此项目引入的是

<dependency>  
 <groupId>com.alibaba</groupId>  
 <artifactId>druid-spring-boot-starter</artifactId>  
 <version>1.1.10</version>  
</dependency>

Springboot和druib整合依赖

<dependency>  
 <groupId>org.slf4j</groupId>  
 <artifactId>log4j-over-slf4j</artifactId>  
 <version>1.7.25</version>  
</dependency>

作用是解决日志问题

注意：此项目不引入

<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-data-jpa</artifactId>  
</dependency>

会报错，原因未知

引入springboot和druib整合依赖 只需要写配置文件不需要写配置类

配置文件：

#jdbc配置  
spring.datasource.driver-class-name=com.mysql.jdbc.Driver  
spring.datasource.url=jdbc:mysql://localhost:3306/sharebed?useSSL=false&serverTimezone=UTC&useUnicode=true&characterEncoding=utf-8&allowMultiQueries=true  
spring.datasource.username=root  
spring.datasource.password=root  
spring.datasource.type=com.alibaba.druid.pool.DruidDataSource  
  
#连接池的设置  
#初始化时建立物理连接的个数  
spring.datasource.druid.initial-size=5  
#最小连接池数量  
spring.datasource.druid.min-idle=5  
#最大连接池数量 maxIdle已经不再使用  
spring.datasource.druid.max-active=20  
#获取连接时最大等待时间，单位毫秒  
spring.datasource.druid.max-wait=60000  
#申请连接的时候检测，如果空闲时间大于timeBetweenEvictionRunsMillis，执行validationQuery检测连接是否有效。  
spring.datasource.druid.test-while-idle=true  
#既作为检测的间隔时间又作为testWhileIdel执行的依据  
spring.datasource.druid.time-between-eviction-runs-millis=60000  
#销毁线程时检测当前连接的最后活动时间和当前时间差大于该值时，关闭当前连接  
spring.datasource.druid.min-evictable-idle-time-millis=30000  
#用来检测连接是否有效的sql 必须是一个查询语句  
#mysql中为 select 'x'  
#oracle中为 select 1 from dual  
spring.datasource.druid.validation-query=select 'x'  
#申请连接时会执行validationQuery检测连接是否有效,开启会降低性能,默认为true  
spring.datasource.druid.test-on-borrow=false  
#归还连接时会执行validationQuery检测连接是否有效,开启会降低性能,默认为true  
spring.datasource.druid.test-on-return=false  
#是否缓存preparedStatement,mysql5.5+建议开启  
spring.datasource.druid.pool-prepared-statements=true  
#当值大于0时poolPreparedStatements会自动修改为true  
spring.datasource.druid.max-pool-prepared-statement-per-connection-size=20  
#配置扩展插件  
spring.datasource.druid.filters=stat,wall  
#通过connectProperties属性来打开mergeSql功能；慢SQL记录  
spring.datasource.druid.connection-properties=druid.stat.mergeSql=true;druid.stat.slowSqlMillis=500  
#合并多个DruidDataSource的监控数据  
spring.datasource.druid.use-global-data-source-stat=true  
#设置访问druid监控页的账号和密码,默认没有  
#spring.datasource.druid.stat-view-servlet.login-username=admin  
#spring.datasource.druid.stat-view-servlet.login-password=admin  
  
#jpa设置  
spring.jpa.database=mysql  
spring.jpa.show-sql=true  
spring.jpa.hibernate.ddl-auto=update

项目启动成功后可访问<http://localhost:8080/druid/datasource.html> 查看数据库。