

实验 9：引入外部属性文件

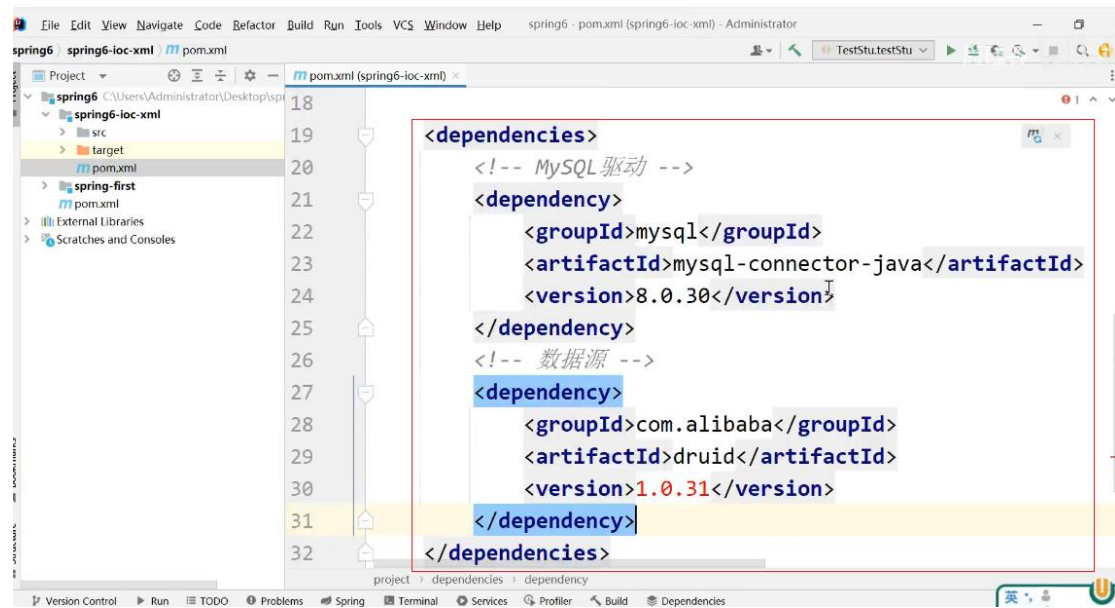
引入外部属性文件实现步骤

1、引入数据库相关依赖

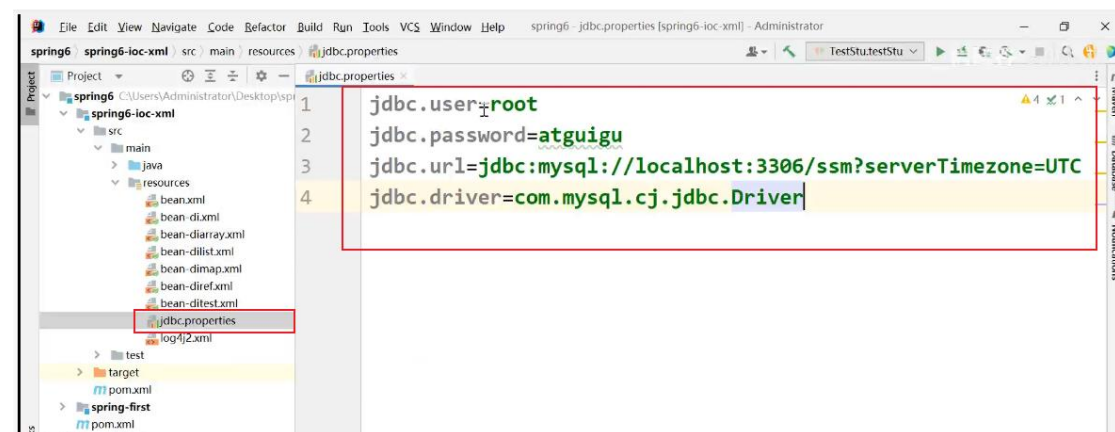
2、创建外部属性文件，properties格式， 定义数据信息：用户名 密码 地址等

3、创建spring配置文件，引入context命名空间 引入属性文件，使用表达式完成注入

1、引入数据库相关依赖



2、创建外部属性文件



3、引入 Context 命名空间

```
spring6 spring6-loc.xml | src | main | resources | bean-jdbc.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4       xmlns:context="http://www.springframework.org/schema/context"
5       xsi:schemaLocation="http://www.springframework.org/schema/context
6       http://www.springframework.org/schema/context/spring-context.xsd
7       http://www.springframework.org/schema/beans
8       http://www.springframework.org/schema/beans/spring-beans.xsd">
9
10 </beans>
```

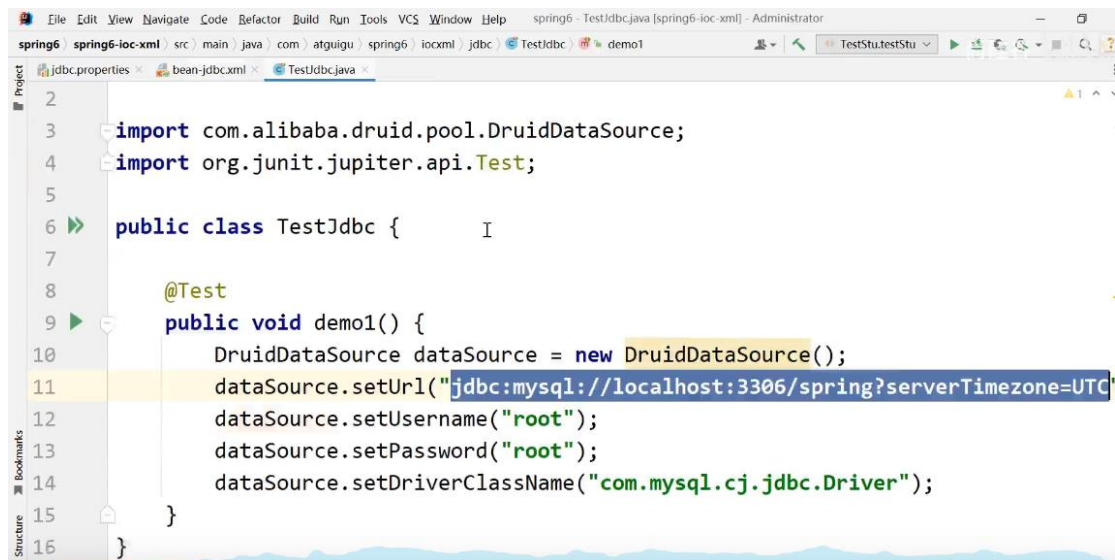
4、

```
spring6 spring6-loc.xml | src | main | resources | bean-jdbc.xml
10 <!-- 引入外部属性文件-->
11 <context:property-placeholder location="classpath:jdbc.properties"/></context:
12
13 <!-- 完成数据库信息注入-->
14 <bean id="druidDataSource" class="com.alibaba.druid.pool.DruidDataSource">
15   <property name="url" value="${jdbc.url}"/></property>
16   <property name="username" value="${jdbc.user}"/></property>
17   <property name="password" value="${jdbc.password}"/></property>
18   <property name="driverClassName" value="${jdbc.driver}"/></property>
19 </bean>
20
21 </beans>
```

5、测试:

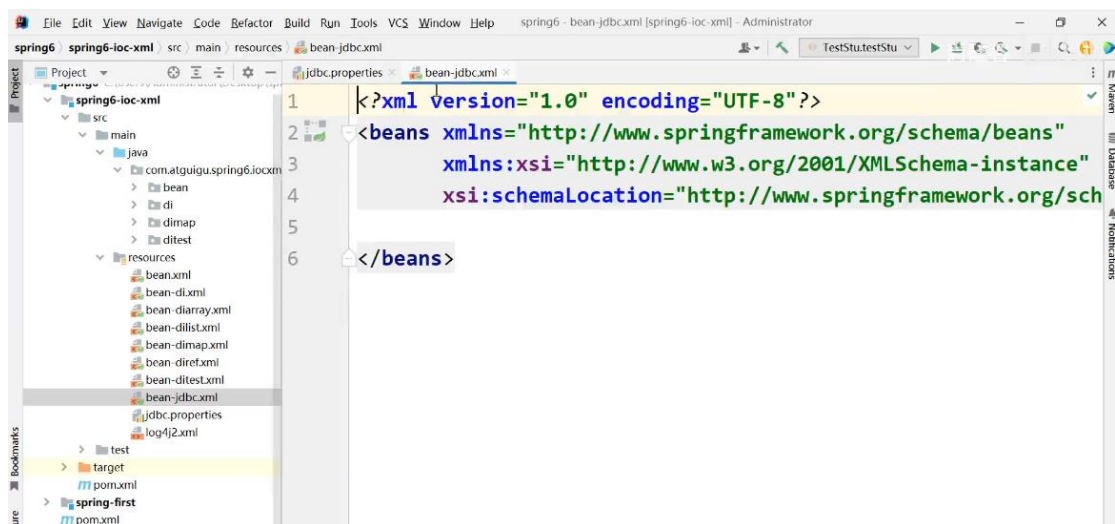
```
spring6 spring6-loc.xml | src | main | java | com | atguigu | spring6 | iocxml | jdbc | TestJdbc.java
3 import com.alibaba.druid.pool.DruidDataSource;
4 import org.junit.jupiter.api.Test;
5 import org.springframework.context.ApplicationContext;
6 import org.springframework.context.support.ClassPathXmlApplicationContext;
7
8 public class TestJdbc {
9
10   @Test
11   public void demo2() {
12     ApplicationContext context =
13       new ClassPathXmlApplicationContext("bean-jdbc.xml");
14     DruidDataSource dataSource = context.getBean(DruidDataSource.class);
15     System.out.println(dataSource.getUrl());
16   }
17 }
```

对比:



```
2
3 import com.alibaba.druid.pool.DruidDataSource;
4 import org.junit.jupiter.api.Test;
5
6 public class TestJdbc {
7
8     @Test
9     public void demo1() {
10         DruidDataSource dataSource = new DruidDataSource();
11         dataSource.setUrl("jdbc:mysql://localhost:3306/spring?serverTimezone=UTC");
12         dataSource.setUsername("root");
13         dataSource.setPassword("root");
14         dataSource.setDriverClassName("com.mysql.cj.jdbc.Driver");
15     }
16 }
```

补充：看目录结构



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4       xsi:schemaLocation="http://www.springframework.org/sch
5
6 </beans>
```