**06\_事件机制**

🖊 人不光是靠他生来就拥有一切，而是靠他从学习中所得到的一切来造就自己。 —— 歌德

**相关课程资料：**

鼠标放置在正文左侧，点击“**+**”工具栏中的“**高亮块**”来插入高亮块。

|  |
| --- |
| 科目：Android  日期：03/27  主题：品牌定位  课前阅读： @ 插入相关文档 |

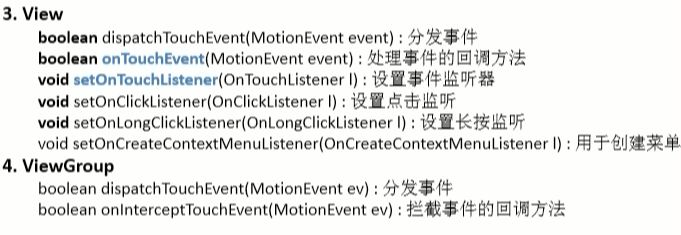
**提出问题？**

1. Android是如何相应对其中某个【视图】的操控操作的呢？----》 【MotionEvent】
2. Android是如何相应对手机的【按键】的操作呢？ ----》【KeyEvent】

**屏幕触控（MotionEvent）机制**

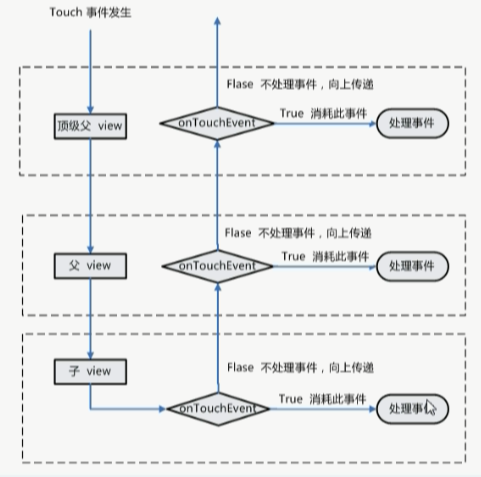
1. **事件类型**
2. down
3. Move
4. up
5. **事件的发生顺序**
6. Down
7. Move
8. Move
9. Move
10. ....
11. Up
12. **API**

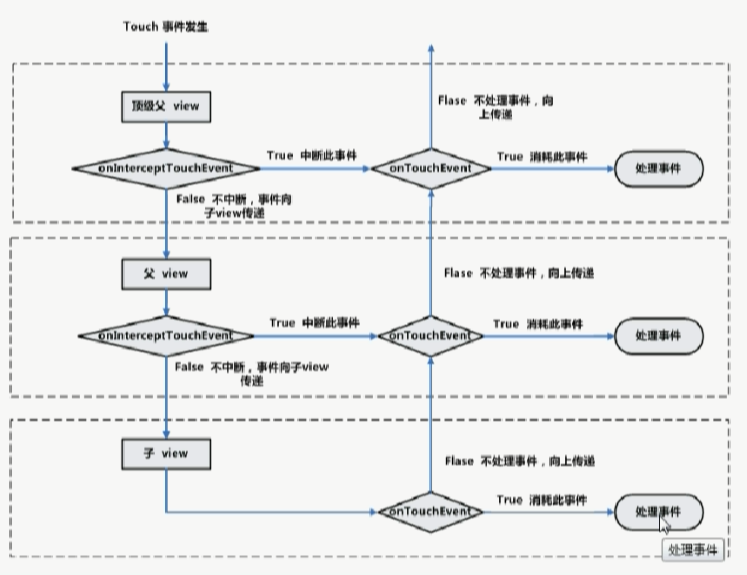




1. **操作事件**
2. 分发：dispatchTouchEvent（）
3. 处理：onTouchEvent（）、onTouch（）
4. 消费：处理事件的方法返回true，当前事件对象就结束了

**事件分发和处理流程**

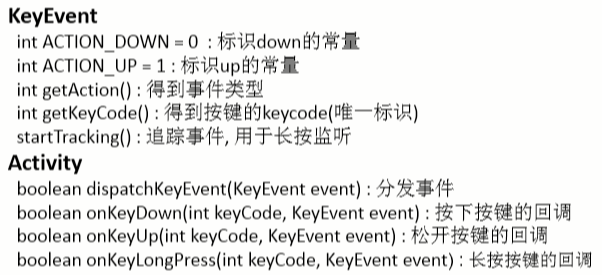




1. 事件产生的顺序为：down--->move--->move...-->up
2. 事件对象被系统创建后，首先会调用对应Activity对象的dispatchTouchEvent（）
3. down在分发给视图对象的过程中要确定消费者（onTouchEvent（）返回true），如果都返回false，那事件的消费者只能是Activity了
4. 后面的move和up事件，将事件分发给消费者（可能是视图对象，也可能是Activity）处理，如果视图不消费，直接交给Activity处理消费
5. 每个事件都要有一个消费者

**按键（KeyEvent）事件**

1. **事件类型**
2. Down
3. Up
4. **API**



1. **应用练习**
2. **拖动图片**
3. 事件处理
4. 得到视图对象的坐标（左下角 / 右下角，left | top | right | bottom）
5. 动态给视图对象进行重新布局定位：view.layout（l，t，r，v）
6. **连续点击back才退出应用**
7. keyEvent的处理
8. Handler处理延迟工作