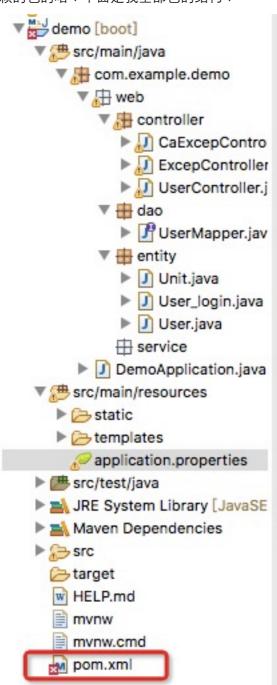
Springboot——连接阿里云 mysql

-----20190731 花名: 韵方

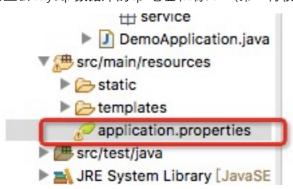
之前如何通过 maven 构建基础的 spring 框架我就不讲了,大家可以自行百度谷歌哈。

如何部署阿里云的 mysql 我也不讲了,大家懂的哈。

直接切入正题好了,首先进行 maven 的依赖的部署,编写以下的代码,他会自动去下载依赖的包的哈:下面是我全部包的结构:



然后是配置数据库连接的代码,连接阿里云数据库就是将原先放 location 的地方替换程阿里云 mysql 数据库的 ip 地址和端口:(第一行模糊的地方放地址,第三行放密码)



```
spring.datasource.url=jdbc:mysql:// :3306/demo_learn
spring.datasource.username=root
spring.datasource.password=i=i=i
spring.datasource.driver-class-name=com.mysql.jdbc.Driver
```

随后创建相关的表, 我的表结构如下:



内容如下:



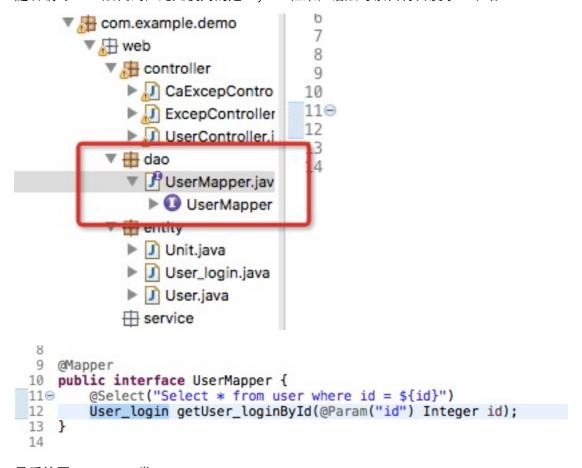
随后编写持久化层代码(可以自动生成的哈):

```
- Occumphenia.
                                                                           1

▼ 
 entity

                                                                           1
                   ▶ J Unit.java
                                                                           1
                         J User_login.java
                                                                           1
                        J User.java
                                                                           2
                 service
   public class User_login {
   private int id;
   private String name;
   private String key;
   public int getId() {
      return id;
}
         public void setId(int id) {
    this.id = id;
public String getName() {
    return name;
         public void setName(String name) {
   this.name = name;
         public String getKey() {
   return key;
         public void setKey(String key) {
   this.key = key;
         public String toString() {
   return "User_login [id=" + id + ", name=" + name + ", key=" + key + "]";
```

随后编写 Dao 层代码,此处使用的是 mybits 框架,语法可以自行百度学一下哈:



最后编写 Controller 类:

```
▼ 
 web

     ▼ Æ controller
        ▶ □ CaExcepContro
      ▶ J ExcepController
        UserController.j

▼ 

dao

        UserMapper.jav
          UserMapper
     entity
        ▶ J Unit.java
        ▶ J User_login.java
   @RestController
   public class ExcepController {
17
180
       @Autowired
        private UserMapper userMapper;
19
20
210
22
23
24
25
26
       @RequestMapping(value="/getuser")
270
        public Object getUser(Integer id) {
28
            return userMapper.getUser_loginById(id);
29
        }
30
31
到这里所有的工作就完成了,打开 springboot 运行 location / getuser?id=1 就可以了
展示的结果如下:
               (i) localhost:8080/getuser?id=1
  {"id":1, "name": "dominic", "key": "bzf19970903"}
```

遇到错误的话看报什么错一一进行排查就好了,最容错的是 sql 编写那块。