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

后30页

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.app.Activity;  
**import** android.content.Intent;  
**import** android.graphics.Color;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.text.Editable;  
**import** android.text.TextWatcher;  
**import** android.view.View;  
**import** android.widget.EditText;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.utilcode.util.ToastUtils;  
**import** com.jinkun\_innovation.pastureland.utils.PhoneFormatCheckUtils;  
  
**import** butterknife.BindView;  
**import** butterknife.ButterKnife;  
**import** butterknife.OnClick;  
  
*/\*\*  
 \* Created by Guan on 2018/4/3.  
 \*/***public class** ModifyPhoneActivity **extends** Activity {  
  
 @BindView(R.id.***ivBack***)  
 ImageView **mIvBack**;  
 @BindView(R.id.***tvConfirm***)  
 TextView **mTvConfirm**;  
 @BindView(R.id.***etNewName***)  
 EditText **mEtNewName**;  
 @BindView(R.id.***tvTips***)  
 TextView **tvTips**;  
  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_modify\_phone***);  
 ButterKnife.*bind*(**this**);  
  
 **mEtNewName**.addTextChangedListener(**new** TextWatcher() {  
 @Override  
 **public void** beforeTextChanged(CharSequence charSequence, **int** i, **int** i1, **int** i2) {  
  
 }  
  
 @Override  
 **public void** onTextChanged(CharSequence charSequence, **int** i, **int** i1, **int** i2) {  
  
 **boolean** mobile = PhoneFormatCheckUtils.*isMobile*(**mEtNewName**.getText().toString());  
  
 **if** (mobile) {  
 **tvTips**.setText(**"此号码是中国大陆号码"**);  
 **tvTips**.setTextColor(Color.***BLACK***);  
 } **else** {  
 **tvTips**.setText(**"请输入中国大陆号码"**);  
 **tvTips**.setTextColor(Color.***RED***);  
 }  
  
  
 }  
  
 @Override  
 **public void** afterTextChanged(Editable editable) {  
  
 }  
 });  
  
  
 }  
  
  
 @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***:  
  
 **boolean** mobile = PhoneFormatCheckUtils.*isMobile*(**mEtNewName**.getText().toString());  
 **if** (mobile) {  
  
 Intent data = **new** Intent();  
 data.putExtra(**"phone"**, **mEtNewName**.getText().toString());  
 setResult(***RESULT\_OK***, data);  
 finish();  
  
 } **else** {  
 ToastUtils.*showShort*(**"请输入中国大陆号码"**);  
 **tvTips**.setText(**"请输入中国大陆号码"**);  
 **tvTips**.setTextColor(Color.***RED***);  
  
 }  
  
  
 **break**;  
  
 **case** R.id.***etNewName***:  
  
  
 **break**;  
 }  
 }  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.app.Activity;  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.view.View;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.baidu.location.BDLocation;  
**import** com.baidu.mapapi.map.BaiduMap;  
**import** com.baidu.mapapi.map.BitmapDescriptor;  
**import** com.baidu.mapapi.map.BitmapDescriptorFactory;  
**import** com.baidu.mapapi.map.MapView;  
**import** com.baidu.mapapi.map.MyLocationConfiguration;  
**import** com.baidu.mapapi.map.MyLocationData;  
**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;  
  
**import static** com.jinkun\_innovation.pastureland.R.id.***tvDetail***;  
  
*/\*\*  
 \* Created by Guan on 2018/4/4.  
 \*/***public class** Muchang2Activity **extends** Activity {  
  
 @BindView(R.id.***ivBack***)  
 ImageView **mIvBack**;  
 @BindView(R.id.***ivEdit***)  
 ImageView **mIvEdit**;  
 @BindView(R.id.***sdvYang***)  
 SimpleDraweeView **mSdvYang**;  
 @BindView(R.id.***tvRancher***)  
 TextView **mTvRancher**;  
 @BindView(R.id.***tvPhone***)  
 TextView **mTvPhone**;  
 @BindView(R.id.***tvMianji***)  
 TextView **mTvMianji**;  
 @BindView(R.id.***tvRiverNearby***)  
 TextView **mTvRiverNearby**;  
 @BindView(R.id.***tvNum***)  
 TextView **mTvNum**;  
 @BindView(R.id.***tvVariety***)  
 TextView **mTvVariety**;  
 @BindView(R.id.***tvOwner***)  
 TextView **tvOwner**;  
  
 @BindView(***tvDetail***)  
 TextView **mTvDetail**;  
  
 **private** LoginSuccess **mLoginSuccess**;  
 **private** String **mLogin\_success**;  
 **private** String **mUsername**;  
  
  
 **private** MapView **mMapView** = **null**;  
 **private** BitmapDescriptor **mCurrentMarker**;  
 BaiduMap **map**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_muchang2***);  
 ButterKnife.*bind*(**this**);  
  
 *//获取地图控件引用* **mMapView** = (MapView) findViewById(R.id.***bmapView***);  
  
 **map** = **mMapView**.getMap();  
  
  
 **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));  
  
 **mTvRancher**.setText(**"牧场名："** + muqunDetail.getRanch().getName());  
 **mTvPhone**.setText(**"联系电话："** + muqunDetail.getRanch().getRancherAccount());  
  
 **mTvRiverNearby**.setText(**"附近河流："** + muqunDetail.getRanch().getIntroduceRiver());  
 **mTvMianji**.setText(**"面积："** + muqunDetail.getRanch().getAcreage() + **"亩"**);  
  
 **mTvNum**.setText(**"数量："** + muqunDetail.getRanch().getIntroduceAnimalCount() + **"头"**);  
 **mTvVariety**.setText(**"种类："** + muqunDetail.getRanch().getIntroduceAnimalType());  
  
 **tvOwner**.setText(**"牧场主："** + muqunDetail.getRanch().getRancherName());  
  
 **mTvDetail**.setText(muqunDetail.getRanch().getIntroduce());  
  
 String lantitudeBaidu = muqunDetail.getRanch().getLantitudeBaidu();  
 String longtitudeBaidu = muqunDetail.getRanch().getLongtitudeBaidu();  
  
 BDLocation bdLocation = **new** BDLocation();  
 bdLocation.setLongitude(Double.*parseDouble*(longtitudeBaidu));  
 bdLocation.setLatitude(Double.*parseDouble*(lantitudeBaidu));  
  
  
 *// 开启定位图层* **map**.setMyLocationEnabled(**true**);  
  
 *// 构造定位数据* MyLocationData locData = **new** MyLocationData.Builder()  
 .accuracy(bdLocation.getRadius())  
 *// 此处设置开发者获取到的方向信息，顺时针0-360  
 //.direction(100)* .latitude(bdLocation.getLatitude())  
 .longitude(bdLocation.getLongitude()).build();  
  
 *// 设置定位数据* **map**.setMyLocationData(locData);  
  
 *// 设置定位图层的配置（定位模式，是否允许方向信息，用户自定义定位图标）* **mCurrentMarker** = BitmapDescriptorFactory  
 .*fromResource*(R.mipmap.***icon\_location\_3***);  
 MyLocationConfiguration config = **new** MyLocationConfiguration(  
 MyLocationConfiguration.LocationMode.***FOLLOWING***,  
 **true**, **mCurrentMarker**);  
  
 **map**.setMyLocationConfiguration(config);  
  
 String address = PrefUtils.*getString*(getApplicationContext(), **"address"**, **null**);  
 TextView tvLoc = (TextView) findViewById(R.id.***tvLoc***);  
 tvLoc.setText(address);  
  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"获取牧场详情失败"**);  
  
 }  
  
  
 }  
 });  
  
  
 }  
  
 @Override  
 **protected void** onDestroy() {  
 **super**.onDestroy();  
 *//在activity执行onDestroy时执行mMapView.onDestroy()，实现地图生命周期管理* **mMapView**.onDestroy();  
 }  
  
 @Override  
 **protected void** onResume() {  
 **super**.onResume();  
 *//在activity执行onResume时执行mMapView. onResume ()，实现地图生命周期管理* **mMapView**.onResume();  
 }  
  
 @Override  
 **protected void** onPause() {  
 **super**.onPause();  
 *//在activity执行onPause时执行mMapView. onPause ()，实现地图生命周期管理* **mMapView**.onPause();  
 }  
  
  
 @OnClick({R.id.***ivBack***, R.id.***ivEdit***})  
 **public void** onViewClicked(View view) {  
 **switch** (view.getId()) {  
 **case** R.id.***ivBack***:  
  
 finish();  
  
 **break**;  
 **case** R.id.***ivEdit***:  
  
 Intent intent = **new** Intent(getApplicationContext(), EditMuchangActivity.**class**);  
 startActivityForResult(intent, 1001);  
  
 **break**;  
  
 }  
  
  
 }  
  
 @Override  
 **protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {  
 **super**.onActivityResult(requestCode, resultCode, data);  
  
 **if** (resultCode == ***RESULT\_OK***) {  
  
 **switch** (requestCode) {  
  
 **case** 1001:  
  
 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));  
  
 **mTvRancher**.setText(**"牧场名："** + muqunDetail.getRanch().getName());  
 **mTvPhone**.setText(**"联系电话："** + muqunDetail.getRanch().getRancherAccount());  
  
 **mTvRiverNearby**.setText(**"附近河流："** + muqunDetail.getRanch().getIntroduceRiver());  
 **mTvMianji**.setText(**"面积："** + muqunDetail.getRanch().getAcreage() + **"亩"**);  
  
 **mTvNum**.setText(**"数量："** + muqunDetail.getRanch().getIntroduceAnimalCount() + **"头"**);  
 **mTvVariety**.setText(**"种类："** + muqunDetail.getRanch().getIntroduceAnimalType());  
  
 **tvOwner**.setText(**"牧场主："** + muqunDetail.getRanch().getRancherName());  
  
 **mTvDetail**.setText(muqunDetail.getRanch().getIntroduce());  
  
  
 } **else** {  
  
 ToastUtils.*showShort*(**"获取牧场详情失败"**);  
  
 }  
  
  
 }  
 });  
  
  
 **break**;  
  
  
 }  
  
 }  
  
 }  
  
  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.view.View;  
**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.LoginSuccess;  
**import** com.jinkun\_innovation.pastureland.bean.MuqunDetail;  
**import** com.jinkun\_innovation.pastureland.common.Constants;  
**import** com.jinkun\_innovation.pastureland.utils.PrefUtils;  
**import** com.lzy.okgo.OkGo;  
**import** com.lzy.okgo.callback.StringCallback;  
**import** com.lzy.okgo.model.Response;  
  
*/\*\*  
 \* Created by Guan on 2018/4/1.  
 \*/***public class** MuchangDetailActivity **extends** Activity {  
  
 **private** LoginSuccess **mLoginSuccess**;  
 **private** String **mLogin\_success**;  
 **private** String **mUsername**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_muchang\_detail***);  
  
 ImageView ivBack = (ImageView) findViewById(R.id.***ivBack***);  
 ivBack.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 finish();  
  
 }  
 });  
  
 **final** TextView tvDetail = (TextView) findViewById(R.id.***tvDetail***);  
  
 **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) {  
  
 Gson gson1 = **new** Gson();  
 MuqunDetail muqunDetail = gson1.fromJson(response.body().toString(), MuqunDetail.**class**);  
  
 String address = muqunDetail.getRanch().getAddress();  
  
 tvDetail.setText(**"地址："**+address);  
  
  
 }  
 });  
  
  
 }  
  
  
}

**package** com.jinkun\_innovation.pastureland.ui.activity;  
  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.support.v7.widget.LinearLayoutManager;  
**import** android.support.v7.widget.RecyclerView;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
  
**import** com.baidu.location.BDLocation;  
**import** com.baidu.mapapi.map.BaiduMap;  
**import** com.baidu.mapapi.map.BitmapDescriptor;  
**import** com.baidu.mapapi.map.MapView;  
**import** com.google.gson.Gson;  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.bean.LoginSuccess;  
**import** com.jinkun\_innovation.pastureland.bean.MuqunLoc;  
**import** com.jinkun\_innovation.pastureland.common.Constants;  
**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 2018/5/4.  
 \*/***public class** MuqunLocActivity **extends** Activity {  
  
 **private static final** String ***TAG1*** = MuqunLocActivity.**class**.getSimpleName();  
 **private** MapView **mMapView** = **null**;  
 BaiduMap **mBaiduMap**;  
 **private** BitmapDescriptor **mCurrentMarker**;  
 **private** BDLocation **mLocation**;  
  
 String **mLogin\_success**;  
 LoginSuccess **mLoginSuccess**;  
 String **mUsername**;  
  
 List<MuqunLoc.LivestockVarietyListBean> **livestockVarietyList**;  
 RecyclerView **mRecyclerView**;  
 RecyclerView.LayoutManager **mLayoutManager**;  
 MyAdapter **mAdapter**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
  
 setContentView(R.layout.***activity\_muqun\_loc***);  
  
 ImageView ivBack = (ImageView) findViewById(R.id.***ivBack***);  
 ivBack.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 finish();  
 }  
 });  
  
 **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**);  
  
  
 **mRecyclerView** = (RecyclerView) findViewById(R.id.***rvList***);  
*//创建默认的线性LayoutManager* **mLayoutManager** = **new** LinearLayoutManager(getApplicationContext());  
 **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**);  
  
 OkGo.<String>*get*(Constants.***queryLivestockList***)  
 .tag(**this**)  
 .params(**"livestockType"**, 1)  
 .params(**"ranchID"**, **mLoginSuccess**.getRanchID())  
 .params(**"current"**, 0)  
 .params(**"pagesize"**, 99999)  
 .params(**"token"**, **mLoginSuccess**.getToken())  
 .params(**"username"**, **mUsername**)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String result = response.body().toString();  
 Gson gson1 = **new** Gson();  
 MuqunLoc muqunLoc = gson1.fromJson(result, MuqunLoc.**class**);  
 String msg = muqunLoc.getMsg();  
 **if** (msg.contains(**"按类型获取牲畜成功"**)) {  
  
 **livestockVarietyList** = muqunLoc.getLivestockVarietyList();  
 **mAdapter** = **new** MyAdapter(**livestockVarietyList**);  
 **mRecyclerView**.setAdapter(**mAdapter**);  
  
  
 }  
  
  
 }  
  
  
 });  
  
  
*//获取地图控件引用* **mMapView** = (MapView) findViewById(R.id.***bmapView***);  
 **mBaiduMap** = **mMapView**.getMap();  
  
  
  
  
  
 */\*// 开启定位图层  
 mBaiduMap.setMyLocationEnabled(true);  
  
 mLocation = new BDLocation();  
 mLocation.setLatitude(22.5366038785);  
 mLocation.setLongitude(113.9381825394);  
  
  
// 构造定位数据  
 MyLocationData locData = new MyLocationData.Builder()  
 .accuracy(mLocation.getRadius())  
 // 此处设置开发者获取到的方向信息，顺时针0-360  
// .direction(100)  
 .latitude(mLocation.getLatitude())  
 .longitude(mLocation.getLongitude()).build();  
  
// 设置定位数据  
 mBaiduMap.setMyLocationData(locData);  
  
// 设置定位图层的配置（定位模式，是否允许方向信息，用户自定义定位图标）  
  
 mCurrentMarker = BitmapDescriptorFactory  
 .fromResource(R.mipmap.icon\_location\_3);  
  
  
 MyLocationConfiguration config = new MyLocationConfiguration(  
 MyLocationConfiguration.LocationMode.FOLLOWING,  
 true, mCurrentMarker);  
 mBaiduMap.setMyLocationConfiguration(config);\*/* }  
  
  
 **public class** MyAdapter **extends** RecyclerView.Adapter<MyAdapter.ViewHolder> {  
  
 **public** List<MuqunLoc.LivestockVarietyListBean> **datas** = **null**;  
  
 **public** MyAdapter(List<MuqunLoc.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\_loc\_yang***, 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, **final int** position) {  
  
 viewHolder.**ivIcon**.setImageResource(R.mipmap.***loc\_yang***);  
 String deviceNo = **datas**.get(position).getDeviceNo();  
 deviceNo = deviceNo.substring(deviceNo.length() - 6, deviceNo.length());  
 viewHolder.**tvId**.setText(deviceNo);  
  
  
 }  
  
 *//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 **tvId**;  
 ImageView **ivIcon**;  
  
  
 **public** ViewHolder(View view) {  
 **super**(view);  
*// mTextView = (TextView) view.findViewById(R.id.text);* **tvId** = view.findViewById(R.id.***tvId***);  
 **ivIcon** = view.findViewById(R.id.***ivIcon***);  
  
  
 }  
 }  
  
  
 }  
  
  
 @Override  
 **protected void** onDestroy() {  
 **super**.onDestroy();  
 *//在activity执行onDestroy时执行mMapView.onDestroy()，实现地图生命周期管理* **mMapView**.onDestroy();  
 }  
  
 @Override  
 **protected void** onResume() {  
 **super**.onResume();  
 *//在activity执行onResume时执行mMapView. onResume ()，实现地图生命周期管理* **mMapView**.onResume();  
 }  
  
 @Override  
 **protected void** onPause() {  
 **super**.onPause();  
 *//在activity执行onPause时执行mMapView. onPause ()，实现地图生命周期管理* **mMapView**.onPause();  
 }  
  
  
}

**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.NiuLocActivity;  
**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** NiuListActivity **extends** AppCompatActivity {  
  
 **private static final** String ***TAG1*** = NiuListActivity.**class**.getSimpleName();  
  
 **private** List<QueryByYang.LivestockVarietyListBean> **mLivestockVarietyList**;  
  
 **int index** = 2;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_niu\_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"**, 2)  
 .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"**, 2)  
 .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(), NiuLocActivity.**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"**, 2)  
 .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(NiuListActivity.**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"**, 2)  
 .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(**"201"**)) {  
 viewHolder.**tvYangName**.setText(**"品种：西门塔尔牛"**);  
 } **else** {  
 viewHolder.**tvYangName**.setText(**"品种：xx牛"**);  
 }  
 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.Button;  
**import** android.widget.ImageView;  
  
**import** com.google.gson.Gson;  
**import** com.jinkun\_innovation.pastureland.R;  
**import** com.jinkun\_innovation.pastureland.bean.LoginSuccess;  
**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;  
  
*/\*\*  
 \* Created by Guan on 2018/4/11.  
 \*/***public class** PayConfirmActivity **extends** Activity{  
  
 String **mLogin\_success**;  
 LoginSuccess **mLoginSuccess**;  
 String **mUsername**;  
  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 setContentView(R.layout.***activity\_pay\_confirm***);  
  
 ImageView ivBack = (ImageView) findViewById(R.id.***ivBack***);  
 ivBack.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 finish();  
 }  
 });  
  
  
 **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**);  
  
 Button btnConfirm = findViewById(R.id.***btnConfirm***);  
 btnConfirm.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 Intent intent = getIntent();  
 String mDeviceNo = intent.getStringExtra(**"mDeviceNo"**);  
  
 OkGo.<String>*get*(Constants.***orderPay***)  
 .tag(**this**)  
 .params(**"token"**,**mLoginSuccess**.getToken())  
 .params(**"username"**,**mUsername**)  
 .params(**"deviceNo"**,mDeviceNo)  
 .execute(**new** StringCallback() {  
 @Override  
 **public void** onSuccess(Response<String> response) {  
  
 String result = response.body().toString();  
 **if** (result.contains(**"success"**)){  
  
 ToastUtils.*showShort*(**"确认订单支付成功"**);  
 setResult(***RESULT\_OK***);  
 finish();  
  
 }**else** {  
  
 ToastUtils.*showShort*(**"确认订单支付失败"**);  
  
  
  
 }  
  
  
 }  
 });  
  
  
  
  
 }  
 });  
  
  
 }  
  
  
}