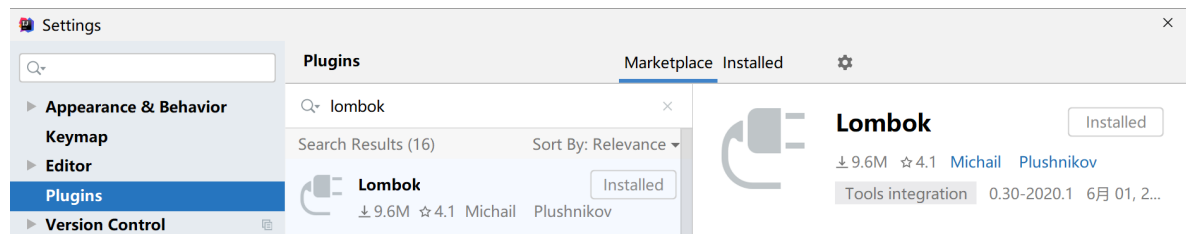


# 一、安装IDEA插件

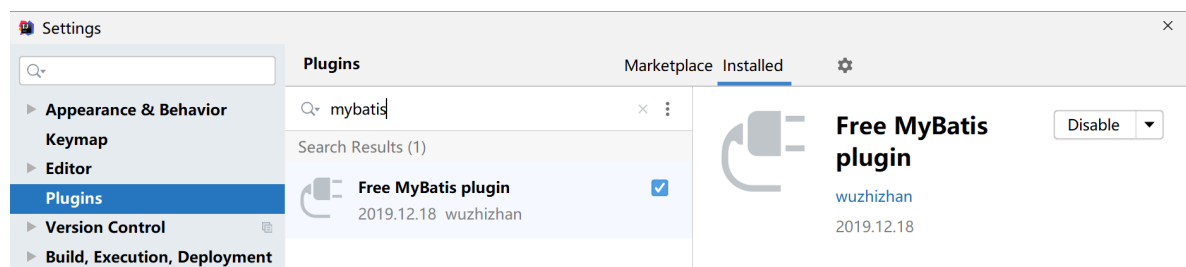
## 1. 安装Lombok插件

Lombok插件可以在编译代码的时候，为封装类生成set/get方法，这可以让封装类的代码保持工整和简洁



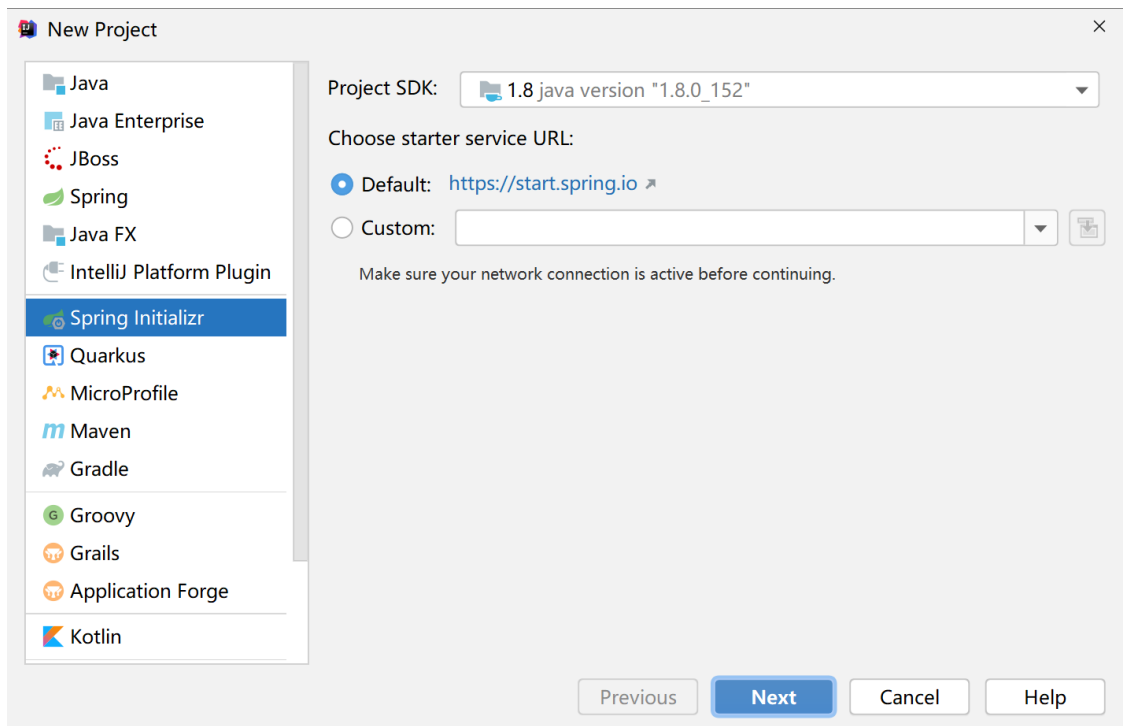
## 2. 安装Free MyBatis plugin插件

这个插件可以根据数据表自动生成DAO和XML文件，提高工作效率。另外，它还可以把XML文件和DAO关联在一起，你写代码的时候，点击DAO方法前面的图标，就会自动跳转到这个DAO方法对应的XML标签。

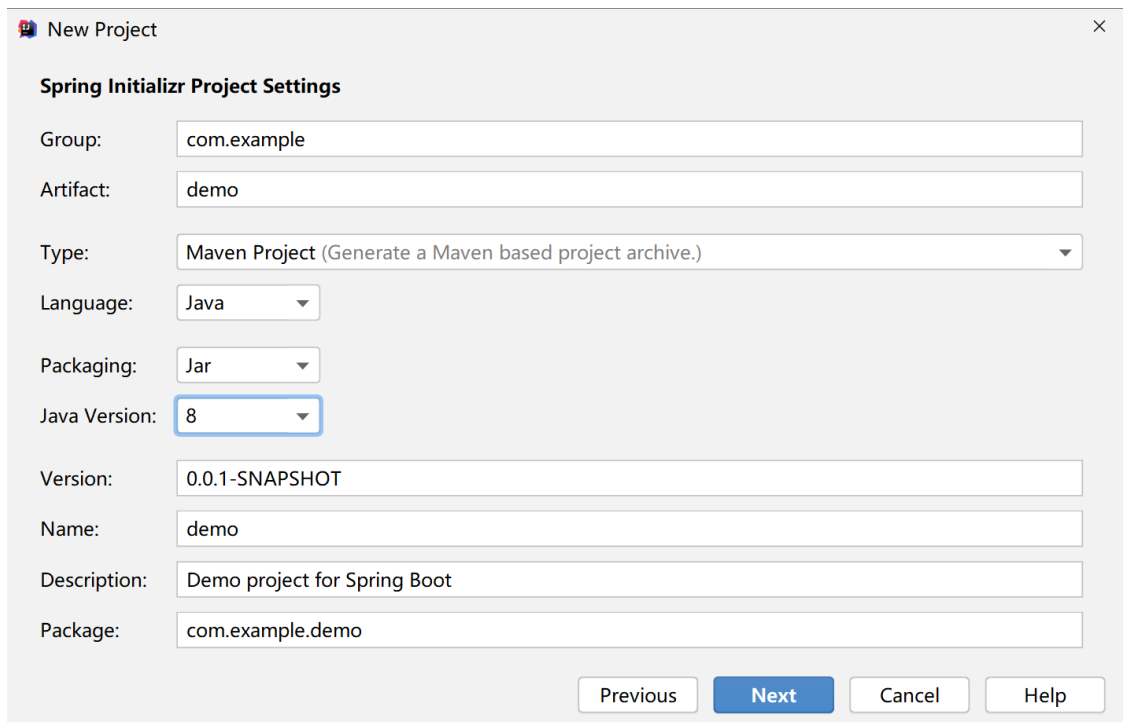


# 二、创建SpringBoot项目

## 1. 利用图形面板创建SpringBoot项目



## 2. JDK版本选择8



## 3. 勾选Lombok、SpringWeb、MySQL Driver, 以及MyBatis Framework

New Project

Dependencies

Spring Boot

2.3.3

Developer Tools

Web

Template Engines

Security

SQL

NoSQL

Messaging

I/O

Ops

Observability

Testing

Spring Cloud

Spring Cloud Security

Spring Cloud Tools

Spring Cloud Config

Spring Cloud Discovery

Spring Cloud Routing

☐ Spring Boot DevTools

☒ Lombok

☐ Spring Configuration Processor

Selected Dependencies

Developer Tools

Lombok

Web

Spring Web

SQL

MyBatis Framework

MySQL Driver

Previous

Next

Cancel

Help

New Project

Dependencies

Spring Boot

2.3.3

Developer Tools

Web

Template Engines

Security

SQL

NoSQL

Messaging

I/O

Ops

Observability

Testing

Spring Cloud

Spring Cloud Security

Spring Cloud Tools

Spring Cloud Config

Spring Cloud Discovery

Spring Cloud Routing

☒ Spring Web

☐ Spring Reactive Web

☐ Rest Repositories

☐ Spring Session

☐ Rest Repositories HAL Explorer

☐ Rest Repositories HAL Browser

☐ Spring HATEOAS

☐ Spring Web Services

☐ Jersey

☐ Vaadin

Selected Dependencies

Developer Tools

Lombok

Web

Spring Web

SQL

MyBatis Framework

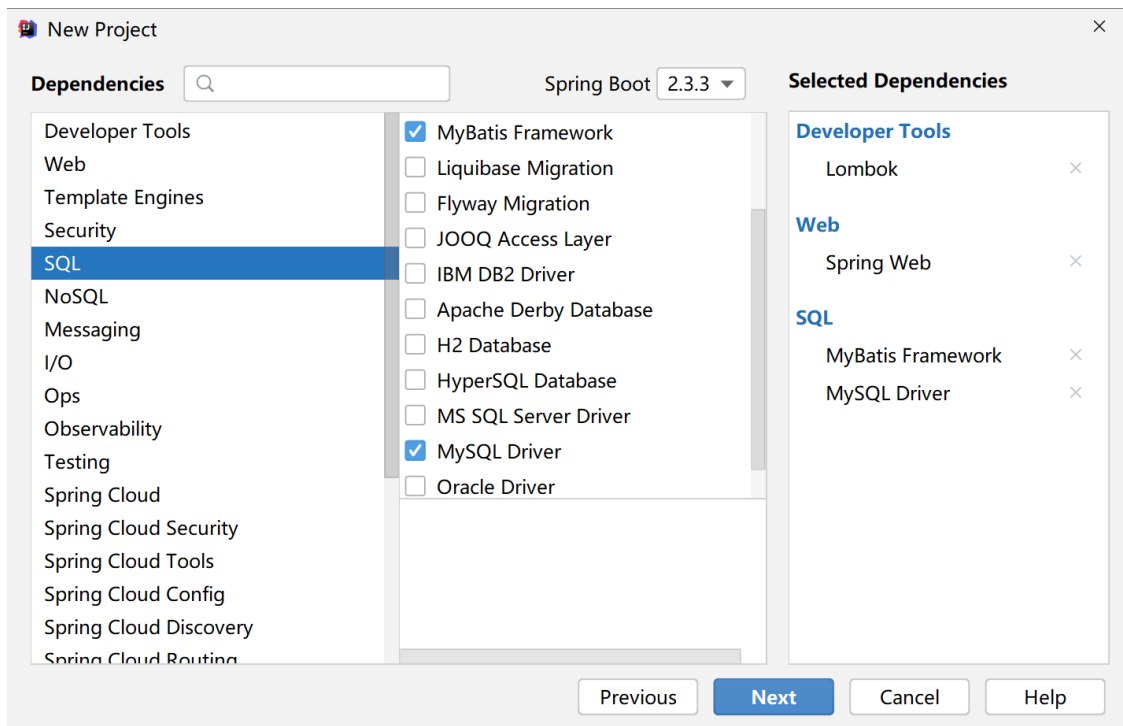
MySQL Driver

Previous

Next

Cancel

Help



#### 4. 在pom.xml文件中加入Druid数据库连接池的依赖

```
<dependency>
  <groupId>com.alibaba</groupId>
  <artifactId>druid-spring-boot-starter</artifactId>
  <version>1.1.23</version>
</dependency>
```

#### 5. 把application.properties文件的后缀名改成yml，然后添加如下内容：

```
server:
  tomcat:
    uri-encoding: UTF-8
    threads:
      max: 200
      min-spare: 30
    connection-timeout: 5000ms
  port: 8080
  servlet:
    context-path: /demo

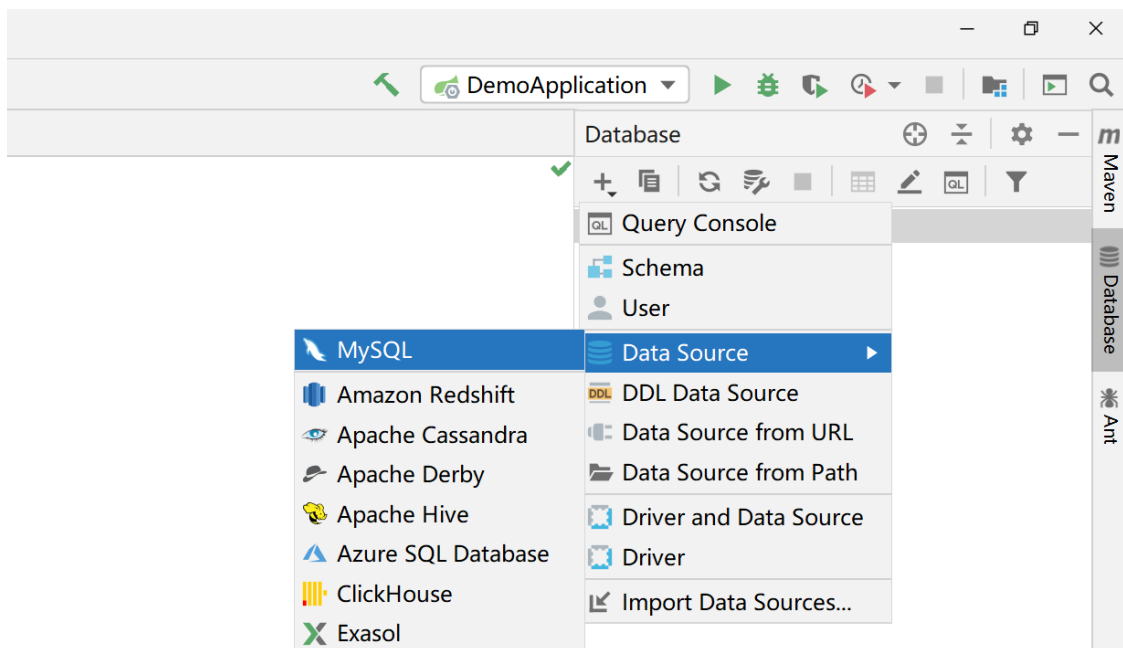
spring:
  datasource:
    type: com.alibaba.druid.pool.DruidDataSource
  druid:
    driver-class-name: com.mysql.cj.jdbc.Driver
    #你数据库的URL
```

```
url: jdbc:mysql://localhost:3306/demo?
useUnicode=true&characterEncoding=UTF-8&serverTimezone=Asia/Shanghai
#你数据库的用户名
username: root
#你数据的密码
password: abc123456
initial-size: 5
max-active: 10
min-idle: 5

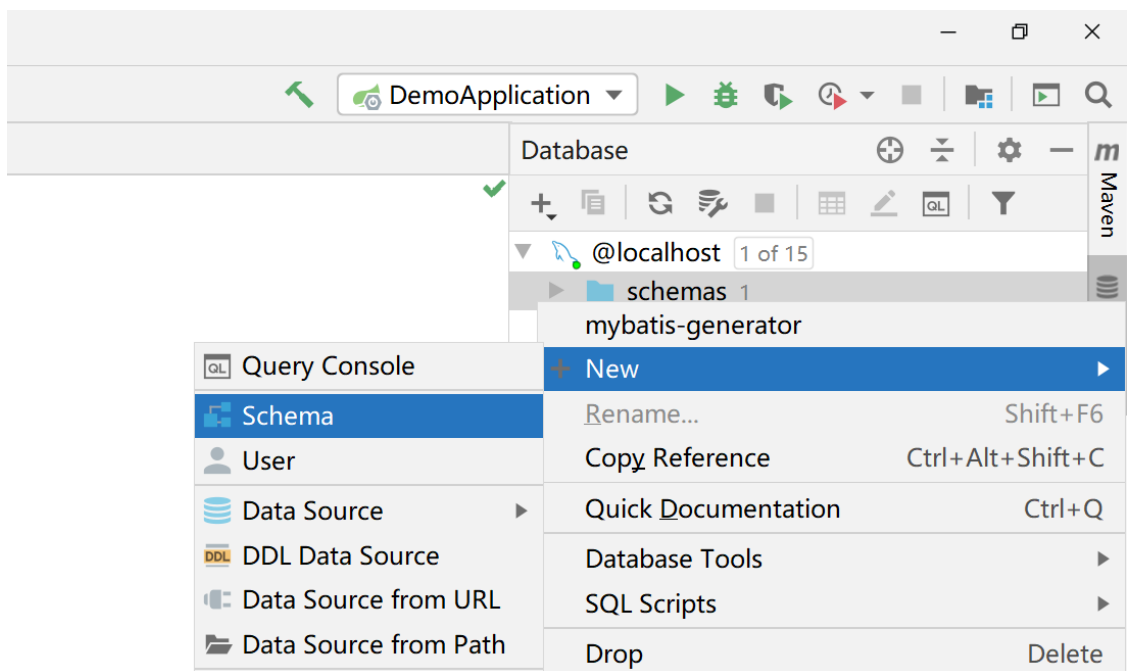
mybatis:
#XML文件的地址
mapper-locations: classpath*:mapper/*.xml
#映射类文件地址
type-aliases-package: com.example.demo.pojo
configuration:
#支持驼峰命名法（很重要）
map-underscore-to-camel-case: true
```

## 三、连接数据库

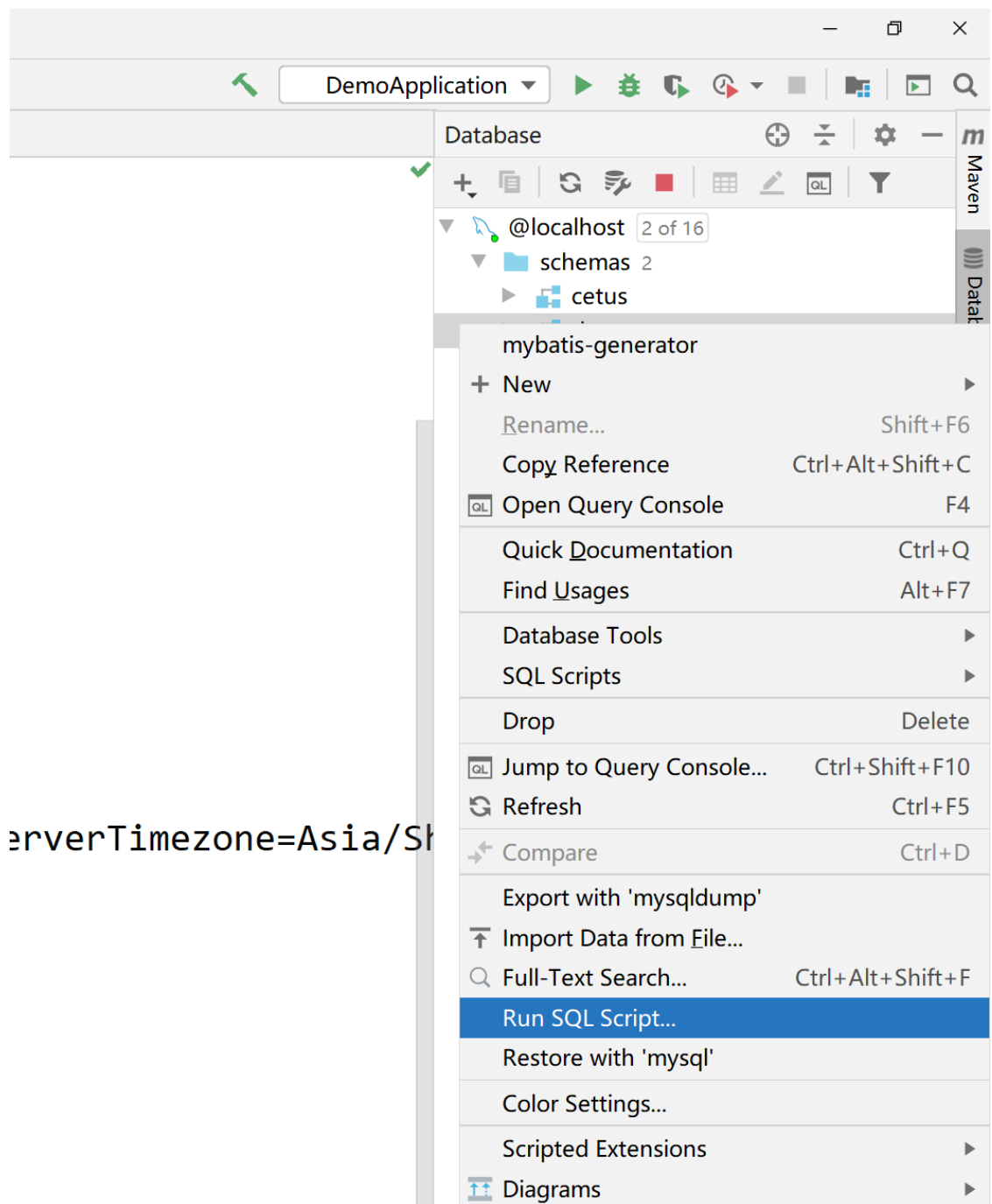
1. 在IDEA工具上面，创建MySQL数据库连接，如下：



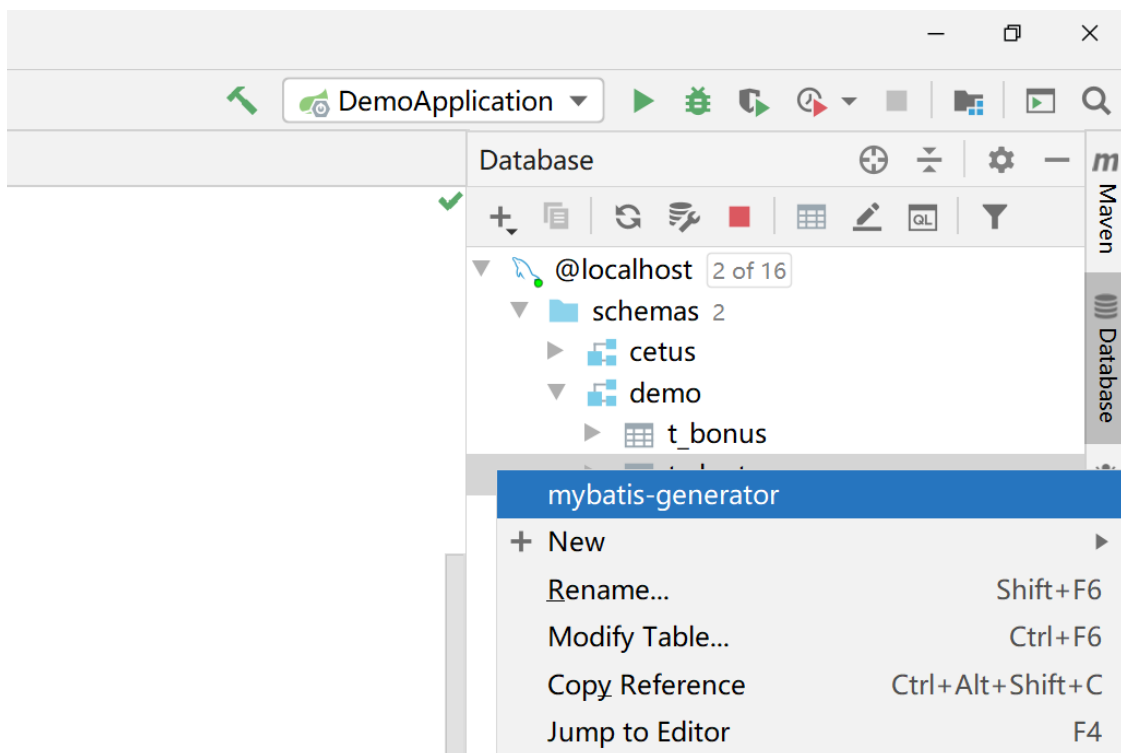
2. 建立连接之后，创建demo逻辑库



3. 在demo逻辑库上鼠标右键选择运行demo.sql文件，导入数据表



4. 选中t\_dept表，鼠标右键，选择mybatis-generator



5. 在图形面板中修改映射类名字为Dept, 路径为com.example.demo.pojo。DAO接口名字为DeptDao, 路径为com.example.demo.dao。XML文件目录为mapper, 生成代码

6. 给DeptDao接口加上@Mapper注解

```
@Mapper
public interface DeptDao {
    .....
}
```

7. 创建service文件夹, 编写DeptService.java类

```
package com.example.demo.service;

import com.example.demo.dao.DeptDao;
```



```

import com.example.demo.pojo.Dept;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

@Service
public class DeptService {
    @Autowired
    private DeptDao deptDao;

    public Dept selectByPrimaryKey(Integer deptno){
        Dept dept=deptDao.selectByPrimaryKey(deptno);
        return dept;
    }
}

```

#### 8. 创建controller文件夹，编写DeptController.java类

```

package com.example.demo.controller;

import com.example.demo.pojo.Dept;
import com.example.demo.service.DeptService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
@RequestMapping("/dept")
public class DeptController {
    @Autowired
    private DeptService deptService;

    @GetMapping("/selectByPrimaryKey")
    public Dept selectByPrimaryKey(Integer deptno){
        Dept dept=deptService.selectByPrimaryKey(deptno);
        return dept;
    }
}

```

#### 9. 运行DemoApplication类

#### 10. 打开浏览器访问 <http://127.0.0.1:8080/demo/dept/selectByPrimaryKey?deptno=10>

浏览器会出现根据10编号查询出来的部门数据

```
{"deptno":10,"dname":"ACCOUNTING","loc":"NEW YORK"}
```

