



链滴

MiniDao 学习系列 1- - - - 集成使用

作者: [panqq](#)

原文链接: <https://ld246.com/article/1537089268066>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



MiniDao学习系列1- - - - 集成使用

背景

springboot		1.5.6.RELEASE 不支持2.x版本
jdk	1.8	
mysql	5.7	
minidao-pe		1.6.7
commons-pool		1.6
commons-dbcp		1.4
druid-spring-boot-starter		1.1.5

引入minidao依赖

```
<!-- https://mvnrepository.com/artifact/org.jeecgframework/minidao-pe -->
<dependency>
  <groupId>org.jeecgframework</groupId>
  <artifactId>minidao-pe</artifactId>
  <version>1.6.7</version>
</dependency>
```

注册MiniDao动态代理类

在*application.java启动类同级或下级包目录下创建config.java。用来注册minidao代理

```

package com.demo;

import org.jeecgframework.minidao.factory.MinidaoBeanScannerConfigurer;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.stereotype.Repository;

@Configuration
public class Config {

    @Bean
    public MiniDaoBeanScannerConfigurer miniDaoHandler() {
        MiniDaoBeanScannerConfigurer miniDaoHandler = new MiniDaoBeanScannerConfigurer
    );
        miniDaoHandler.setAnnotation(Repository.class);//使用的注解,默认是Minidao,推荐 Repository
        miniDaoHandler.setBasePackage("com.jeecg.demo.dao,com.jeecg.demo,com.demo.users
dao");//dao地址,配置符合spring方式
        miniDaoHandler.setDbType("mysql");//数据库类型 mysql/postgresql/oracle/sqlserver
        miniDaoHandler.setShowSql(false);//输出sql
        miniDaoHandler.setFormatSql(false);//格式化sql
        miniDaoHandler.setKeyType("lower");//是使用什么字母做关键字Map的关键字 默认值origin
即和sql保持一致,lower小写(推荐),upper 大写
        return miniDaoHandler;
    }

}

```

测试用例

```

package com.demo;

import com.demo.users.dao.UsersDao;
import com.demo.users.entity.Users;
import org.junit.Test;
import org.junit.runner.RunWith;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.context.annotation.ImportResource;
import org.springframework.test.context.junit4.SpringRunner;

import java.util.List;

@RunWith(SpringRunner.class)
@SpringBootTest
public class MinidaoDemoApplicationTests {

    @Autowired(required = false)
    private UsersDao usersDao;
    @Test
    public void contextLoads() {
        List<Users> users = usersDao.list();
    }
}

```

```
    if (users.isEmpty()) {
        System.out.println("空的");
    }else{
        System.out.println("启动完成:" + users.toString());
    }
}
}
```

运行结果

![1537087593661](J:\18 我的文章\1537087593661.png)

更换连接池为Druid

修改pom.xml, 注释掉DBCP连接池

```
<dependency>
  <groupId>com.alibaba</groupId>
  <artifactId>druid-spring-boot-starter</artifactId>
  <version>1.1.5</version> <!-- 依赖springboot 1.5.6.RELEASE ; 1.1.7 对应 2.x版本-->
</dependency>
```

修改application.properties

```
# Druid连接池配置, 官方配置参考: https://github.com/alibaba/druid/tree/master/druid-spring
boot-starter
spring.datasource.druid.driver-class-name= com.mysql.jdbc.Driver
spring.datasource.druid.url= jdbc:mysql://localhost:3306/test?useUnicode=true&characterEn
oding=utf-8&useSSL=false
spring.datasource.druid.username= root
spring.datasource.druid.password= 123456
```

运行结果:

![1537088545473](J:\18 我的文章\1537088545473.png)

[项目代码](#)

普通SpringMVC 集成使用

参考: [Minidao 教程^V1.1^](#)