​

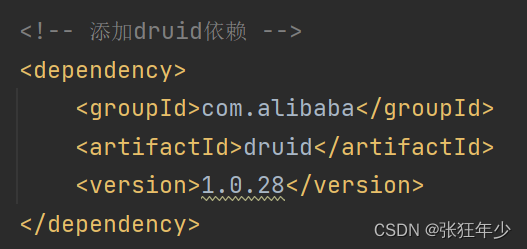
# spring boot集成数据库连接池druid

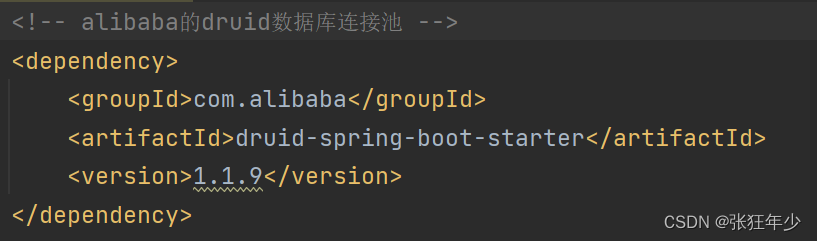
        最近在进行程序优化的过程中发现程序瓶颈在数据库连接这块，于是开始研究怎么对数据库连接池参数进行调优，在这个过程中发现很多人使用druid很不规范，经常会出现导入的包和配置参数不对应的情况，这些都是因为对集成druid一知半解导致的，因此决心写一篇关于spring boot集成数据库连接池druid的博客来彻底理解清楚其中的原理。

        在使用的过程中遇到spring boot集成druid分别有这三种情况：springboot集成druid、springboot集成druid-spring-boot-starter、springboot集成shardingsphere和druid，下面将分别来对这三种情况进行详细的分析。

# 一、springboot集成druid

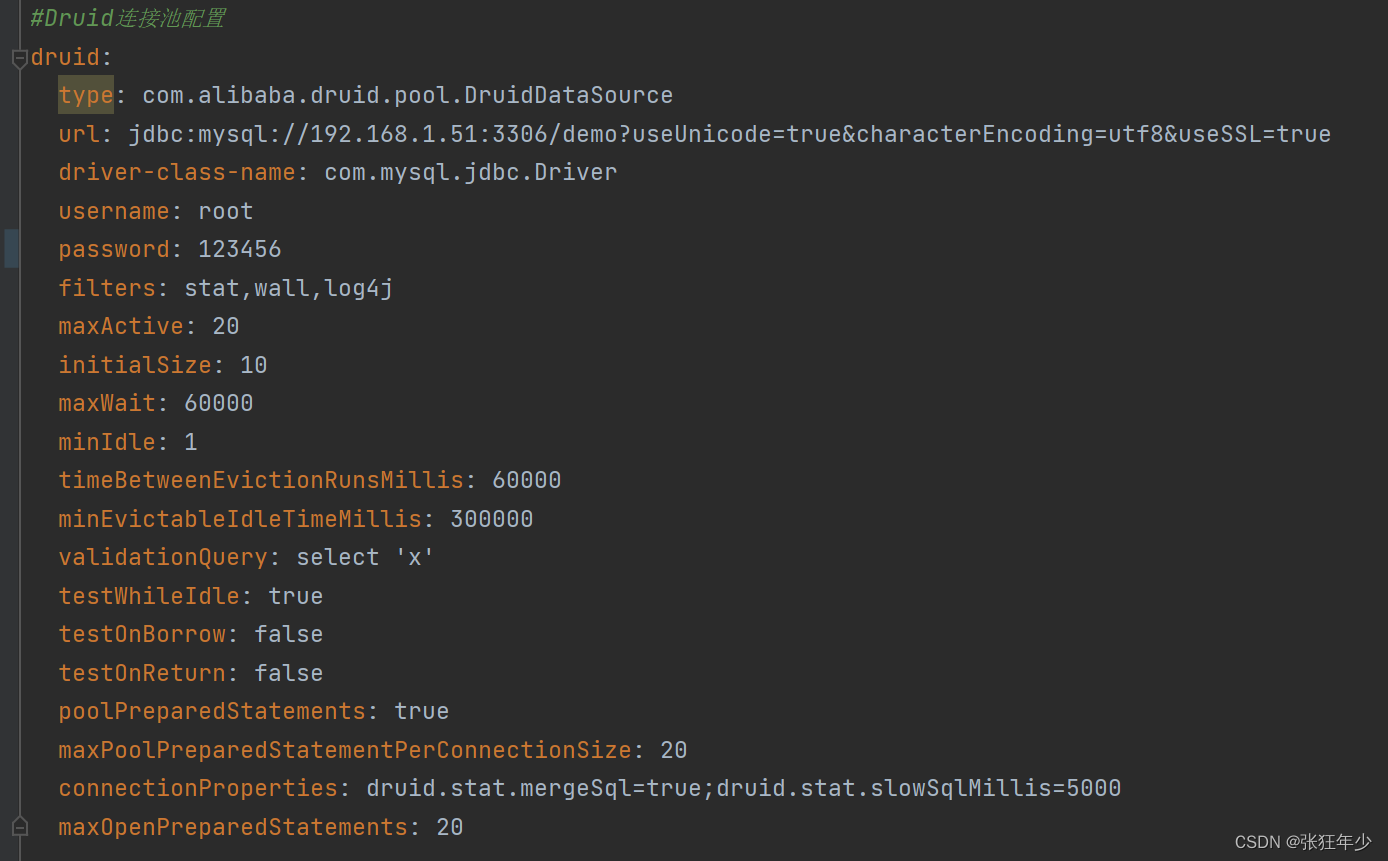
        springboot集成druid分别可以引入如下两种jar包：

点击并拖拽以移动​

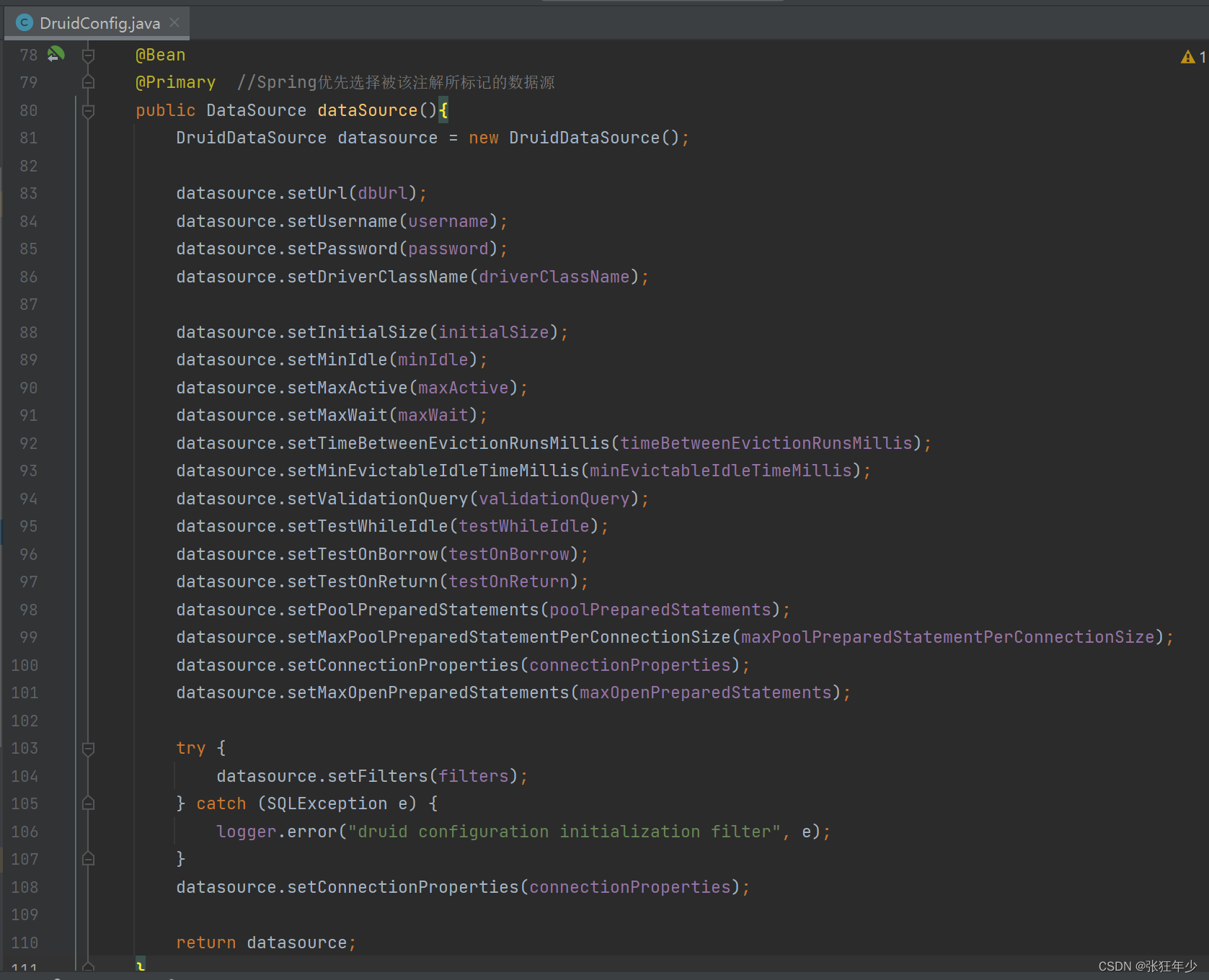
点击并拖拽以移动

springboot框架之前都是使用druid包， springboot问世之后开始引用starter包依赖，springboot的设计有一个很重要的思想是约定大于配置，第二节会讲解集成starter包依赖的方式，大家可以对比一下就能一目了然了。

集成druid包首先引入依赖，之后开始进行配置，yml配置如下：

点击并拖拽以移动

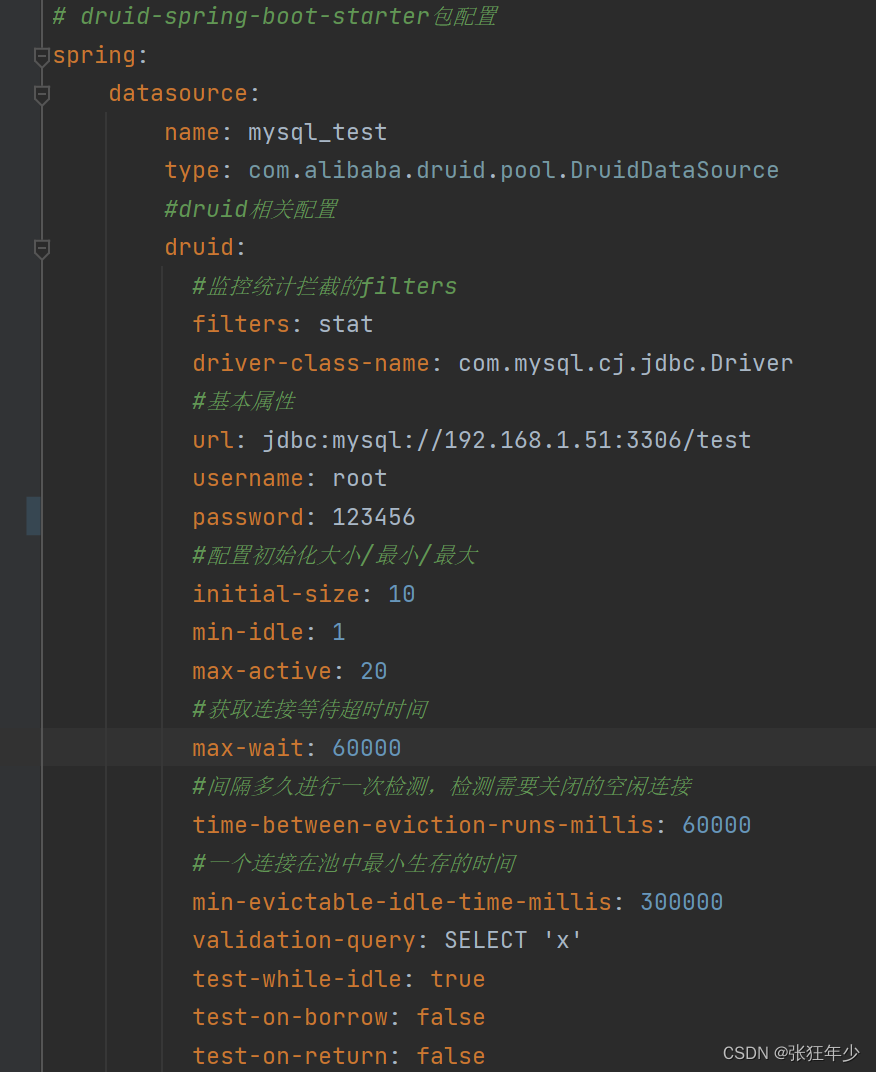
 光有这些配置druid并不能够生效，这里还需要加一个Javaconfig配置类DruidConfig，核心配置内容如下：

点击并拖拽以移动​

 这样spring boot集成druid的就完成了。

# 二、springboot集成druid-spring-boot-starter

        第一节讲到集成druid包，这一节会讲解集成starer包的方式，因为spring boot具备自动装配的功能，这里我们只需要引入starter包之后，加上如下配置就可以了：

点击并拖拽以移动

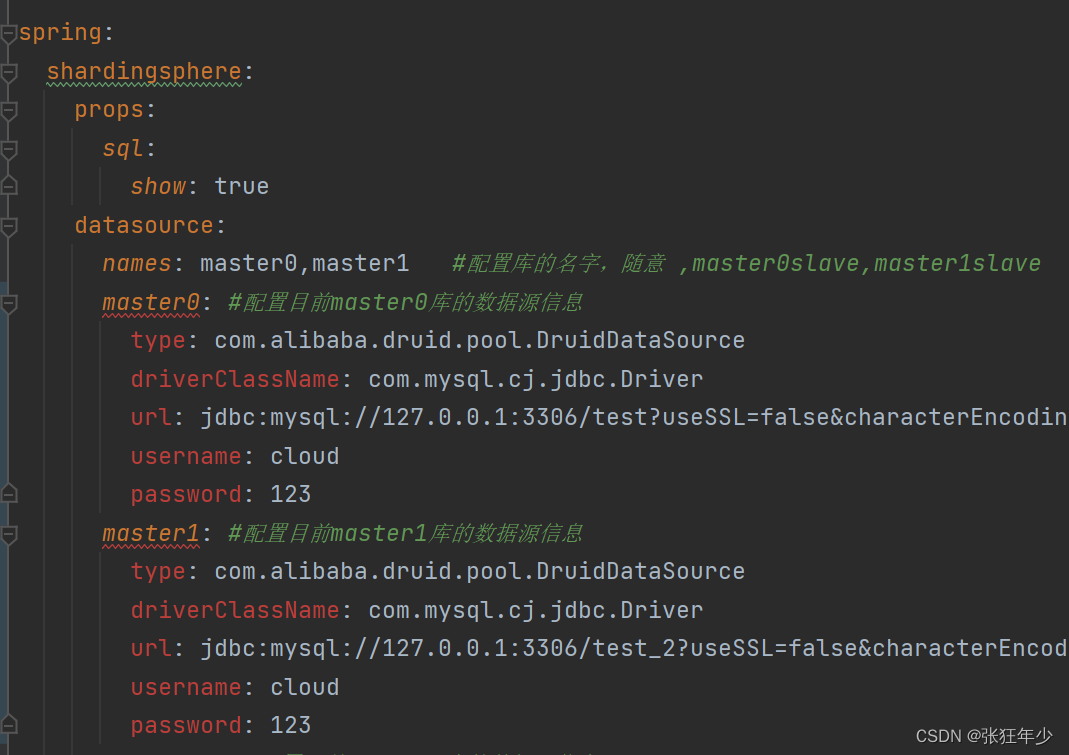
可见starter包依赖，简单了许多，这些归功于spring boot的约定大于配置的特性。

# 三、springboot集成shardingsphere和druid

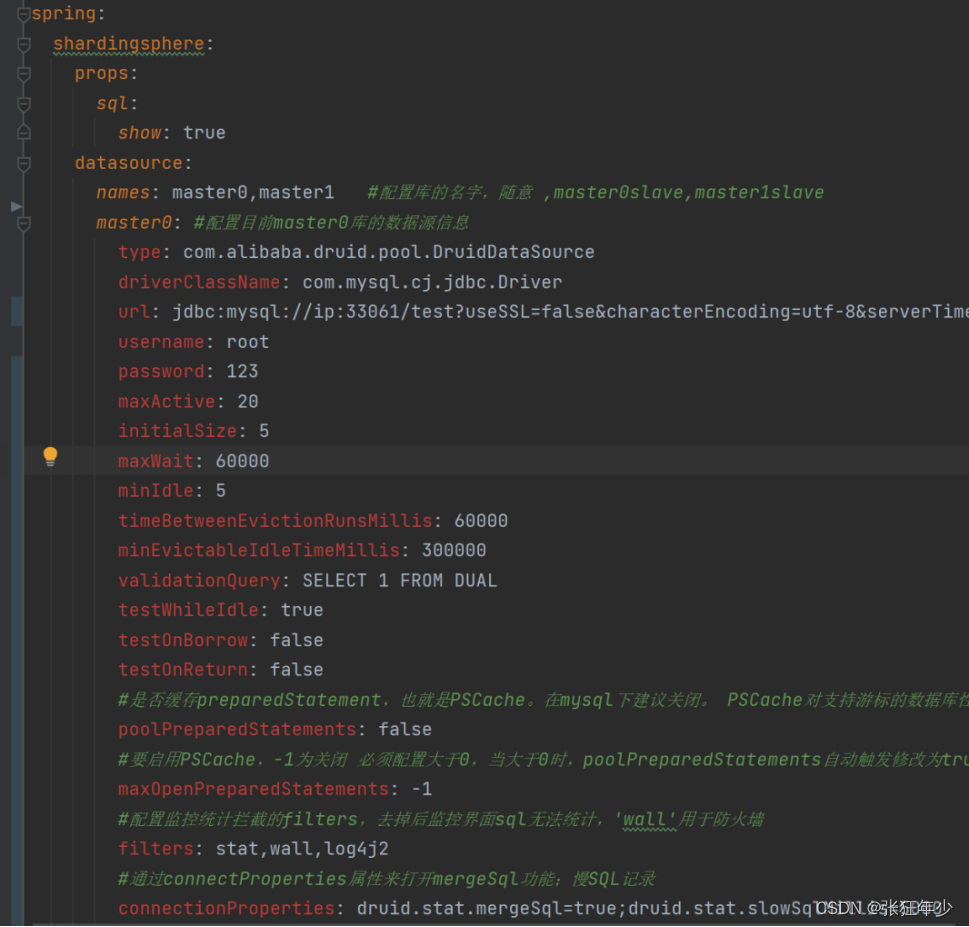
        这一步是由我们的业务系统联想到的，前面的集成是比较常规的，但是当我们由于业务量增大，服务扩展到了分库分表，而且采用了shardingjdbc方案的时候，这时候需要引入如下依赖：

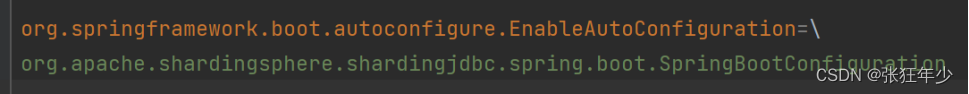
点击并拖拽以移动​

引入shardingjdbc之后数据库配置是这样的：

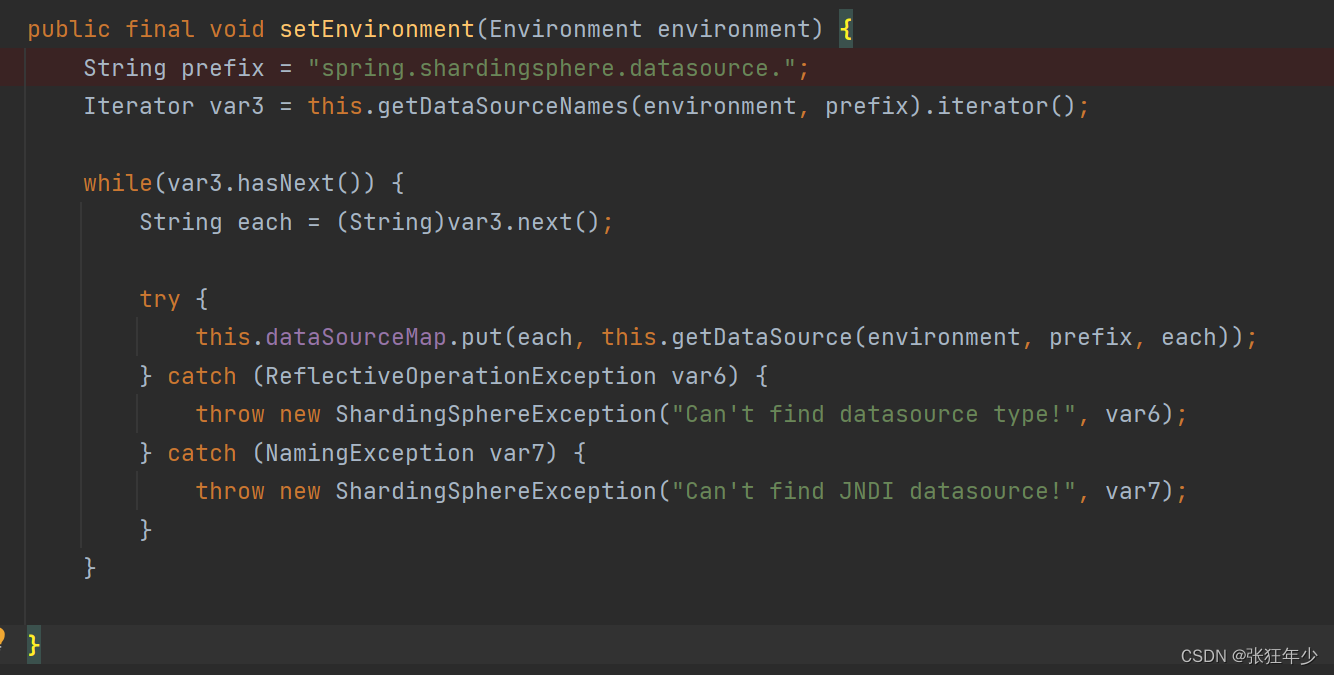
点击并拖拽以移动

        这时候很多人又开始迷糊了，连接池的配置该写在哪里呢，经过一系列资料阅读和源码研究，发现数据库连接池配置是这样的：

点击并拖拽以移动​  这里我们需找到sharding-jdbc-spring-boot-starter包下面的spring.factories文件，可以看到：

点击并拖拽以移动

 进入SpringBootConfiguration类，druid的配置通过如下方法加载:

点击并拖拽以移动​

顺着这个方法往下debug，我们就可以彻底理解清楚shardingjdbc方案分库分表之后数据库连接池怎么配置生效了。

总结：到此springboot集成数据库连接池druid常见的几种方式应该是讲清楚了，大家对着讲解可以去理解一下。

​