|  |  |  |
| --- | --- | --- |
| 2013-12-10 | java用占位符拼接字符串：   1. String.format(formatter,args)：用%s为String类型占位； 2. MessageFormat.format(formatter,args)：用{0}做占位符，和C#用法一致。 | 梁志华 |
| 2013-12-11 | Google专为Android开发提供的Gson架包，使用toJson(Object obj)函数可以将对象序列化为json，也可以使用fromJson(String json,Class<T> classOfT)函数将json反序列化为对象，需要注意的是：传入的T必须是具体类型才能执行反序列化操作，否则序列化得到的对象为LinkedTreeMap类型，此类型无法直接转换为具体对象，强制转换会报错。 | 梁志华 |
| 2013-12-12 | C#下，判断某个Object对象的具体类型，使用is，例如：  string a=”hello world”;  if(a is string)  {  string str=a as string;  }  同理，在Java下，也可以通过instanceof来判断，例如：  String a=”hello world”;  if(a instanceof String){  String str=(String)a;  } | 梁志华 |
| 2014-01-03 | ListView在添加了HeaderView的情况下，在OnItemClick事件里，取得的position会从1开始，而不是从0开始，因此从集合中取数据时也要相应的减1。 | 梁志华 |
| 2014-01-03 | 在adapterView中有Button的情况下会导致onItemClickListener失效，解决办法：在adapterView最外层layout里面设置属性android:descendantFocusability="blocksDescendants"， 在adapterView的button上设置属性android:focusable="false"。 | 梁志华 |
| 2014-01-15 | 使用ListView、GridView等继承AbsListView的空间时，在修改某一列表项的数据后需通知界面刷新，使用notifyDataSetChanged()函数会造成界面卡顿一秒的现象，不建议使用，而是自定义Item项，修改数据的同时，修改自定义Item项的值。例如：  // 不建议使用这行notifyDataSetChanged()，这行刷新适配器会导致界面卡顿1秒  // shopImageItemAdapter.notifyDataSetChanged();  // 建议采用以下代码直接修改当前Item项的值来刷新界面  ((GridItem) view).setChecked(!isCanRemove); | 梁志华 |
| 2014-01-24 | 在ListView添加有HeaderView和FooterView的情况下，刷新数据会遇到数组越界的问题，报错的堆栈信息如下：  01-23 18:53:25.629: E/AndroidRuntime(3392): FATAL EXCEPTION: main  01-23 18:53:25.629: E/AndroidRuntime(3392): java.lang.IndexOutOfBoundsException: Invalid index 1, size is 1  01-23 18:53:25.629: E/AndroidRuntime(3392): at java.util.ArrayList.throwIndexOutOfBoundsException(ArrayList.java:251)  01-23 18:53:25.629: E/AndroidRuntime(3392): at java.util.ArrayList.get(ArrayList.java:304)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.widget.HeaderViewListAdapter.isEnabled(HeaderViewListAdapter.java:164)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.widget.ListView.dispatchDraw(ListView.java:3190)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.draw(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.widget.AbsListView.draw(AbsListView.java:3411)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewGroup.dispatchGetDisplayList(ViewGroup.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.View.getDisplayList(View.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.HardwareRenderer$GlRenderer.draw(HardwareRenderer.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewRootImpl.draw(ViewRootImpl.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewRootImpl.performTraversals(ViewRootImpl.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.view.ViewRootImpl.handleMessage(ViewRootImpl.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.os.Handler.dispatchMessage(Handler.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.os.Looper.loop(Looper.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at android.app.ActivityThread.main(ActivityThread.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at java.lang.reflect.Method.invokeNative(Native Method)  01-23 18:53:25.629: E/AndroidRuntime(3392): at java.lang.reflect.Method.invoke(Method.java:511)  01-23 18:53:25.629: E/AndroidRuntime(3392): at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at com.android.internal.os.ZygoteInit.main(ZygoteInit.java)  01-23 18:53:25.629: E/AndroidRuntime(3392): at dalvik.system.NativeStart.main(Native Method)  解决办法：   1. 当adapter的数据被全部clear时，使用adapter.notifyDataSetInvalidated(); 2. 在加入新的数据之后，使用adapter.notifyDataSetChanged()。   示例代码：  **if** (mIsRefresh) {  mOrderBillList.clear();  mAdapter.notifyDataSetInvalidated();  }  **for** (OrderBillModel orderBillModel : orderBillList) {  // 友好的方式显示时间  String createTime = FunctionHelper  .*getFriendlyTime*(orderBillModel.getCreateTime());  orderBillModel.setCreateTime(createTime);  mOrderBillList.add(orderBillModel);  }  mAdapter.notifyDataSetChanged(); | 梁志华 |