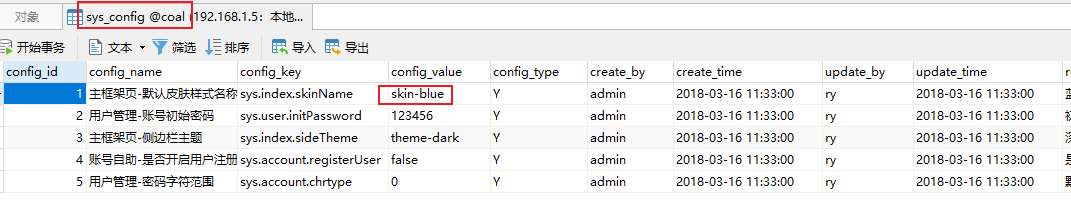
1,主页皮肤在数据库里指定

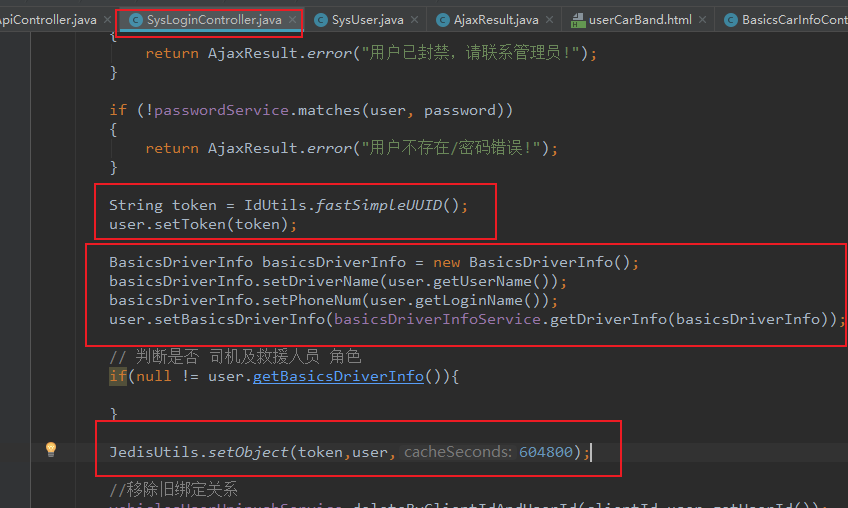


1. app主页的数量从原表或二次表里获取，有的从redis里拿
2. SysLoginController.java里有不同客户端登录的接口，注意在地址栏区分 ../appLogin或../carLogin
3. sys-user表里有账户密码，用户密码都是123456
4. 查询的数据放到AjaxResult.success(...)方法里返回

例：

AjaxResult.success("查询成功",vehiclesAttendanceService.getDriverNum(vehiclesAttendance));

6，app端.../appLogin用户登录是，后台对应接口内创建了一个mpToken放到了redis里，作为方位接口的验证



7, 使用RuoYi框架自带的分页的话

|  |
| --- |
| List集合的泛型必须继承BaseEntity, 不能放Map<String, String>  List<BasicsLocation> basicsLocationList = iBasicsLocationService.showAllCars(bcr);  return AjaxResult.success(getDataTable(basicsLocationList)); |

8, 去掉返回前端的null属性或用不到的false等默认属性的办法

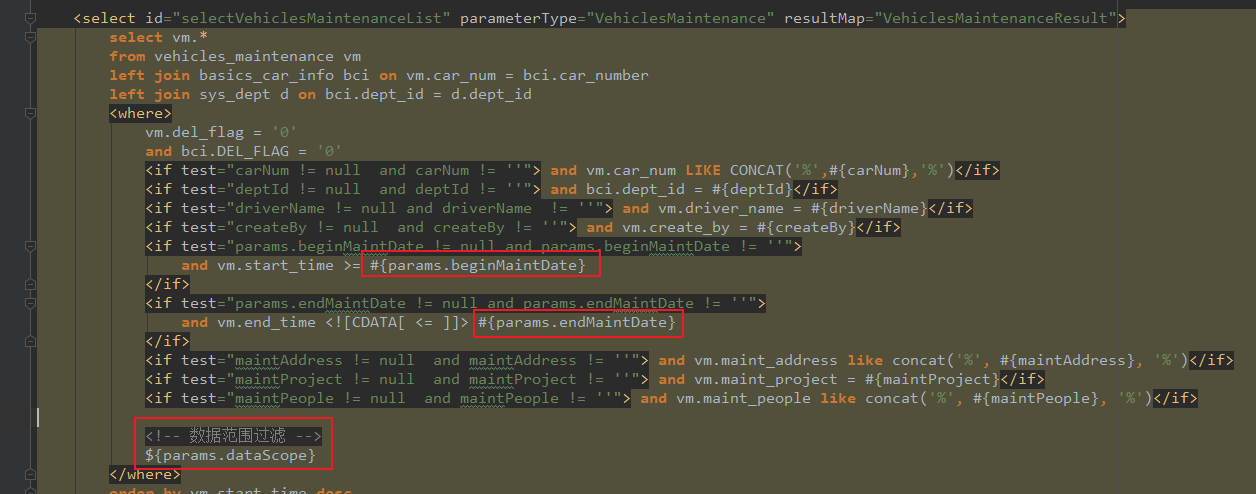
|  |
| --- |
| 在实体类上加@JsonInclude注解，去掉默认值，@JsonInclude还有其它属性值，多看看  @JsonInclude(JsonInclude.Include.NON\_DEFAULT) public class BasicsLocation extends BaseEntity { |

9, vehicles\_attendance表里有司机姓名和车牌号的对应关系

10, 前端传来的日期查询条件放到了实体类的Map<String, Object> params里了

|  |
| --- |
| /\*\*  \* 查询车队管理维修保养列表  \*/ @RequiresPermissions("system:maintenance:list") @PostMapping("/list") @ResponseBody public TableDataInfo list(VehiclesMaintenance vehiclesMaintenance) {  if("true".equals(vehiclesMaintenance.getIsParent())){  vehiclesMaintenance.setDeptId(null);  }  startPage();  List<VehiclesMaintenance> list = vehiclesMaintenanceService.selectVehiclesMaintenanceList(vehiclesMaintenance);  return getDataTable(list); } |

在sql语句中直接取出，还有${params.dataScope}，也是前端传入的



**10.2, [查询的时候@DataScope放到service层的方法上，可给${params.dataScope}赋值](mailto:查询的时候@DataScope放到service层的方法上，可给${params.dataScope}赋值)**

|  |
| --- |
| public class BasicsCarInfoServiceImpl implements IBasicsCarInfoService {  @Override @DataScope(deptAlias = "d") public List<BasicsCarInfo> getVehicles(BasicsCarInfo basicsCarInfo) {  return basicsCarInfoMapper.getVehicles(basicsCarInfo); } |

这样不用给登录用户设置deptId了，后台自动给了

|  |
| --- |
| //basicsCarInfo.setDeptId(currentUser.getDeptId().toString()); 不用写了 |

1. 本月车辆状态，Web首页查询条件只是 有开始时间，lmt.start\_time >=当月第一天，没有end\_time，所以数量和app不一致

Web端有每天对应的运单数，可用此接口显示日历天数

12, 实时油耗从f3\_location中的fuel\_1字段取值准确。

13, stateDayList接口点击日期显示当天的运输的车辆状态列表



点击运输车辆显示当天运输的车辆列表，点击休整则显示休整

思路：

(1)先查出运输车辆的车牌号

(2)休整车辆需计算排除运输的车辆

(3)注意列表数量与日历展示的数量一致，可从dayCarState接口中存好今天的车牌号，这样可保持数量一致

\* 日历页面数量和自己单独sql查询结果不一致原因：

sql中语句“ lmt.start\_time <= concat('2021-12-15', ' 23:59:59' ) and lmt.end\_time IS NULL”有误，把所有结束时间为空且时间小于今天的算了进来，

而“dayCarState”接口中只计算开始时间与当天相同且结束为空的车

14, Web找到账户下的所有车辆当dept\_id为null时，不是靠dept\_id, 而是以下语句的

* find\_in\_set函数起作用

|  |
| --- |
| -- 靠@DataScope注解，#{params.dataScope}拼接的语句  and SELECT dept\_id FROM sys\_dept WHERE dept\_id = 116 OR find\_in\_set( 116, ancestors ) ) |

见DAO层的getVehicles(BasicsCarInfo basicsCarInfo)方法

* MyBatis中的以下语句也是相同作用

|  |
| --- |
| -- 纯手写的语句：  <if test="deptId != null and deptId != ''"> and (a.dept\_id = #{deptId} or d.parent\_id = #{deptId} OR d.ancestors LIKE CONCAT('%,', #{deptId}, ',%'))</if> |