# 4-7 封装检测当天是否可以签到(持久层)

140.143.132.225:8000/project-1/doc-55

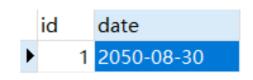
## 一、为什么要检测是否可以签到?

上节课我们通过分装Emos系统常量信息,从而得知考勤是分为起止时间的。在考勤开始之前,用户是不能考勤签到的。同理,在当天考勤结束之后,用户也是不能考勤签到的。甚至节假日也不能考勤,只有正常的工作日才能考勤签到。

### 怎么判断当天是工作日还是节假日?

在数据库中有 tb\_workday 和 tb\_holidays 两张数据表,记录着哪天是工作日,哪天是休息日。

Emos系统默认周一至周五为工作日,周六周日为休息日。 但是这两张表不是把所有的工作日和休息日都记录下来,只 是记录比特殊的工作日或者休息日。比如说今年的中秋节赶 上了礼拜四,于是就把周五和周六设置成休息日,跟中秋节 连成三连休,然后周日正常上班。这种特殊情况我们就要记



录下来。在 tb\_workday 记录周日是工作日,在 tb\_holidays 表中记录周五那天是休息日。这样 Emos系统在中秋三连休期间不会执行考勤签到。

## 二、封装持久层代码

# 查询特殊休息日

在 TbHolidaysDao.xml 文件中,添加查询语句

- 1. <select id="searchTodayIsHolidays" resultType="Integer">
- SELECT id FROM tb\_holidays WHERE date=CURRENT\_DATE LIMIT 1;
- 3. </select>

### 在 TbHolidaysDao.java 文件中,添加抽象方法

- 1. @Mapper
- 2. public interface TbHolidaysDao {
- public Integer searchTodayIsHolidays();
- 4. }

# 查询特殊工作日

#### 在 TbWorkdayDao.xml 文件中,添加查询语句

- 1. <select id="searchTodayIsWorkday" resultType="Integer">
- SELECT id FROM tb workday WHERE date=CURRENT DATE LIMIT 1;
- 3. </select>

#### 在 TbWorkdayDao.java 文件中,添加抽象方法

- @Mapper
  public interface TbWorkdayDao {
  public Integer searchTodayIsWorkday();
  }
- 查询当天是否已经签到

tb\_checkin 表结构如下,Emos不仅要记录考勤的人员、时间,还要记录考勤人的地理坐标,然后根据疫情实时信息,判定用户所处的地区是新冠疫情的高危地区还是低风险地区。这部分功能,后续的章节我们再实现。

名	类型	长度	小数点	不是 null	虚拟	键	注释
id	int	10	0	$\checkmark$		<b>P</b> 1	主键
user_id	int	10	0	$\checkmark$			用户ID
address	varchar	255	0				签到地址
country	varchar	255	0				国家
province	varchar	255	0				省份
city	varchar	255	0				城市
district	varchar	255	0 1				区划
status	tinyint	3	0, M	1 D			考勤结果
risk	int	255	0				风险等级
date	date	0 秋	0				签到日期
create_time	datetime	0	0				签到时间

在 TbCheckinDao.xml 文件中,添加查询语句

- 1. <select id="haveCheckin" parameterType="HashMap" resultType="Integer">
- 2. SELECT id
- FROM tb\_checkin
- 4. WHERE user\_id = #{userId} AND date = CURRENT\_DATE
- 5. AND create\_time BETWEEN #{start} AND #{end}
- 6. LIMIT 1;
- 7. </select>

在 TbCheckinDao.java 文件中,添加抽象方法

- 1. @Mapper
- 2. public interface TbCheckinDao {
- public Integer haveCheckin(HashMap param);
- 4. }