在 ListView、GridView 中使用 ContextMenu

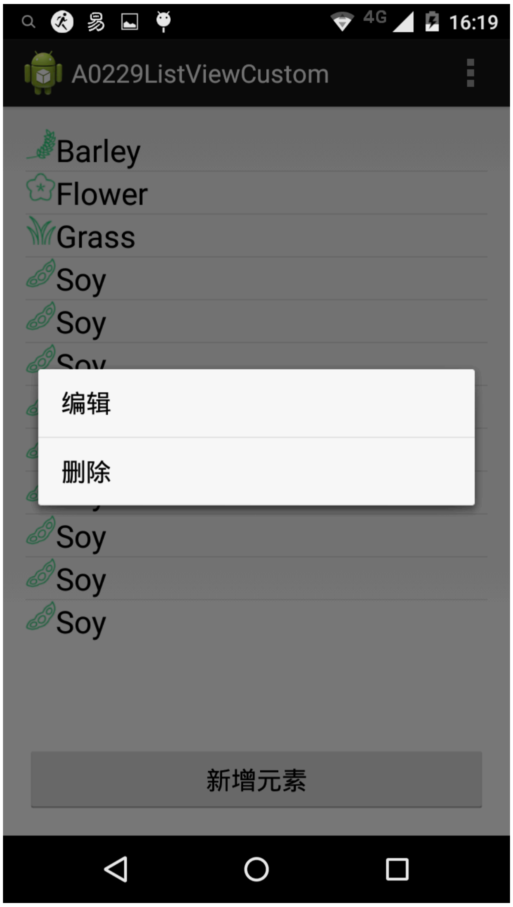
知识解析

ListView、GridView 中是最常用到上下文菜单的组件。但是，在这些组件中使用上下文菜单会有一个难题：一般上下文菜单中的选项是针对 ListView/GridView 中的某个选项的，如何确定上下文菜单是在哪个选项中被调用出来的?

使用 ContextMenuInfo（其实是其子类：AdapterView.AdapterContextMenuInfo）可以获得调出上下文菜单的时候所长按的 ListView/GridView 选项的 id、position 以及选项 view，此时根据 position 就可以知道哪个 ListView/GridView 选项上长按调出上下文菜单了。如下例：

```
 @Override
   public boolean onContextItemSelected(MenuItem item) {
        // TODO Auto-generated method stub
        AdapterContextMenuInfo              info              =
(AdapterContextMenuInfo) item.getMenuInfo();
        int pos = info.position;
        if(item.getItemId() == R.id.item2){
             list.remove(pos);
             adapter.notifyDataSetChanged();
        }
        return super.onContextItemSelected(item);
   }
```

功能演示



实战操作

```
public class MainActivity extends Activity {
    ArrayList<ImageText> list;
    MyAdapter adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        int[]  imgs  =  {  R.drawable.barley,  R.drawable.flower,
R.drawable.grapes, R.drawable.grass, R.drawable.hazelnut };
        String[] names = { "Barley", "Flower", "Grapes", "Grass",
"Hazelnut" };
```

```
        list = new ArrayList<ImageText>();
        for (int i = 0; i < imgs.length; i++) {
            ImageText it = new ImageText(imgs[i], names[i]);
            list.add(it);
        }
        adapter = new MyAdapter(this, 0, list);
        adapter.setNotifyOnChange(true);
        ListView lv = (ListView) this.findViewById(R.id.listView1);
        lv.setAdapter(adapter);
        Button btn = (Button) this.findViewById(R.id.button1);
        btn.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                adapter.add(new ImageText(R.drawable.soy, "Soy"));
            }
        });
        this.registerForContextMenu(lv);
        }
@Override
    public boolean onContextItemSelected(MenuItem item) {
        AdapterContextMenuInfo   info   =   (AdapterContextMenuInfo)
item.getMenuInfo();
        int pos = info.position;
        if(item.getItemId() == R.id.item2){
            list.remove(pos);
            adapter.notifyDataSetChanged();
        }
        return super.onContextItemSelected(item);
    }

    @Override
    public  void  onCreateContextMenu(ContextMenu  menu,  View  v,
ContextMenuInfo menuInfo) {
        getMenuInflater().inflate(R.menu.list_menu, menu);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.action_settings) {
            return true;}
        return super.onOptionsItemSelected(item);}
```

```
class ImageText {
        private int imageId;
        private String text;
        public ImageText(int imageId, String text) {
            this.setImageId(imageId);
            this.setText(text);}
        public int getImageId() {
            return imageId;}
        public void setImageId(int imageId) {
            this.imageId = imageId;}
        public String getText() {
            return text;}
        public void setText(String text) {
            this.text = text;}
```

```
    }
    class MyAdapter extends ArrayAdapter<ImageText> {
        List<ImageText> list;

        public MyAdapter(Context context, int resource, List<ImageText>
objects) {
            super(context, resource, objects);
                    this.list = objects;}
        @Override
        public int getCount() {
            return list.size();}
        @Override
        public long getItemId(int position) {
                return position;
}
@Override
        public View getView(int position, View convertView, ViewGroup
parent) {
            View view = null;
            ViewHolder vh = null;
            if (convertView == null) {
                LayoutInflater inflater = (LayoutInflater)
MainActivity.this.getSystemService(LAYOUT_INFLATER_SERVICE);
                view = inflater.inflate(R.layout.item, parent, false);
                ImageView iv = (ImageView)
view.findViewById(R.id.imageView1);
                TextView tv = (TextView)
view.findViewById(R.id.textView1);
                vh = new ViewHolder();
                vh.vh_iv = iv;
                vh.vh_tv = tv;
                view.setTag(vh);} else {
```

```
                view = convertView;
                vh = (ViewHolder) view.getTag();}
            ImageText it = list.get(position);
            vh.vh_iv.setImageResource(it.getImageId());
            vh.vh_tv.setText(it.getText());
            return view;}
        class ViewHolder {
            TextView vh_tv;
            ImageView vh_iv;
        }}}
```

职业素质

在开发中我们需要记住 Options Menu 的拥有者是 Activity，而上下文菜单的拥有者是 Activity 中的 view。ListView、GridView 中是最常用到上下文菜单的组件。但是，在这些组件中使用上下文菜单会有一个难题：一般上下文菜单中的选项是针对 ListView/GridView 中的某个选项的，