

## Spring Boot整合MyBatis多数据源

### Spring Boot整合MyBatis多数据源

在一些复杂的应用开发中，一个应用可能会涉及到连接多个数据源，所谓多数据源这里就定义为至少连接两个及以上的数据库了。

### 首先创建Spring Boot工程

添加MyBatis、Druid依赖（这里的Druid必须是专为Spring boot打造的Druid，不能使用传统的Druid）、MySQL以及Web依赖，完整的依赖如下：

```
<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
    <dependency>
        <groupId>org.mybatis.spring.boot</groupId>
        <artifactId>mybatis-spring-boot-starter</artifactId>
        <version>2.1.0</version>
    </dependency>
    <dependency>
        <groupId>mysql</groupId>
        <artifactId>mysql-connector-java</artifactId>
        <!--注意配置版本号-->
        <version>5.1.28</version>
        <scope>runtime</scope>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>
    <!--druid连接池-->
    <dependency>
        <groupId>com.alibaba</groupId>
        <artifactId>druid-spring-boot-starter</artifactId>
        <version>1.1.10</version>
    </dependency>
</dependencies>
```

### 多数据源配置

接下来配置多数据源，在application.properties中配置数据库基本信息，然后提供两个DataSource即可，下面是简单的application.properties中的配置：

```
spring.datasource.one.url=jdbc:mysql://localhost:3306/test01?
useUnicode=true&characterEncoding=utf-8
spring.datasource.one.username=root
spring.datasource.one.password=123456
spring.datasource.one.type=com.alibaba.druid.pool.DruidDataSource
```

### 公告

昵称： nextsummer  
园龄： 5个月  
粉丝： 9  
关注： 17  
[+加关注](#)

< 2019年9月 >						
日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

### 搜索

<input type="text"/>	<input type="button" value="找找看"/>
<input type="text"/>	<input type="button" value="谷歌搜索"/>

### 常用链接

[我的随笔](#)  
[我的评论](#)  
[我的参与](#)  
[最新评论](#)  
[我的标签](#)

### 随笔分类

[JavaScript + jQuery\(1\)](#)  
[JAVA面试资料\(8\)](#)  
[JDBC学习之路\(9\)](#)  
[Spring Boot\(2\)](#)  
[SpringMVC\(1\)](#)

### 随笔档案

[2019年8月\(1\)](#)  
[2019年7月\(7\)](#)  
[2019年4月\(11\)](#)

### 最新评论

1. Re:Spring事务管理

大佬牛批，偷偷学习被我发现

--偷偷学习被我发现

2. Re:Spring 中Bean的生命周期

大佬牛批，不亏是阿里巴巴的程序员

--偷偷学习被我发现

3. Re:Java面试题：final修饰的对象，有几种初始化方式

BAT大佬这么低调的吗？牛批

--偷偷学习被我发现

4. Re:HTTP的一些基本概念

优秀

--69之王

## 阅读排行榜

1. Java面试题：final修饰的对象，有几种初始化方式(332)

2. springboot+druid+mybatis+mysql+多数据源事务管理(161)

3. Java面试：用set集合的时候，重写过hashCode()和equal()方法吗？有什么作用？(146)

4. JVM双亲委派模型(93)

5. JDBC 初识(88)

## 评论排行榜

1. HTTP的一些基本概念(1)

2. Spring 中Bean的生命周期(1)

3. Java面试题：final修饰的对象，有几种初始化方式(1)

4. Spring事务管理(1)

```
spring.datasource.two.url=jdbc:mysql://localhost:3306/test02?
useUnicode=true&characterEncoding=utf-8
spring.datasource.two.username=root
spring.datasource.two.password=123456
spring.datasource.two.type=com.alibaba.druid.pool.DruidDataSource
```

然后创建DataSourceConfig 配置类，里面配置对应的两个数据源：

```
@Configuration
public class DataSourceConfig {

    @Bean
    @ConfigurationProperties(prefix = "spring.datasource.one")
    DruidDataSource dsOne() {
        return DruidDataSourceBuilder.create().build();
    }

    @Bean
    @ConfigurationProperties(prefix = "spring.datasource.two")
    DruidDataSource dsTwo() {
        return DruidDataSourceBuilder.create().build();
    }
}
```

## MyBatis配置

这里需要配置两个Bean，所以两个数据源分两个类配置：

```
@Configuration//声明该类是一个配置类
@MapperScan(basePackages = "com.lwh.mybatistest.mapper", sqlSessionSessionFactoryRef =
"sqlSessionFactory1", sqlSessionTemplateRef = "sqlSessionTemplate1")
//扫描的包是com.lwh.mybatistest.mapper
//SqlSessionFactory根据dsOne创建，然后再根据创建好的SqlSessionFactory创建一个SqlSessionTemplate。
public class MyBatisConfigOne {

    @Resource(name = "dsOne")
    DataSource dsOne;

    @Bean
    SqlSessionFactory sqlSessionFactory1() {
        SqlSessionFactory sessionFactory = null;
        try {
            SqlSessionFactoryBean bean = new SqlSessionFactoryBean();
            bean.setDataSource(dsOne);
            sessionFactory = bean.getObject();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return sessionFactory;
    }

    @Bean
    SqlSessionTemplate sqlSessionTemplate1() {
        return new SqlSessionTemplate(sqlSessionFactory1());
    }
}
```

依照第一个配置，配置第二个数据源即可，这里需要注意因为操作的是不同的数据源，所以扫描的包是不一样的，不同的Mapper将操作不同的数据源。如下：

```
@Configuration//声明该类是一个配置类
@MapperScan(basePackages = "com.lwh.mybatistest.mapper2", sqlSessionSessionFactoryRef =
"sqlSessionFactory2", sqlSessionTemplateRef = "sqlSessionTemplate2")
public class MyBatisConfigTwo {

    @Resource(name = "dsTwo")
    DataSource dsTwo;

    @Bean
    SqlSessionFactory sqlSessionFactory2() {
        SqlSessionFactory sessionFactory = null;
    }
}
```

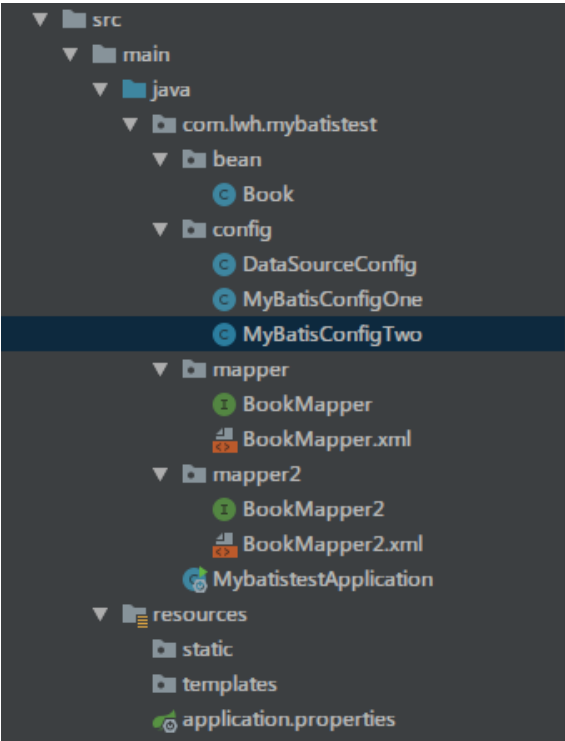
```

    try {
        SqlSessionFactoryBean bean = new SqlSessionFactoryBean();
        bean.setDataSource(dsTwo);
        sessionFactory = bean.getObject();
    } catch (Exception e) {
        e.printStackTrace();
    }
    return sessionFactory;
}

@Bean
SqlSessionTemplate sqlSessionTemplate2() {
    return new SqlSessionTemplate(sqlSessionFactory2());
}
}

```

项目结构如下：



上面的mapper.xml文件是用MyBatis逆向工程生成的，这里不再赘述。

在Maven项目中，官方建议我们把XML文件放在resources文件下，如果放在java下面，项目打包的时候会自动的过滤掉，所以我们在Maven的配置文件中，让项目不要过滤我们的XML文件，因此我们需要在pom.xml配置文件中的 build 节点里面添加配置信息，如下：

```

<build>
    <resources>
        <resource>
            <directory>src/main/java</directory>
            <includes>
                <include>/**/*.xml</include>
            </includes>
        </resource>
        <resource>
            <directory>src/main/resources</directory>
        </resource>
    </resources>
</build>

```

这样项目在打包发布的时候，就不会过滤掉我们的mapper.xml文件了

数据库one的数据为：

id	bookname	author
1	三国演义	罗贯中

数据库two的数据为:

id	bookname	author
2	红楼梦	曹雪芹

现在去测试类测试我们的配置是否成功:

```
@RunWith(SpringRunner.class)
@SpringBootTest
public class MybatisTestApplicationTests {

    @Autowired
    BookMapper bookMapper;

    @Autowired
    BookMapper2 bookMapper2;

    @Test
    public void contextLoads() {
        List<Book> list = bookMapper.getAllBook();
        System.out.println("第一个: " + list);
        System.out.println("第二个: " + bookMapper2.getAllBook());
    }
}
```

输出结果为:

```
第一个: [Book{id=1, bookname='三国演义', author='罗贯中'}]
第二个: [Book{id=2, bookname='红楼梦', author='曹雪芹'}]
```

因此我们的项目多数据源的操作就配置成功了!

## 总结:

- 1, 首先给Spring Boot项目引入相关的依赖。
- 2, 在application.properties中配置多个数据库相关信息, 创建DataSource配置类。
- 3, 分别给不同的数据库配置对应的Mybatis配置类, 不同的mapper操作对应的数据库。

以上配置教程参考江南一点雨的博客。

分类: [JAVA面试资料](#), [Spring Boot](#)

好文要顶 关注我 收藏该文





[nextsummer](#)  
关注 - 17  
粉丝 - 9

00

[+加关注](#)

« 上一篇: [Spring事务管理](#)  
» 下一篇: [springboot+druid+mybatis+mysql+多数据源事务管理](#)

注册用户登录后才能发表评论, 请 [登录](#) 或 [注册](#), [访问网站首页](#)。

【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库  
【活动】京东云限时优惠1.5折购云主机, 最高返价值1000元礼品!  
【推荐】零基础轻松玩转华为云产品, 获赠礼加返百元大礼  
【推荐】919 天翼云钜惠, 全网低价, 云主机9元轻松购  
【推荐】华为云文字识别资源包重磅上市, 1元万次限时抢购  
【推荐】腾讯云海外云服务器1核2G19.8元/月  
【福利】git pull && cherry-pick 博客园&华为云百万代金券

#### 相关博文:

- 小D课堂-SpringBoot2.x微信支付在线教育网站项目实战\_3-1.整合Mybatis访问数据库和阿里巴巴数据源
- springboot配置druid连接池连接mysql
- spring\_boot与MyBatis整合
- Spring Boot Mybatis 整合
- springboot集成mybatis(1)

#### 最新 IT 新闻:

- WFIRST太空望远镜“恒星眼镜”将帮助研究系外行星
  - 美国警察开特斯拉追疑犯，时速飙到193公里，然后发现没电了...
  - 阿里云发布第三代神龙架构：5倍性能提升，统一IaaS平台
  - 美图与华为签订战略合作协议 将在5G等领域深度合作
  - 不止发烫、信号差！iPhone 11系列再曝“进灰门” 处女座表示受不了
- » 更多新闻...