桂林电子科技大学

序号：

课号：1922137

**Android应用开发**  实验报告

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | 附加实验一 使用 ViewHolder 提升 ListView 显示性能 | | | | | | | |  | 辅导员意见：  成绩 辅导员  签 名 |
| 院 系 | 计算机与信息安全学院 | | | 专业 | | 计算机科学与技术 | | |
| 学 号 | 1800301037 | | | 姓名 | | 禤成伟 | | |
| 实验日期 | 2020 | 年 | 5 | | 月 | | 2 | 日 |
|  |  | | | | | | | |

|  |
| --- |
| **一、实验目的**  • 掌握 ListView 中 Item 布局加载机制；  • 掌握 View.setTag() 及 View.getTag() 读取数据的⽅式； |
| **二、实验内容** |
| 1.定义一个ViewHolder  2.将convetView的tag设置为ViewHolder,不为空时重新使用    图1. 运行结果  **三、实验总结**  问题：对Viewholder不太理解，不明白tag 是什么，有什么用。  解决方法：网络查询。  Android知识——ViewHolder的作用与用法：  <https://blog.csdn.net/weixin_30709929/article/details/94900226?utm_medium=distribute.pc_relevant.none-task-blog-OPENSEARCH-2&depth_1-utm_source=distribute.pc_relevant.none-task-blog-OPENSEARCH-2>  Android中ViewHolder模式开发的详细解释：  <https://blog.csdn.net/u014225510/article/details/46762743?utm_medium=distribute.pc_relevant.none-task-blog-BlogCommendFromBaidu-8&depth_1-utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromBaidu-8> |
| **四、实验代码**  表格1. 核心代码   |  | | --- | | public View getView(int position , View convertView , ViewGroup parent) {  News news = getItem(position);  View view;  ViewHolder viewHolder;  //当convertView不为空的时候  // 直接重新使用convertView从而减少了很多不必要的View的创建，然后加载数据  if (convertView == null) {  view = LayoutInflater.from(getContext()).inflate(resourceId , parent , false);  viewHolder = new ViewHolder();  //获取控件  viewHolder.textView = view.findViewById(R.id.tv\_title);  viewHolder.tvAuthor = view.findViewById(R.id.tv\_subtitle);  viewHolder.ivImage = view.findViewById(R.id.iv\_image);  // (1).Tag不像ID是用标示view的。Tag从本质上来讲是就是相关联的view的额外的信息。  // 它们经常用来存储一些view的数据，这样做非常方便而不用存入另外的单独结构。  // (2). 首先我们要知道setTag方法是干什么的：他是给View对象的一个标签。  // 标签可以是任何内容，我们这里把他设置成了一个对象，  // 因为我们是把vlist2.xml的元素抽象出来成为一个类ViewHolder，  // 用了setTag，这个标签就是ViewHolder实例化后对象的一个属性。  // 我们之后对于ViewHolder实例化的对象holder的操作，  // 都会因为java的引用机制而一直存活并改变convertView的内容，  // 而不是每次都是去new一个。我们就这样达到的重用。  view.setTag(viewHolder);  } else {  view = convertView;  viewHolder = (ViewHolder) view.getTag();  }  //重置控件  viewHolder.textView.setText(news.getTitle());  viewHolder.tvAuthor.setText(news.getmAuthor());  viewHolder.ivImage.setImageResource(news.getmImageId());  return view;  } | |