

Quiz后端-4、实现删除用户接口

日期：2025.07.19 作者：tfzhang

后端主要提供用户与题目的注册、查询、删除等操作。使用到的开发工具：

- IDEA2024; 关注公众号”青椒工具”，发送”IDEA”，获取windows下的IDEA安装包
- mysql 5.7; 关注公众号”青椒工具”，发送”mysql”，获取windows下的mysql5.7安装包；

1、前情回顾：

- 1、Java访问数据库：MyBatis框架；
- 2、浏览器访问Java：Controller；
- 3、实现插入新用户接口；

2、上次视频遗留的查询用户；

步骤1：查询用户的UserMapper端代码

```
1 @Select("SELECT COUNT(*) FROM user WHERE userName = #{username} AND isDelete = 0")
2 public int existsByName(@Param("username") String username);
```

按照用户名查询，用户是否存在，如果用户存在，则返回1；否则返回0；

步骤2：测试上述的方法

步骤3：在UserService的实现类中，根据上述代码，进行完善；

3、实现根据用户Id或者用户名，删除用户：

步骤1：在usermapper中添加方法

在UserMapper中添加如下的代码：

```
1 //根据id删除用户;
2 @Update("UPDATE user SET isDelete = 1, updateTime = NOW() WHERE id = #{id} AND isDelete = 0")
3 public int deleteUserById(@Param("id") Long id);
4
5 //根据username删除用户;
6 @Update("UPDATE user SET isDelete = 1, updateTime = NOW() WHERE userName = #{username} AND isDelete = 0")
7 public int deleteByUsername(@Param("username") String username);
```

步骤2：测试usermapper方法

测试上述的UserMapper中的两个删除方法；

步骤3：在UserService中添加下面的接口和实现

```
1 //接口:  
2 public boolean deleteUserById(Long id);  
3 public boolean deleteUserByName(String username);  
4  
5 //实现:  
6 public boolean deleteUserById(Long id) {  
7     int result = userMapper.deleteUserById(id);  
8     return result > 0;  
9 }  
10  
11 public boolean deleteByName(String username) {  
12     int result = userMapper.deleteUserByUsername(username);  
13     return result > 0;  
14 }
```

步骤4：controller中添加代码

```
1 @GetMapping("/deleteById")  
2 public Result deleteUserById(Long id) {  
3     boolean success = userService.deleteUserById(id);  
4     if (success) {  
5         return Result.success("用户已删除");  
6     }  
7     return Result.error("用户不存在或已被删除");  
8 }  
9  
10 @GetMapping("/deleteByName")  
11 public Result deleteUser(String username) {  
12     boolean success = userService.deleteUserByName(username);  
13     if (success) {  
14         return Result.success("用户已删除");  
15     }  
16     return Result.error("用户不存在或已被删除");  
17 }
```

步骤5：apifox测试上述的delete接口

作业：书写hard delete的接口