

HenCoder Plus 讲义

多点触控的原理和写法全解析

MotionEvent.getActionMasked()

常见值：

- ACTION_DOWN 第一个手指按下（之前没有任何手指触摸到 View）
- ACTION_UP
最后一个手指抬起（抬起之后没有任何手指触摸到 View，这个手指未必是 ACTION_DOWN 的那个手指）
- ACTION_MOVE 有手指发生移动
- ACTION_POINTER_DOWN 额外手指按下（按下之前已经有别的手指触摸到 View）
- ACTION_POINTER_UP 有手指抬起，但不是最后一个（抬起之后，仍然还有别的手指在触摸着 View）

MotionEvent.getActionIndex(index)

获取「非第一根按下手指」或「非最后一根抬起手指」的 index

触摸事件的结构

- 触摸事件是按序列来分组的，每一组事件必然以 ACTION_DOWN 开头，以 ACTION_UP 或 ACTION_CANCEL 结束。
- ACTION_POINTER_DOWN 和 ACTION_POINTER_UP 和 ACTION_MOVE 一样，只是事件序列中的组成部分，并不会单独分出新的事件序列
- 触摸事件序列是针对 View 的，而不是针对 pointer 的。「某个 pointer 的事件」这种说法是不正确的。
- 在一个触摸时间里，每个 Pointer 除了 x 和 y 之外，还有 index 和 id。index 和

id 分别是什么、怎么用？去课程里回顾一下，很重要！

- 「移动的那个手指」这个概念是伪概念，「寻找移动的那个手指」这个需求是个伪需求。

多点触控的三种类型

- 接力型 同一时刻只有一个 pointer 起作用，即最新的 pointer。典型：ListView、RecyclerView。实现方式：在 ACTION_POINTER_DOWN 和 ACTION_POINTER_UP 时记录下最新的 pointer，在之后的 ACTION_MOVE 事件中使用这个 pointer 来判断位置。
- 配合型 / 协作型 所有触摸到 View 的 pointer 共同起作用。
典型：ScaleGestureDetector，以及 GestureDetector 的 onScroll() 方法判断。实现方式：在每个 DOWN、POINTER_DOWN、POINTER_UP、UP 事件中使用所有 pointer 的坐标来共同更新焦点坐标，并在 MOVE 事件中使用所有 pointer 的坐标来判断位置。
- 各自为战型 各个 pointer 做不同的事，互不影响。典型：支持多画笔的画板应用。实现方式：在每个 DOWN、POINTER_DOWN 事件中记录下每个 pointer 的 id，在 MOVE 事件中使用 id 对它们进行跟踪。

问题和建议？

课上技术相关的问题，都可以去群里和大家讨论，对于比较通用的、有价值的问题，可以去我们的知识星球提问。

具体技术之外的问题和建议，都可以找丢物线（微信：diuwuxian），丢丢会为你解答技术以外的一切。



觉得好？

如果你觉得课程很棒，欢迎给我们好评呀！<https://ke.qq.com/comment/index.html?cid=381952>

一定要是你真的觉得好，再给我们好评。不要仅仅因为对扔物线的支持而好评（报名课程已经是你最大的支持了，再不够的话 B 站多来点三连我也很开心），另外我们也坚决不做好评返现等任何的交易。我们只希望，在课程对你有帮助的前提下，可以看到你温暖的评价。

更多内容：

- 网站：<https://hencoder.com>；<https://kaixue.io>
- 各大搜索引擎、微信公众号、微博、知乎、掘金、哔哩哔哩、YouTube、西瓜视频、抖音、快手、微视：统一账号「扔物线」，我会持续输出优质的技术内容，欢迎大家关注。
- 哔哩哔哩快捷传送门：<https://space.bilibili.com/27559447>

大家如果喜欢我们的课程，还请去扔物线的哔哩哔哩，帮我素质三连，感谢大家！