor 💮 😅 😭 📉 👚 restoors.testsette 💌 📂 😅 🔩 💢 🛣 💢 💢
© Bookjava × ♣ bean-dixml × © TestBookjava × new ClassPatnxmlApplicationContext( CONTIGEOCATION: "bean-di.xml")
13
                   Book book = context.getBean( name: "book", Book.class);
 14
                   System.out.println(book);
 15
 16
 17
              @Test
              public void testConstructor() {
 18 ▶ □
 19
                   ApplicationContext context =
                            new ClassPathXmlApplicationContext( configLocation: "bean-di.xml")
 20
                   Book book = context.getBean( name: "bookCon", Book.class);
 21
                   System.out.println(book);
 22
23
              }
         }
24
25
```