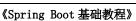


作者:冯永伟



第14节

使用 SQL 关系型数据库-spring-data-jpa

一、配置数据源:

二、Jpa 配置

```
# JPA
spring. jpa. hibernate. ddl-auto= update
#显示 sql 语句
spring. jpa. show-sql=true
```

三、实体类

```
@Entity
public class RoncooUserLog {
    @Id
    @GeneratedValue
    private Integer id;

    @Column
    private Date createTime;

    @Column
    private String userName;

    @Column
    private String userIp;

public Integer getId() {
        return id;
        ...
}
```





作者: 冯永伟



```
public void setId(Integer id) {
                                            this.id = id;
                      public Date getCreateTime() {
                                            return createTime;
                       public void setCreateTime(Date createTime) {
                                            this.createTime = createTime;
                       public String getUserName() {
                                            return userName;
                       public void setUserName(String userName) {
                                            this.userName = userName;
                      public String getUserIp() {
                                            return userIp;
                      public void setUserIp(String userIp) {
                                           this.userIp = userIp;
                       @Override
                       public String toString() {
                                           \label{local_control_control} \mbox{return "RoncooUserLog [id=" + id + ", createTime=" + createTime + ", userName=" - id + "]} \mbox{ } \mbox{$ = 0$} \mbo
+ userName + ", userIp=" + userIp + "]";
```

四、定义接口(继承 JpaRepository)

```
public\ interface\ RoncooUserLogDao\ extends\ JpaRepository < RoncooUserLog,\ Integer> \{\ \}
```

五、测试

@Autowired

private RoncooUserLogDao roncooUserLogDao;





作者: 冯永伟



```
@Test
public void insert() {
    RoncooUserLog entity = new RoncooUserLog();
    entity.setUserName("无境");
    entity.setUserIp("192.168.0.1");
    entity.setCreateTime(new Date());
    roncooUserLogDao.save(entity);
@Test
public void delete() {
    roncooUserLogDao.delete(1);
@Test
public void update() {
    RoncooUserLog entity = new RoncooUserLog();
    entity.setId(2);
    entity.setUserName("无境2");
    entity.setUserIp("192.168.0.1");
    entity.setCreateTime(new Date());
    roncooUserLogDao.save(entity);
@Test
public void select() {
    RoncooUserLog result = roncooUserLogDao.findOne(1);
    System.out.println(result);
```

六、扩展进行自定义的查询

1. 使用内置的关键词查询

http://docs.spring.io/spring-data/jpa/docs/1.10.2.RELEASE/reference/html/

2. 使用自定义语句查询

各种语法, 具体查看

http://docs.spring.io/spring-data/jpa/docs/1.10.2.RELEASE/reference/html/

这个优先级比上面的高

@Query(value = "select u from RoncooUserLog u where u.userName=?1")

测试

@Test





作者: 冯永伟



```
public void select2() {
      List<RoncooUserLog> result =
roncooUserLogDao.findByUserName("无境");
      System.out.println(result);
@Test
   public void select3() {
      List<RoncooUserLog> result =
roncooUserLogDao.findByUserNameAndUserIp("无境",
"192.168.0.1");
      System.out.println(result);
// 分页
   @Test
   public void queryForPage() {
      Pageable pageable = new PageRequest(0, 20, new Sort(new
Order(Direction. DESC, "id")));
      Page<RoncooUserLog> result =
roncooUserLogDao.findByUserName("无境", pageable);
      System.out.println(result.getContent());
      }
```

更多课程信息,请关注 <mark>龙果学院</mark> 官方网站 <u>http://www.roncoo.com/</u>或关注 <mark>龙果</mark> 微信公众号 <u>RonCoo_com</u>



