金坤科创智慧牧场之牧场主手机客户端V1.0

源代码

北京金坤科创技术有限公司

2018-05-10

前30页

**package** com.jinkun\_innovation.pastureland.common;  
  
  
*/\*\*  
 \* Created by Guan on 2018/3/13.  
 \*/***public class** Constants {  
  
 *//  
// public static final String BASE\_URL = "http://192.168.50.215:8080";  
  
// public static final String IP = "222.249.165.94";  
// public static final String PORT = ":10100";  
  
  
// public static final String IP = "192.168.50.215";  
// public static final String PORT = ":8080";* **public static final** String ***IP*** = **"182.92.3.109"**;  
 **public static final** String ***PORT*** = **":10100"**;  
  
*// public static final String IP = "116.95.255.211";  
// public static final String PORT = ":10100";  
  
  
 //测试服务器* **public static final** String ***BASE\_URL*** = **"http://"** + ***IP*** + ***PORT***;  
  
  
 *//内部正式服务器  
 //http://182.92.3.109:10100 testranch 12345678 这是109访问地址跟用户名密码  
// public static final String BASE\_URL = "http://182.92.3.109:10100";  
  
 //客户服务器  
// public static final String BASE\_URL ="http://116.95.255.211:10120";* **public static final int *ranchID*** = 1;  
  
 **public static final** String ***token*** = **"ce5cc971-5904-4454-b639-cb8d48215963"**;  
  
  
 *//牧场主找回密码  
 //1)点击发送验证码：* **public static final** String ***VERIFY\_CODE*** = ***BASE\_URL*** + **"/mobleClaim/txCode.do"**;  
  
 *//2)修改密码:* **public static final** String ***MODIFY\_PASSWORD*** = ***BASE\_URL*** + **"/ranchLogin/backPass.do"**;  
  
 *//登录:* **public static final** String ***LOGIN*** = ***BASE\_URL*** + **"/ranchLogin/login.do"**;  
  
  
 *//发布认领  
 //判断是被是否被绑定* **public static final** String ***ISDEVICEBINDED*** = ***BASE\_URL*** + **"/releaseLivestocks/isDeviceBinded.do"**;  
  
  
 *//1、查看发布情况selectLivestock.do，未发布执行release.do* **public static final** String ***SELECT\_LIVE\_STOCK*** = ***BASE\_URL*** + **"/releaseLivestocks/selectLivestock.do"**;  
  
  
 *//根据字典表类型选字典表品种(牲畜扫描时选类型时调用)* **public static final** String ***SELECTVARIETY*** = ***BASE\_URL*** + **"/releaseLivestocks/selectVariety.do"**;  
  
 *//直接发布到认领表接口* **public static final** String ***RELEASE*** = ***BASE\_URL*** + **"/releaseLivestocks/release.do"**;  
  
 *//打疫苗登记-新增牲畜* **public static final** String ***SAVELIVESTOCK*** = ***BASE\_URL*** + **"/releaseLivestocks/saveLivestock.do"**;  
  
  
 *//重新发布到牲畜认领表* **public static final** String ***IS\_CLAIMED*** = ***BASE\_URL*** + **"/releaseLivestocks/isClaimed.do"**;  
  
 *//获取已发布牲畜信息* **public static final** String ***LIVE\_STOCK\_CLAIM\_LIST*** = ***BASE\_URL*** + **"/releaseLivestocks/livestockClaimList.do"**;  
  
 *//剪毛* **public static final** String ***SHEARING*** = ***BASE\_URL*** + **"/releaseLivestocks/shearing.do"**;  
  
 *//牧场拍生活照和视频* **public static final** String ***RANCHIMGVIDEO*** = ***BASE\_URL*** + **"/releaseLivestocks/ranchImgVideo.do"**;  
  
 *//牧场详情* **public static final** String ***RANCH*** = ***BASE\_URL*** + **"/releaseLivestocks/ranch.do"**;  
  
 *//更新牧场* **public static final** String ***UPDRANCH*** = ***BASE\_URL*** + **"/releaseLivestocks/updRanch.do"**;  
  
 *//牲畜详情* **public static final** String ***LIVESTOCK*** = ***BASE\_URL*** + **"/releaseLivestocks/livestock.do"**;  
  
 *//设备消息* **public static final** String ***DEVICEMSG*** = ***BASE\_URL*** + **"/releaseLivestocks/deviceMsg.do"**;  
  
 *//牧场主处理拍照和视频请求* **public static final** String ***updLivestockClaim*** = ***BASE\_URL*** + **"/releaseLivestocks/updLivestockClaim.do"**;  
  
  
 *//设备解绑（解绑之前要弹出确认提示框）* **public static final** String ***DOUNBINDDEVICE*** = ***BASE\_URL*** + **"/releaseLivestocks/doUnBindDevice.do"**;  
  
 *//个人信息  
 //1）查询个人信息* **public static final** String ***ADMINLIST*** = ***BASE\_URL*** + **"/adminMember/adminList.do"**;  
  
 *//2）更新个人信息* **public static final** String ***UPDADMIN*** = ***BASE\_URL*** + **"/adminMember/updAdmin.do"**;  
  
 *//文件上传（图片、视频） /adminMember/headImgUrl.do* **public static final** String ***HEADIMGURL*** = ***BASE\_URL*** + **"/adminMember/headImgUrl.do"**;  
  
 *//获取牧场牲畜类型和每种类型牲畜的数量* **public static final** String ***QUERYTYPEANDSUM*** = ***BASE\_URL*** + **"/releaseLivestocks/queryTypeAndSum.do"**;  
  
 *//通过牲畜类型查询所有牲畜* **public static final** String ***QUERYLIVESTOCKVARIETYLIST*** = ***BASE\_URL*** + **"/releaseLivestocks/queryLivestockVarietyList.do"**;  
  
 *//电子档案* **public static final** String ***DianziDangan*** = ***BASE\_URL*** + **"/claimLivestock/getArchivesElectronicURL.do"**;  
  
 *//牧场主确认订单支付* **public static final** String ***orderPay*** = ***BASE\_URL*** + **"/releaseLivestocks/orderPay.do"**;  
  
 *//牲畜删除* **public static final** String ***delLivestock*** = ***BASE\_URL*** + **"/releaseLivestocks/delLivestock.do"**;  
  
  
 *//跳转WEB首页* **public static final** String ***homeWeb*** = ***BASE\_URL*** + **"/releaseLivestocks/homeWeb.do"**;  
  
 *//牲畜定位* **public static final** String ***queryLivestockList*** = ***BASE\_URL*** + **"/releaseLivestocks/queryLivestockList.do"**;  
  
 *//农用工具  
 //添加* **public static final** String ***saveTool*** = ***BASE\_URL*** + **"/releaseLivestocks/saveTool.do"**;  
 *//更新* **public static final** String ***updTool*** = ***BASE\_URL*** + **"/releaseLivestocks/updTool.do"**;  
 *//删除* **public static final** String ***delTool*** = ***BASE\_URL*** + **"/releaseLivestocks/delTool.do"**;  
  
  
  
  
}

**package** com.jinkun\_innovation.pastureland.ui;  
  
**import** android.Manifest;  
**import** android.content.Intent;  
**import** android.content.pm.PackageManager;  
**import** android.graphics.Color;  
**import** android.os.Build;  
**import** android.os.Bundle;  
**import** android.support.annotation.NonNull;  
**import** android.support.annotation.Nullable;  
**import** android.support.annotation.RequiresApi;  
**import** android.support.design.widget.TextInputEditText;  
**import** android.support.v4.app.ActivityCompat;  
**import** android.support.v4.content.ContextCompat;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.text.TextUtils;  
**import** android.util.Log;  
**import** android.view.View;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.common.Constants;  
**import** com.jinkun\_innovation.pastureland.utilcode.SpUtil;  
**import** com.jinkun\_innovation.pastureland.utilcode.util.AppUtils;  
**import** com.jinkun\_innovation.pastureland.utilcode.util.ToastUtils;  
**import** com.jinkun\_innovation.pastureland.utils.PhoneFormatCheckUtils;  
**import** com.jinkun\_innovation.pastureland.utils.PrefUtils;  
**import** com.lzy.okgo.OkGo;  
**import** com.lzy.okgo.callback.StringCallback;  
**import** com.lzy.okgo.model.Response;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
  
**import** butterknife.BindView;  
**import** butterknife.ButterKnife;  
**import** butterknife.OnClick;  
**import** cn.jpush.android.api.JPushInterface;  
**import** cn.pedant.SweetAlert.SweetAlertDialog;  
  
*/\*\*  
 \* Created by Guan on 2018/3/19.  
 \*/***public class** LoginActivity1 **extends** AppCompatActivity {  
  
 **private static final** String ***TAG1*** = LoginActivity1.**class**.getSimpleName();  
  
 @BindView(R.id.***tieAccount***)  
 TextInputEditText **mTieAccount**;  
 @BindView(R.id.***tiePwd***)  
 TextInputEditText **mTiePwd**;  
 @BindView(R.id.***tvForgetPwd***)  
 TextView **mTvForgetPwd**;  
 @BindView(R.id.***btnLogin***)  
 ImageView **mBtnLogin**;  
  
 **private** SweetAlertDialog **mPDialog**;  
 **private** String **mUsername**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_login1***);  
 ButterKnife.*bind*(**this**);  
  
  
 **if** (Build.VERSION.***SDK\_INT*** >= Build.VERSION\_CODES.***M***) {  
 checkRuntimePermissions();  
 }  
  
 TextView tvVersion = (TextView) findViewById(R.id.***tvVersion***);  
 tvVersion.setText(**"版本号："** + AppUtils.*getAppVersionCode*() + **"，IP="** + Constants.***IP***);  
  
  
 **mUsername** = PrefUtils.*getString*(getApplicationContext(), **"username"**, **null**);  
 **if** (!TextUtils.*isEmpty*(**mUsername**)) {  
 **mTieAccount**.setText(**mUsername**);  
 }  
  
  
 }  
  
 @RequiresApi(api = Build.VERSION\_CODES.***M***)  
 **private void** checkRuntimePermissions() {  
 List<String> permissions = **new** ArrayList<>();  
 **if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***CAMERA***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***CAMERA***);  
 }  
 **if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***WRITE\_EXTERNAL\_STORAGE***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***WRITE\_EXTERNAL\_STORAGE***);  
 }  
 **if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***READ\_EXTERNAL\_STORAGE***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***READ\_EXTERNAL\_STORAGE***);  
 }  
 **if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***RECORD\_AUDIO***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***RECORD\_AUDIO***);  
 }  
 **if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***ACCESS\_FINE\_LOCATION***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_FINE\_LOCATION***);  
 }  
  
 **if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***ACCESS\_COARSE\_LOCATION***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }  
  
 **if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***WAKE\_LOCK***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }**if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***READ\_PHONE\_STATE***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }**if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***WRITE\_EXTERNAL\_STORAGE***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }**if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***READ\_EXTERNAL\_STORAGE***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }**if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***WRITE\_SETTINGS***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }**if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***VIBRATE***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }**if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***MOUNT\_UNMOUNT\_FILESYSTEMS***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }**if** (ContextCompat.*checkSelfPermission*(**this**, Manifest.permission.***ACCESS\_NETWORK\_STATE***) != PackageManager.***PERMISSION\_GRANTED***) {  
 permissions.add(Manifest.permission.***ACCESS\_COARSE\_LOCATION***);  
 }  
  
  
  
  
 **if** (permissions.size() > 0) {  
 ActivityCompat.*requestPermissions*(LoginActivity1.**this**, permissions.toArray(**new** String[permissions.size()]),  
 ***REQUEST\_CODE\_ASK\_MULTIPLE\_PERMISSIONS***);  
 } **else** {  
*// startScanActivity();* }  
  
 }  
  
 @Override  
 **public void** onRequestPermissionsResult(**int** requestCode, @NonNull String[] permissions, @NonNull **int**[] grantResults) {  
 **switch** (requestCode) {  
 **case *REQUEST\_CODE\_ASK\_MULTIPLE\_PERMISSIONS***:  
 **if** (grantResults.**length** > 0  
 && grantResults[0] == PackageManager.***PERMISSION\_GRANTED***) {  
 } **else** {  
 Toast.*makeText*(LoginActivity1.**this**, **"请手动打开摄像头权限"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **default**:  
 **break**;  
 }  
 }  
  
 **private static final int *REQUEST\_CODE\_ASK\_MULTIPLE\_PERMISSIONS*** = 3;*//权限请求* @OnClick({R.id.***tvForgetPwd***, R.id.***btnLogin***})  
 **public void** onViewClicked(View view) {  
 **switch** (view.getId()) {  
  
  
 **case** R.id.***tvForgetPwd***:  
  
 startActivity(**new** Intent(getApplicationContext(), FindPwdActivity.**class**));  
  
 **break**;  
 **case** R.id.***btnLogin***:  
  
 **final** String account = **mTieAccount**.getText().toString().trim();  
 String pwd = **mTiePwd**.getText().toString();  
  
  
 **if** (!TextUtils.*isEmpty*(account) && !TextUtils.*isEmpty*(pwd)) {  
  
  
 **if** (!TextUtils.*isEmpty*(account) && !TextUtils.*isEmpty*(pwd)) {  
  
 **if** (PhoneFormatCheckUtils.*isMobile*(account)) {  
  
 *//大陆号码，可以登录* **mPDialog** = **new** SweetAlertDialog(**this**, SweetAlertDialog.***PROGRESS\_TYPE***);  
 **mPDialog**.getProgressHelper().setBarColor(Color.*parseColor*(**"#A5DC86"**));  
 **mPDialog**.setTitleText(**"正在登录..."**);  
 **mPDialog**.setCancelable(**true**);  
 **mPDialog**.show();  
  
 OkGo.<String>*post*(Constants.***LOGIN***)  
 .tag(**this**)  
 .params(**"username"**, account)  
 .params(**"password"**, pwd)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String result = response.body().toString();  
 Log.*d*(***TAG1***, result);  
 **if** (result.contains(**"error"**)) {  
 Toast.*makeText*(getApplicationContext(),  
 **"账号或者密码错误"**,  
 Toast.***LENGTH\_SHORT***).show();  
 **mPDialog**.cancel();  
 } **else if** (result.contains(**"token"**)) {  
  
  
 *//登陆成功* PrefUtils.*setString*(getApplicationContext(), **"login\_success"**, result);  
 PrefUtils.*setString*(getApplicationContext(), **"username"**, account);  
 **mPDialog**.cancel();  
 SpUtil.*saveLoginState*(**true**);  
 Toast.*makeText*(getApplicationContext(), **"登录成功"**,  
 Toast.***LENGTH\_SHORT***).show();  
 *//恢复极光服务* JPushInterface.*resumePush*(getApplicationContext());  
 startActivity(**new** Intent(getApplicationContext(), HomeActivity.**class**));  
 finish();  
  
 } **else** {  
 Toast.*makeText*(getApplicationContext(),  
 **"登录异常"**,  
 Toast.***LENGTH\_SHORT***).show();  
 **mPDialog**.cancel();  
  
 }  
  
  
 }  
  
 @Override  
 **public void** onError(Response<String> response) {  
 **super**.onError(response);  
  
 ToastUtils.*showShort*(**"没有网络，请检查网络"**);  
  
 }  
 });  
  
  
 } **else** {  
  
 *//非法电话号码* ToastUtils.*showShort*(**"请输入正确的电话号码"**);  
  
 }  
  
 }  
  
  
 **break**;  
 } **else** {  
  
  
 ToastUtils.*showShort*(**"账号和密码不能为空"**);  
  
 }  
  
  
 }  
 }  
  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.annotation.NonNull;  
**import** android.support.annotation.Nullable;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.LinearLayoutManager;  
**import** android.support.v7.widget.RecyclerView;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.facebook.drawee.view.SimpleDraweeView;  
**import** com.google.gson.Gson;  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.bean.LoginSuccess;  
**import** com.jinkun\_innovation.pastureland.bean.QueryByYang;  
**import** com.jinkun\_innovation.pastureland.common.Constants;  
**import** com.jinkun\_innovation.pastureland.ui.YangDetailActivity;  
**import** com.jinkun\_innovation.pastureland.ui.locui.CamelLocActivity;  
**import** com.jinkun\_innovation.pastureland.utilcode.util.ToastUtils;  
**import** com.jinkun\_innovation.pastureland.utils.PrefUtils;  
**import** com.lzy.okgo.OkGo;  
**import** com.lzy.okgo.callback.StringCallback;  
**import** com.lzy.okgo.model.Response;  
**import** com.scwang.smartrefresh.layout.api.RefreshLayout;  
**import** com.scwang.smartrefresh.layout.listener.OnLoadMoreListener;  
**import** com.scwang.smartrefresh.layout.listener.OnRefreshListener;  
  
**import** java.util.List;  
  
**import** cn.pedant.SweetAlert.SweetAlertDialog;  
  
*/\*\*  
 \* Created by Guan on 2018/3/16.  
 \*/***public class** CamelListActivity **extends** AppCompatActivity {  
  
 **private static final** String ***TAG1*** = CamelListActivity.**class**.getSimpleName();  
  
 **private** List<QueryByYang.LivestockVarietyListBean> **mLivestockVarietyList**;  
  
 **int index** = 2;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_camel\_list***);  
  
 ImageView ivMuqunLoc = findViewById(R.id.***ivMuqunLoc***);  
 ivMuqunLoc.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 Intent intent = **new** Intent(getApplicationContext(), CamelLocActivity.**class**);  
 startActivity(intent);  
  
 }  
 });  
  
  
 ImageView ivBack = (ImageView) findViewById(R.id.***ivBack***);  
 ivBack.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 finish();  
  
 }  
 });  
  
  
 RefreshLayout refreshLayout = findViewById(R.id.***refreshLayout***);  
 refreshLayout.setOnRefreshListener(**new** OnRefreshListener() {  
 @Override  
 **public void** onRefresh(RefreshLayout refreshlayout) {  
  
  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.QUERYLIVESTOCKVARIETYLIST)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 7)  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 **mLivestockVarietyList** = queryByYang.getLivestockVarietyList();  
 String deviceNo = **mLivestockVarietyList**.get(0).getDeviceNo();  
 Log.*d*(***TAG1***, deviceNo);  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
 } **else** {  
  
  
 }  
  
  
 }  
 });  
  
  
 refreshlayout.finishRefresh(2000);*//传入false表示刷新失败* }  
 });  
  
 refreshLayout.setOnLoadMoreListener(**new** OnLoadMoreListener() {  
 @Override  
 **public void** onLoadMore(@NonNull RefreshLayout refreshLayout) {  
  
  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.QUERYLIVESTOCKVARIETYLIST)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 7)  
 .params(**"current"**, **index**)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 **index**++;  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 List<QueryByYang.LivestockVarietyListBean> mylist =  
 queryByYang.getLivestockVarietyList();  
  
  
 **if** (mylist.size() == 0) {  
  
 ToastUtils.*showShort*(**"没有更多数据了"**);  
  
 } **else** {  
 **for** (**int** i = 0; i < mylist.size(); i++) {  
 **mLivestockVarietyList**.add(mylist.get(i));  
 }  
 *MoveToPosition*(**mLayoutManager**, 10 \* (**index** - 1));  
  
  
 }  
  
  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"没有更多数据了"**);  
  
  
 }  
  
  
 }  
 });  
  
 refreshLayout.finishLoadMore();*//传入false表示加载失败* }  
 });  
  
  
 **mRecyclerView** = findViewById(R.id.***my\_recycler\_view***);  
*//创建默认的线性LayoutManager* **mLayoutManager** = **new** LinearLayoutManager(**this**);  
 **mRecyclerView**.setLayoutManager(**mLayoutManager**);  
*//如果可以确定每个item的高度是固定的，设置这个选项可以提高性能* **mRecyclerView**.setHasFixedSize(**true**);  
  
 initData();  
  
  
 }  
  
 String **mLogin\_success**;  
 LoginSuccess **mLoginSuccess**;  
 String **mUsername**;  
  
 */\*\*  
 \* RecyclerView 移动到当前位置，  
 \*  
 \** ***@param manager*** *设置RecyclerView对应的manager  
 \** ***@param n*** *要跳转的位置  
 \*/* **public static void** MoveToPosition(LinearLayoutManager manager, **int** n) {  
 manager.scrollToPositionWithOffset(n, 0);  
 manager.setStackFromEnd(**true**);  
 }  
  
  
 **private void** initData() {  
  
 **mLogin\_success** = PrefUtils.*getString*(**this**, **"login\_success"**, **null**);  
 **final** Gson gson = **new** Gson();  
 **mLoginSuccess** = gson.fromJson(**mLogin\_success**, LoginSuccess.**class**);  
 **mUsername** = PrefUtils.*getString*(**this**, **"username"**, **null**);  
  
  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.QUERYLIVESTOCKVARIETYLIST)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 7)  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 **mLivestockVarietyList** = queryByYang.getLivestockVarietyList();  
 String deviceNo = **mLivestockVarietyList**.get(0).getDeviceNo();  
 Log.*d*(***TAG1***, deviceNo);  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
 } **else** {  
  
  
 }  
  
  
 }  
 });  
  
  
 }  
  
 **private** String[] getDummyDatas() {  
  
 String[] arr = {**"北京"**, **"上海"**, **"广州"**, **"深圳"**};  
  
 **return** arr;  
  
 }  
  
 **private** RecyclerView **mRecyclerView**;  
 **private** LinearLayoutManager **mLayoutManager**;  
 **private** MyAdapter **mAdapter**;  
  
 **public static interface** OnRecyclerViewItemClickListener {  
 **void** onItemClick(View view, **int** data);  
  
  
 }  
  
 **private** OnRecyclerViewItemClickListener **mOnItemClickListener** = **null**;  
  
 **public void** setOnItemClickListener(OnRecyclerViewItemClickListener listener) {  
 **this**.**mOnItemClickListener** = listener;  
  
  
 }  
  
  
 **public class** MyAdapter **extends** RecyclerView.Adapter<MyAdapter.ViewHolder> **implements** View.OnClickListener {  
  
 **private** OnRecyclerViewItemClickListener **mOnItemClickListener** = **null**;  
  
  
 **public void** setOnItemClickListener(OnRecyclerViewItemClickListener listener) {  
 **this**.**mOnItemClickListener** = listener;  
  
 }  
  
 **public** List<QueryByYang.LivestockVarietyListBean> **datas** = **null**;  
  
 **public** MyAdapter(List<QueryByYang.LivestockVarietyListBean> datas) {  
  
 **this**.**datas** = datas;  
 }  
  
 *//创建新View，被LayoutManager所调用* @Override  
 **public** ViewHolder onCreateViewHolder(ViewGroup viewGroup, **int** viewType) {  
 View view = LayoutInflater.*from*(  
 viewGroup.getContext()).inflate(R.layout.***item\_yang\_list***,  
 viewGroup, **false**);  
  
 ViewHolder vh = **new** ViewHolder(view);  
  
 *//将创建的View注册点击事件* view.setOnClickListener(**this**);  
  
 **return** vh;  
 }  
  
  
 *//将数据与界面进行绑定的操作* @Override  
 **public void** onBindViewHolder(ViewHolder viewHolder,**final int** position) {  
  
*// viewHolder.mTextView.setText(datas[position]);  
 //将数据保存在itemView的Tag中，以便点击时进行获取* viewHolder.**itemView**.setTag(position);  
  
 viewHolder.**itemView**.setOnLongClickListener(**new** View.OnLongClickListener() {  
 @Override  
 **public boolean** onLongClick(View view) {  
  
 *//删除第 position 条目* **new** SweetAlertDialog(CamelListActivity.**this**, SweetAlertDialog.***WARNING\_TYPE***)  
 .setTitleText(**"删除?"**)  
 .setContentText(**"删除此条目"**)  
 .setConfirmText(**"确定"**)  
 .setConfirmClickListener(**new** SweetAlertDialog.OnSweetClickListener() {  
 @Override  
 **public void** onClick(SweetAlertDialog sDialog) {  
  
 sDialog.cancel();  
 OkGo.<String>*get*(Constants.***delLivestock***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"deviceNo"**, **datas**.get(position).getDeviceNo())  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String result = response.body().toString();  
 **if** (result.contains(**"删除成功"**)) {  
  
 ToastUtils.*showShort*(**"删除成功"**);  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.QUERYLIVESTOCKVARIETYLIST)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 7)  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 **mLivestockVarietyList** = queryByYang.getLivestockVarietyList();  
 String deviceNo = **mLivestockVarietyList**.get(0).getDeviceNo();  
 Log.*d*(***TAG1***, deviceNo);  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
  
  
 } **else** {  
  
  
 }  
  
  
 }  
 });  
  
  
 } **else if** (result.contains(**"删除失败"**)) {  
  
  
 ToastUtils.*showShort*(**"删除失败"**);  
  
  
 } **else if** (result.contains(**"已被认领不可删除"**)) {  
  
 ToastUtils.*showShort*(**"已被认领不可删除"**);  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"删除异常"**);  
  
 }  
  
 }  
 });  
  
  
 }  
 })  
 .setCancelText(**"取消"**)  
 .setCancelClickListener(**new** SweetAlertDialog.OnSweetClickListener() {  
 @Override  
 **public void** onClick(SweetAlertDialog sDialog) {  
 sDialog.cancel();  
  
  
 }  
 })  
 .show();  
  
 **return false**;  
  
 }  
  
  
 });  
  
  
 String imgUrl = **datas**.get(position).getImgUrl();  
 imgUrl = Constants.***BASE\_URL*** + imgUrl;  
 Uri uri = Uri.*parse*(imgUrl);  
 viewHolder.**dvYang**.setImageURI(uri);  
  
 String variety = **datas**.get(position).getVariety();  
  
 Log.*d*(***TAG1***, **"variety = "** + variety);  
 **if** (variety.equals(**"701"**)) {  
 viewHolder.**tvYangName**.setText(**"品种：骆驼"**);  
 } **else** {  
 viewHolder.**tvYangName**.setText(**"品种：其他"**);  
 }  
  
 viewHolder.**tvDeviceNo**.setText(**"设备号："** + **datas**.get(position).getDeviceNo());  
 viewHolder.**tvPublishTime**.setText(**"发布时间："** + **datas**.get(position).getUpdateTime());  
  
 viewHolder.**tvLocation**.setText(**"牧场："** + **datas**.get(position).**name**);  
  
  
 }  
  
 @Override  
 **public void** onClick(View v) {  
 **if** (**mOnItemClickListener** != **null**) {  
 *//注意这里使用getTag方法获取数据* **mOnItemClickListener**.onItemClick(v, (**int**) v.getTag());  
  
 }  
 }  
  
 *//获取数据的数量* @Override  
 **public int** getItemCount() {  
 **return datas**.size();  
 }  
  
  
 *//自定义的ViewHolder，持有每个Item的的所有界面元素* **public class** ViewHolder **extends** RecyclerView.ViewHolder {  
 **public** TextView **mTextView**;  
 **public** SimpleDraweeView **dvYang**;  
 **public** TextView **tvYangName**, **tvDeviceNo**, **tvPublishTime**, **tvLocation**;  
  
  
 **public** ViewHolder(View view) {  
 **super**(view);  
*// mTextView = view.findViewById(R.id.tvClaim);* **dvYang** = view.findViewById(R.id.***dvYang***);  
 **tvYangName** = view.findViewById(R.id.***tvYangName***);  
 **tvDeviceNo** = view.findViewById(R.id.***tvDeviceNo***);  
 **tvPublishTime** = view.findViewById(R.id.***tvPublishTime***);  
 **tvLocation** = view.findViewById(R.id.***tvLocation***);  
  
  
 }  
  
 }  
  
 }  
  
  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.support.annotation.Nullable;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.LinearLayoutManager;  
**import** android.support.v7.widget.RecyclerView;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.google.gson.Gson;  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.bean.DeviceMsg;  
**import** com.jinkun\_innovation.pastureland.bean.LoginSuccess;  
**import** com.jinkun\_innovation.pastureland.common.Constants;  
**import** com.jinkun\_innovation.pastureland.utilcode.AppManager;  
**import** com.jinkun\_innovation.pastureland.utils.PrefUtils;  
**import** com.lzy.okgo.OkGo;  
**import** com.lzy.okgo.callback.StringCallback;  
**import** com.lzy.okgo.model.Response;  
  
**import** java.util.List;  
  
  
*/\*\*  
 \* Created by Guan on 2018/4/21.  
 \*/***public class** ClaimMsgActivity **extends** AppCompatActivity {  
  
  
 **private static final** String ***TAG1*** = ClaimMsgActivity.**class**.getSimpleName();  
  
 RecyclerView **mRecyclerView**;  
 LinearLayoutManager **mLayoutManager**;  
 MyAdapter **mAdapter**;  
  
 **public class** MyAdapter **extends** RecyclerView.Adapter<MyAdapter.ViewHolder> {  
 **public** List<DeviceMsg.BatteryListBean> **datas** = **null**;  
  
 **public** MyAdapter(List<DeviceMsg.BatteryListBean> datas) {  
 **this**.**datas** = datas;  
 }  
  
 *//创建新View，被LayoutManager所调用* @Override  
 **public** ViewHolder onCreateViewHolder(ViewGroup viewGroup, **int** viewType) {  
  
 View view = LayoutInflater.*from*(viewGroup.getContext()).  
 inflate(R.layout.***item\_device\_msg***, viewGroup, **false**);  
 ViewHolder vh = **new** ViewHolder(view);  
 **return** vh;  
  
 }  
  
 *//将数据与界面进行绑定的操作* @Override  
 **public void** onBindViewHolder(ViewHolder viewHolder, **int** position) {  
  
*// viewHolder.mTextView.setText(datas[position]);* viewHolder.**tvTime1**.setText(**datas**.get(position).getCreateTime());  
 viewHolder.**tvLowPower**.setText(**"设备"** + **datas**.get(position).getDeviceNo() + **"于"** +  
 **datas**.get(position).getCreateTime() + **"发出了低电报警，请尽快去确认，及时充电"**);  
  
  
 }  
  
 *//获取数据的数量* @Override  
 **public int** getItemCount() {  
 **return datas**.size();  
 }  
  
 *//自定义的ViewHolder，持有每个Item的的所有界面元素* **public class** ViewHolder **extends** RecyclerView.ViewHolder {  
  
 *// public TextView mTextView;* TextView **tvTime1**;  
 TextView **tvLowPower**;  
  
  
 **public** ViewHolder(View view) {  
 **super**(view);  
  
*// mTextView = (TextView) view.findViewById(R.id.text);* **tvTime1** = view.findViewById(R.id.***tvTime1***);  
 **tvLowPower** = view.findViewById(R.id.***tvLowPower***);  
  
  
 }  
  
  
 }  
 }  
  
 String **mLogin\_success**;  
 LoginSuccess **mLoginSuccess**;  
 String **mUsername**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_claim\_msg***);  
  
 AppManager.*getAppManager*().addActivity(**this**);  
  
 ImageView ivBack = (ImageView) findViewById(R.id.***ivBack***);  
 ivBack.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 finish();  
 }  
 });  
  
 **mRecyclerView** = (RecyclerView) findViewById(R.id.***my\_recycler\_view***);  
 *//创建默认的线性LayoutManager* **mLayoutManager** = **new** LinearLayoutManager(**this**);  
 **mRecyclerView**.setLayoutManager(**mLayoutManager**);  
 *//如果可以确定每个item的高度是固定的，设置这个选项可以提高性能* **mRecyclerView**.setHasFixedSize(**true**);  
  
 **mLogin\_success** = PrefUtils.*getString*(**this**, **"login\_success"**, **null**);  
 Gson gson = **new** Gson();  
 **mLoginSuccess** = gson.fromJson(**mLogin\_success**, LoginSuccess.**class**);  
 **mUsername** = PrefUtils.*getString*(**this**, **"username"**, **null**);  
  
 OkGo.<String>*post*(Constants.***DEVICEMSG***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 1000)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
  
  
 Gson gson1 = **new** Gson();  
 DeviceMsg deviceMsg = gson1.fromJson(s, DeviceMsg.**class**);  
 List<DeviceMsg.BatteryListBean> batteryList = deviceMsg.getBatteryList();  
 List<DeviceMsg.LivestockClaimListBean> livestockClaimList  
 = deviceMsg.getLivestockClaimList();  
  
 **if** (batteryList.size() != 0) {  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(batteryList);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
  
 }  
  
  
 }  
 });  
  
  
 *//轮询  
// apHandler.postDelayed(apRunnable, 1000);* }  
  
 **private** Handler **apHandler** = **new** Handler();  
 **int count** = 0;  
 **private** Runnable **apRunnable** = **new** Runnable() {  
  
 @Override  
 **public void** run() {  
 *// handler自带方法实现定时器* **apHandler**.postDelayed(**this**, 1000\*60);  
  
  
 Log.*d*(***TAG1***, **"每隔一秒请求服务器，第"** + **count**++ + **"次了"**);  
  
 OkGo.<String>*post*(Constants.***DEVICEMSG***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 1000)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
  
  
 Gson gson1 = **new** Gson();  
 DeviceMsg deviceMsg = gson1.fromJson(s, DeviceMsg.**class**);  
 List<DeviceMsg.BatteryListBean> batteryList = deviceMsg.getBatteryList();  
 List<DeviceMsg.LivestockClaimListBean> livestockClaimList  
 = deviceMsg.getLivestockClaimList();  
  
 **if** (batteryList.size() != 0) {  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(batteryList);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
 }  
  
  
 }  
 });  
  
  
 }  
  
 };  
  
  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.app.Activity;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.view.View;  
**import** android.widget.EditText;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.facebook.drawee.view.SimpleDraweeView;  
**import** com.google.gson.Gson;  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.bean.LoginSuccess;  
**import** com.jinkun\_innovation.pastureland.bean.MuqunDetail;  
**import** com.jinkun\_innovation.pastureland.common.Constants;  
**import** com.jinkun\_innovation.pastureland.utilcode.util.ToastUtils;  
**import** com.jinkun\_innovation.pastureland.utils.PrefUtils;  
**import** com.lzy.okgo.OkGo;  
**import** com.lzy.okgo.callback.StringCallback;  
**import** com.lzy.okgo.model.Response;  
  
**import** butterknife.BindView;  
**import** butterknife.ButterKnife;  
**import** butterknife.OnClick;  
  
*/\*\*  
 \* Created by Guan on 2018/4/4.  
 \*/***public class** EditMuchangActivity **extends** Activity {  
  
 @BindView(R.id.***ivBack***)  
 ImageView **mIvBack**;  
 @BindView(R.id.***tvSave***)  
 TextView **mTvSave**;  
 @BindView(R.id.***sdvYang***)  
 SimpleDraweeView **mSdvYang**;  
  
 @BindView(R.id.***etMuChangName***)  
 EditText **etMuChangName**;  
 @BindView(R.id.***tvMianji***)  
 EditText **mTvMianji**;  
 @BindView(R.id.***tvRiverNearby***)  
 EditText **mTvRiverNearby**;  
  
  
  
 @BindView(R.id.***tvDetail***)  
 EditText **mTvDetail**;  
 **private** LoginSuccess **mLoginSuccess**;  
 **private** String **mLogin\_success**;  
 **private** String **mUsername**;  
 **private** String **mRanchImgUrl**;  
 **private** String **mImgUrl**;  
 **private** String **mRanchurl**;  
 **private** String **mName**;  
 **private** String **mIntroduceAnimalCount**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_muchang\_edit***);  
 ButterKnife.*bind*(**this**);  
  
 **mLogin\_success** = PrefUtils.*getString*(**this**, **"login\_success"**, **null**);  
 Gson gson = **new** Gson();  
 **mLoginSuccess** = gson.fromJson(**mLogin\_success**, LoginSuccess.**class**);  
 **mUsername** = PrefUtils.*getString*(**this**, **"username"**, **null**);  
  
 OkGo.<String>*get*(Constants.***RANCH***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String result = response.body().toString();  
 Gson gson1 = **new** Gson();  
 MuqunDetail muqunDetail = gson1.fromJson(result, MuqunDetail.**class**);  
 String msg = muqunDetail.getMsg();  
 **if** (msg.equals(**"获取牧场详情成功"**)) {  
  
 String imgUrl = muqunDetail.getRanch().getImgUrl();  
 imgUrl = Constants.***BASE\_URL*** + imgUrl;  
 **mSdvYang**.setImageURI(Uri.*parse*(imgUrl));  
  
 **mSdvYang**.setImageURI(Uri.*parse*(imgUrl));  
  
 **mImgUrl** = muqunDetail.getRanch().getImgUrl();  
  
 **mName** = muqunDetail.getRanch().getName();  
  
 **etMuChangName**.setText(muqunDetail.getRanch().getName());  
 **mTvMianji**.setText(muqunDetail.getRanch().getAcreage());  
 **mTvRiverNearby**.setText(muqunDetail.getRanch().getIntroduceRiver());  
  
 **mTvDetail**.setText(muqunDetail.getRanch().getIntroduce());  
  
 **mIntroduceAnimalCount** = muqunDetail.getRanch().getIntroduceAnimalCount();  
  
  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"获取牧场详情失败"**);  
  
 }  
 }  
 });  
  
  
 }  
  
  
 @OnClick({R.id.***ivBack***, R.id.***tvSave***})  
 **public void** onViewClicked(View view) {  
 **switch** (view.getId()) {  
 **case** R.id.***ivBack***:  
  
 finish();  
  
 **break**;  
 **case** R.id.***tvSave***:  
  
 *//更新牧场* OkGo.<String>*post*(Constants.***UPDRANCH***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"imgUrl"**, **mImgUrl**)  
 .params(**"name"**, **etMuChangName**.getText().toString()) *//牧场名称* .params(**"acreage"**, **mTvMianji**.getText().toString()) *//面积* .params(**"introduceAnimalCount"**, **mIntroduceAnimalCount**) *//数量* .params(**"introduceRiver"**, **mTvRiverNearby**.getText().toString()) *// 河流 0：无，1有* .params(**"isFenceClose"**, **""**) *// 围栏封闭 0：封闭，1开放* .params(**"introduce"**, **mTvDetail**.getText().toString()) *//介绍  
// .params("longtitudeBaidu", "")  
// .params("lantitudeBaidu", "")* .params(**"ranchImgUrl"**, **mRanchurl**)  
 .params(**"videoUrl"**, **""**)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String result = response.body().toString();  
 **if** (result.contains(**"success"**)) {  
  
 ToastUtils.*showShort*(**"牧场信息更新成功"**);  
 setResult(***RESULT\_OK***);  
 finish();  
  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"牧场信息更新失败"**);  
 }  
  
 }  
 });  
  
  
 **break**;  
 }  
 }  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.annotation.NonNull;  
**import** android.support.annotation.Nullable;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.LinearLayoutManager;  
**import** android.support.v7.widget.RecyclerView;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.facebook.drawee.view.SimpleDraweeView;  
**import** com.google.gson.Gson;  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.bean.LoginSuccess;  
**import** com.jinkun\_innovation.pastureland.bean.QueryByYang;  
**import** com.jinkun\_innovation.pastureland.common.Constants;  
**import** com.jinkun\_innovation.pastureland.ui.YangDetailActivity;  
**import** com.jinkun\_innovation.pastureland.ui.locui.MaLocActivity;  
**import** com.jinkun\_innovation.pastureland.utilcode.util.ToastUtils;  
**import** com.jinkun\_innovation.pastureland.utils.PrefUtils;  
**import** com.lzy.okgo.OkGo;  
**import** com.lzy.okgo.callback.StringCallback;  
**import** com.lzy.okgo.model.Response;  
**import** com.scwang.smartrefresh.layout.api.RefreshLayout;  
**import** com.scwang.smartrefresh.layout.listener.OnLoadMoreListener;  
**import** com.scwang.smartrefresh.layout.listener.OnRefreshListener;  
  
**import** java.util.List;  
  
**import** cn.pedant.SweetAlert.SweetAlertDialog;  
  
*/\*\*  
 \* Created by Guan on 2018/3/16.  
 \*/***public class** MaListActivity **extends** AppCompatActivity {  
  
 **private static final** String ***TAG1*** = MaListActivity.**class**.getSimpleName();  
  
 **private** List<QueryByYang.LivestockVarietyListBean> **mLivestockVarietyList**;  
  
 **int index** = 2;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_ma\_list***);  
  
 ImageView ivBack = (ImageView) findViewById(R.id.***ivBack***);  
 ivBack.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 finish();  
  
 }  
 });  
  
  
 RefreshLayout refreshLayout = findViewById(R.id.***refreshLayout***);  
 refreshLayout.setOnRefreshListener(**new** OnRefreshListener() {  
 @Override  
 **public void** onRefresh(RefreshLayout refreshlayout) {  
  
  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.***QUERYLIVESTOCKVARIETYLIST***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 3)  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 **mLivestockVarietyList** = queryByYang.getLivestockVarietyList();  
 String deviceNo = **mLivestockVarietyList**.get(0).getDeviceNo();  
 Log.*d*(***TAG1***, deviceNo);  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
 } **else** {  
  
  
 }  
  
  
 }  
 });  
  
  
 refreshlayout.finishRefresh(2000);*//传入false表示刷新失败* }  
 });  
  
 refreshLayout.setOnLoadMoreListener(**new** OnLoadMoreListener() {  
 @Override  
 **public void** onLoadMore(@NonNull RefreshLayout refreshLayout) {  
  
  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.***QUERYLIVESTOCKVARIETYLIST***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 3)  
 .params(**"current"**, **index**)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 **index**++;  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 List<QueryByYang.LivestockVarietyListBean> mylist =  
 queryByYang.getLivestockVarietyList();  
  
  
 **if** (mylist.size() == 0) {  
  
 ToastUtils.*showShort*(**"没有更多数据了"**);  
  
 } **else** {  
 **for** (**int** i = 0; i < mylist.size(); i++) {  
 **mLivestockVarietyList**.add(mylist.get(i));  
 }  
 *MoveToPosition*(**mLayoutManager**, 10 \* (**index** - 1));  
  
  
 }  
  
  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"没有更多数据了"**);  
  
  
 }  
  
  
 }  
 });  
  
 refreshLayout.finishLoadMore();*//传入false表示加载失败* }  
 });  
  
  
 **mRecyclerView** = findViewById(R.id.***my\_recycler\_view***);  
*//创建默认的线性LayoutManager* **mLayoutManager** = **new** LinearLayoutManager(**this**);  
 **mRecyclerView**.setLayoutManager(**mLayoutManager**);  
*//如果可以确定每个item的高度是固定的，设置这个选项可以提高性能* **mRecyclerView**.setHasFixedSize(**true**);  
  
 initData();  
  
 ImageView ivMuqunLoc = findViewById(R.id.***ivMuqunLoc***);  
 ivMuqunLoc.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 Intent intent = **new** Intent(getApplicationContext(), MaLocActivity.**class**);  
 startActivity(intent);  
  
 }  
 });  
  
  
 }  
  
 String **mLogin\_success**;  
 LoginSuccess **mLoginSuccess**;  
 String **mUsername**;  
  
 */\*\*  
 \* RecyclerView 移动到当前位置，  
 \*  
 \** ***@param manager*** *设置RecyclerView对应的manager  
 \** ***@param n*** *要跳转的位置  
 \*/* **public static void** MoveToPosition(LinearLayoutManager manager, **int** n) {  
 manager.scrollToPositionWithOffset(n, 0);  
 manager.setStackFromEnd(**true**);  
 }  
  
  
 **private void** initData() {  
  
 **mLogin\_success** = PrefUtils.*getString*(**this**, **"login\_success"**, **null**);  
 **final** Gson gson = **new** Gson();  
 **mLoginSuccess** = gson.fromJson(**mLogin\_success**, LoginSuccess.**class**);  
 **mUsername** = PrefUtils.*getString*(**this**, **"username"**, **null**);  
  
  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.***QUERYLIVESTOCKVARIETYLIST***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 3)  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 **mLivestockVarietyList** = queryByYang.getLivestockVarietyList();  
 String deviceNo = **mLivestockVarietyList**.get(0).getDeviceNo();  
 Log.*d*(***TAG1***, deviceNo);  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
 } **else** {  
  
  
 }  
  
  
 }  
 });  
  
  
 }  
  
 **private** String[] getDummyDatas() {  
  
 String[] arr = {**"北京"**, **"上海"**, **"广州"**, **"深圳"**};  
  
 **return** arr;  
  
 }  
  
 **private** RecyclerView **mRecyclerView**;  
 **private** LinearLayoutManager **mLayoutManager**;  
 **private** MyAdapter **mAdapter**;  
  
 **public static interface** OnRecyclerViewItemClickListener {  
 **void** onItemClick(View view, **int** data);  
  
  
 }  
  
 **private** OnRecyclerViewItemClickListener **mOnItemClickListener** = **null**;  
  
 **public void** setOnItemClickListener(OnRecyclerViewItemClickListener listener) {  
 **this**.**mOnItemClickListener** = listener;  
  
  
 }  
  
  
 **public class** MyAdapter **extends** RecyclerView.Adapter<MyAdapter.ViewHolder> **implements** View.OnClickListener {  
  
 **private** OnRecyclerViewItemClickListener **mOnItemClickListener** = **null**;  
  
  
 **public void** setOnItemClickListener(OnRecyclerViewItemClickListener listener) {  
 **this**.**mOnItemClickListener** = listener;  
  
 }  
  
 **public** List<QueryByYang.LivestockVarietyListBean> **datas** = **null**;  
  
 **public** MyAdapter(List<QueryByYang.LivestockVarietyListBean> datas) {  
  
 **this**.**datas** = datas;  
 }  
  
 *//创建新View，被LayoutManager所调用* @Override  
 **public** ViewHolder onCreateViewHolder(ViewGroup viewGroup, **int** viewType) {  
 View view = LayoutInflater.*from*(  
 viewGroup.getContext()).inflate(R.layout.***item\_yang\_list***,  
 viewGroup, **false**);  
  
 ViewHolder vh = **new** ViewHolder(view);  
  
 *//将创建的View注册点击事件* view.setOnClickListener(**this**);  
  
 **return** vh;  
 }  
  
  
 *//将数据与界面进行绑定的操作* @Override  
 **public void** onBindViewHolder(ViewHolder viewHolder,**final int** position) {  
  
*// viewHolder.mTextView.setText(datas[position]);  
 //将数据保存在itemView的Tag中，以便点击时进行获取* viewHolder.**itemView**.setTag(position);  
  
  
 viewHolder.**itemView**.setOnLongClickListener(**new** View.OnLongClickListener() {  
 @Override  
 **public boolean** onLongClick(View view) {  
  
 *//删除第 position 条目* **new** SweetAlertDialog(MaListActivity.**this**, SweetAlertDialog.***WARNING\_TYPE***)  
 .setTitleText(**"删除?"**)  
 .setContentText(**"删除此条目"**)  
 .setConfirmText(**"确定"**)  
 .setConfirmClickListener(**new** SweetAlertDialog.OnSweetClickListener() {  
 @Override  
 **public void** onClick(SweetAlertDialog sDialog) {  
  
 sDialog.cancel();  
 OkGo.<String>*get*(Constants.***delLivestock***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"deviceNo"**, **datas**.get(position).getDeviceNo())  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String result = response.body().toString();  
 **if** (result.contains(**"删除成功"**)) {  
  
 ToastUtils.*showShort*(**"删除成功"**);  
 *//通过牲畜类型查询所有牲畜* OkGo.<String>*get*(Constants.***QUERYLIVESTOCKVARIETYLIST***)  
 .tag(**this**)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"livestockType"**, 3)  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 10)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String s = response.body().toString();  
 Log.*d*(***TAG1***, s);  
  
 **if** (s.contains(**"imgUrl"**)) {  
 *//有数据* Gson gson1 = **new** Gson();  
 QueryByYang queryByYang = gson1.fromJson(s, QueryByYang.**class**);  
 **mLivestockVarietyList** = queryByYang.getLivestockVarietyList();  
 String deviceNo = **mLivestockVarietyList**.get(0).getDeviceNo();  
 Log.*d*(***TAG1***, deviceNo);  
 *//创建并设置Adapter* **mAdapter** = **new** MyAdapter(**mLivestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
 **mAdapter**.setOnItemClickListener(**new** OnRecyclerViewItemClickListener() {  
 @Override  
 **public void** onItemClick(View view, **int** position) {  
  
 Intent intent = **new** Intent(getApplicationContext(), YangDetailActivity.**class**);  
 intent.putExtra(**"getVariety"**, **mLivestockVarietyList**.get(position).getVariety());  
 intent.putExtra(**"getImgUrl"**, **mLivestockVarietyList**.get(position).getImgUrl());  
 intent.putExtra(**"getDeviceNo"**, **mLivestockVarietyList**.get(position).getDeviceNo());  
 intent.putExtra(**"getWeight"**, **mLivestockVarietyList**.get(position).getWeight());  
 intent.putExtra(**"getBindStatus"**, **mLivestockVarietyList**.get(position).getBindStatus());  
 intent.putExtra(**"getIsClaimed"**, **mLivestockVarietyList**.get(position).getIsClaimed());  
 intent.putExtra(**"getUpdateTime"**, **mLivestockVarietyList**.get(position).getUpdateTime());  
 startActivity(intent);  
  
 }  
 });  
  
  
  
 } **else** {  
  
  
 }  
  
  
 }  
 });  
  
  
 } **else if** (result.contains(**"删除失败"**)) {  
  
  
 ToastUtils.*showShort*(**"删除失败"**);  
  
  
 } **else if** (result.contains(**"已被认领不可删除"**)) {  
  
 ToastUtils.*showShort*(**"已被认领不可删除"**);  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"删除异常"**);  
  
 }  
  
 }  
 });  
  
  
 }  
 })  
 .setCancelText(**"取消"**)  
 .setCancelClickListener(**new** SweetAlertDialog.OnSweetClickListener() {  
 @Override  
 **public void** onClick(SweetAlertDialog sDialog) {  
 sDialog.cancel();  
  
  
 }  
 })  
 .show();  
  
 **return false**;  
  
 }  
  
  
 });  
  
  
 String imgUrl = **datas**.get(position).getImgUrl();  
 imgUrl = Constants.***BASE\_URL*** + imgUrl;  
 Uri uri = Uri.*parse*(imgUrl);  
 viewHolder.**dvYang**.setImageURI(uri);  
  
 String variety = **datas**.get(position).getVariety();  
  
 Log.*d*(***TAG1***, **"variety = "** + variety);  
 **if** (variety.equals(**"301"**)) {  
 viewHolder.**tvYangName**.setText(**"品种：蒙古马"**);  
 } **else** {  
 viewHolder.**tvYangName**.setText(**"品种：其他"**);  
 }  
  
 viewHolder.**tvDeviceNo**.setText(**"设备号："** + **datas**.get(position).getDeviceNo());  
 viewHolder.**tvPublishTime**.setText(**"发布时间："** + **datas**.get(position).getUpdateTime());  
 viewHolder.**tvLocation**.setText(**"牧场："** + **datas**.get(position).**name**);  
  
  
 }  
  
 @Override  
 **public void** onClick(View v) {  
 **if** (**mOnItemClickListener** != **null**) {  
 *//注意这里使用getTag方法获取数据* **mOnItemClickListener**.onItemClick(v, (**int**) v.getTag());  
  
 }  
 }  
  
 *//获取数据的数量* @Override  
 **public int** getItemCount() {  
 **return datas**.size();  
 }  
  
  
 *//自定义的ViewHolder，持有每个Item的的所有界面元素* **public class** ViewHolder **extends** RecyclerView.ViewHolder {  
 **public** TextView **mTextView**;  
 **public** SimpleDraweeView **dvYang**;  
 **public** TextView **tvYangName**, **tvDeviceNo**, **tvPublishTime**, **tvLocation**;  
  
  
 **public** ViewHolder(View view) {  
 **super**(view);  
*// mTextView = view.findViewById(R.id.tvClaim);* **dvYang** = view.findViewById(R.id.***dvYang***);  
 **tvYangName** = view.findViewById(R.id.***tvYangName***);  
 **tvDeviceNo** = view.findViewById(R.id.***tvDeviceNo***);  
 **tvPublishTime** = view.findViewById(R.id.***tvPublishTime***);  
 **tvLocation** = view.findViewById(R.id.***tvLocation***);  
  
  
 }  
  
 }  
  
 }  
  
  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.app.Activity;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.view.View;  
**import** android.widget.EditText;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.jinkun\_innovation.pastureland.R;  
  
**import** butterknife.BindView;  
**import** butterknife.ButterKnife;  
**import** butterknife.OnClick;  
  
*/\*\*  
 \* Created by Guan on 2018/4/2.  
 \*/***public class** ModifyNameActivity **extends** Activity {  
  
 @BindView(R.id.***ivBack***)  
 ImageView **mIvBack**;  
 @BindView(R.id.***tvConfirm***)  
 TextView **mTvConfirm**;  
 @BindView(R.id.***etNewName***)  
 EditText **mEtNewName**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
  
 setContentView(R.layout.***activity\_modify\_name***);  
 ButterKnife.*bind*(**this**);  
  
  
 }  
  
  
 @OnClick({R.id.***ivBack***, R.id.***tvConfirm***, R.id.***etNewName***})  
 **public void** onViewClicked(View view) {  
 **switch** (view.getId()) {  
 **case** R.id.***ivBack***:  
  
 finish();  
  
 **break**;  
 **case** R.id.***tvConfirm***:  
  
 Intent data = **new** Intent();  
 data.putExtra(**"name"**, **mEtNewName**.getText().toString());  
 setResult(***RESULT\_OK***, data);  
 finish();  
  
 **break**;  
  
  
 **case** R.id.***etNewName***:  
  
  
 **break**;  
  
  
 }  
 }  
}