https://mp.weixin.qq.com/s/9Q2xk3L9OErPga6rAN25Fg

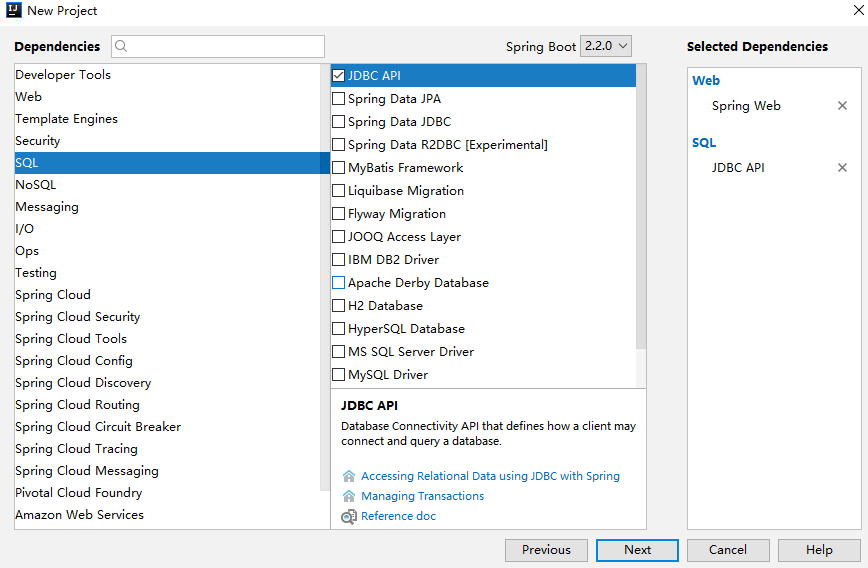
# Spring Boot 整合JdbcTemplate 多数据源

多数据源配置也算是一个常见的开发需求，Spring 和 SpringBoot 中，对此都有相应的解决方案，不过一般来说，如果有多数据源的需求，我还是建议首选分布式数据库中间件 MyCat 去解决相关问题，之前有小伙伴在我的知识星球上提问，他的数据根据条件的不同，可能保存在四十多个不同的数据库中，怎么办？这种场景下使用多数据源其实就有些费事了，我给的建议是使用 MyCat，然后分表策略使用 sharding-by-intfile 。

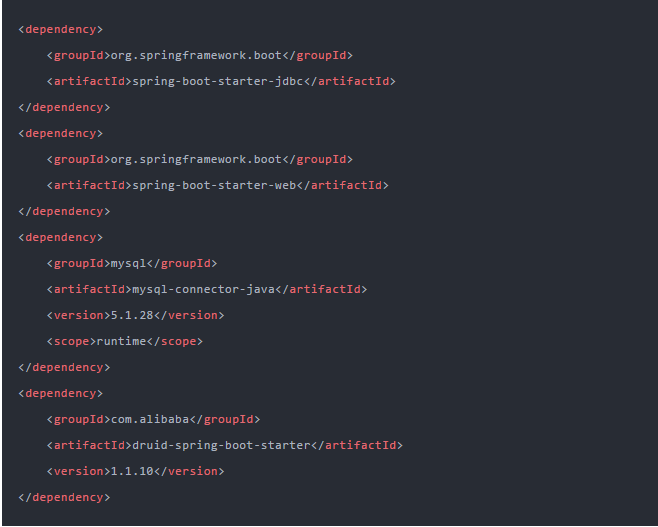
当然如果一些简单的需求，还是可以使用多数据源的，Spring Boot 中，JdbcTemplate、MyBatis 以及 Jpa 都可以配置多数据源，本文就先和大伙聊一聊 JdbcTemplate 中多数据源的配置（关于JdbcTemplate的用法，如果还有小伙伴不了解，可以参考我的 [**Spring Boot2 系列教程(十九)Spring Boot 整合 JdbcTemplate**](https://mp.weixin.qq.com/s?__biz=MzI1NDY0MTkzNQ==&mid=2247486840&idx=2&sn=5d4f75a81b3d8eb02f288a92015f247c&scene=21#wechat_redirect)）。

## 创建工程

首先是创建工程，和前文一样，创建工程时，也是选择 Web、Jdbc 以及 MySQL 驱动，如下图：

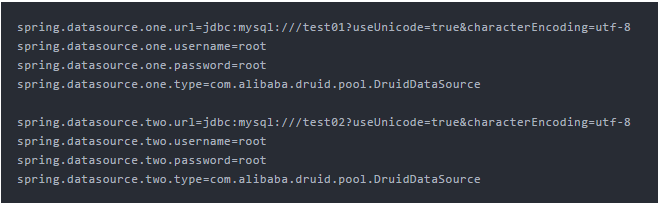


创建成功之后，一定接下来手动添加 Druid 依赖，由于这里一会需要开发者自己配置 DataSoruce，所以这里必须要使用 druid-spring-boot-starter 依赖，而不是传统的那个 druid 依赖，因为 druid-spring-boot-starter 依赖提供了 DruidDataSourceBuilder 类，这个可以用来构建一个 DataSource 实例，而传统的 Druid 则没有该类。完整的依赖如下：



## 配置数据源

接下来，在 application.properties 中配置数据源，不同于上文，这里的数据源需要配置两个，如下：



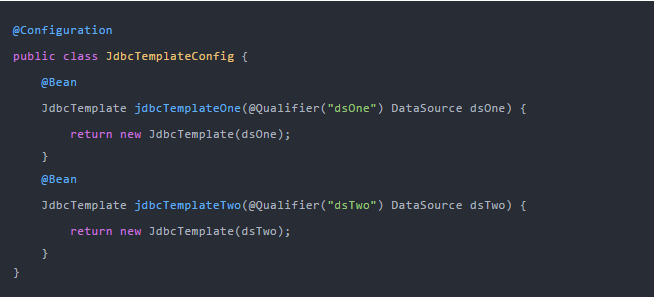
这里通过 one 和 two 对数据源进行了区分，但是加了 one 和 two 之后，这里的配置就没法被 SpringBoot 自动加载了（因为前面的 key 变了），需要我们自己去加载 DataSource 了，此时，需要自己配置一个 DataSourceConfig，用来提供两个 DataSource Bean，如下：



这里提供了两个 Bean，其中 @ConfigurationProperties 是 Spring Boot 提供的类型安全的属性绑定，以第一个Bean为例， @ConfigurationProperties(prefix = "spring.datasource.one") 表示使用 spring.datasource.one 前缀的数据库配置去创建一个 DataSource，这样配置之后，我们就有了两个不同的 DataSource，接下来再用这两个不同的 DataSource 去创建两个不同的 JdbcTemplate。

## 配置 JdbcTemplate 实例

创建 JdbcTemplateConfig 类，用来提供两个不同的 JdbcTemplate 实例，如下：



每一个 JdbcTemplate 的创建都需要一个 DataSource，由于 Spring 容器中现在存在两个 DataSource，默认使用类型查找，会报错，因此加上 @Qualifier 注解，表示按照名称查找。这里创建了两个 JdbcTemplate 实例，分别对应了两个 DataSource。

接下来直接去使用这个 JdbcTemplate 就可以了。

## 测试使用

关于 JdbcTemplate 的详细用法大伙可以参考我的上篇文章，这里我主要演示数据源的差异，在 Controller 中注入两个不同的 JdbcTemplate ，这两个 JdbcTemplate 分别对应了不同的数据源，如下：



和 DataSource 一样，Spring 容器中的 JdbcTemplate 也是有两个，因此不能通过 byType 的方式注入进来，这里给大伙提供了两种注入思路，一种是使用 @Resource 注解，直接通过 byName 的方式注入进来，另外一种就是 @Autowired 注解加上 @Qualifier 注解，两者联合起来，实际上也是 byName。将 JdbcTemplate 注入进来之后，jdbcTemplateOne 和 jdbcTemplateTwo 此时就代表操作不同的数据源，使用不同的 JdbcTemplate 操作不同的数据源，实现了多数据源配置。