

HenCoder Plus 第 20 课 讲义

拖拽和嵌套滑动

拖拽

OnDragListener

- 通过 `startDrag()` 来启动拖拽
- 用 `setOnDragListener()` 来监听
 - `OnDragListener` 内部只有一个方法: `onDrag()`
 - `onDragEvent()` 方法也会收到拖拽回调 (界面中的每个 `View` 都会收到)

ViewDragHelper

- 需要创建一个 `ViewDragHelper` 和 `Callback()`
- 需要写在 `ViewGroup` 里面, 重写 `onIntercept()` 和 `onTouchEvent()`

为什么要这两个东西, 而不是一个?

- `OnDragListener`
 - API 11 加入的工具类, 用于拖拽操作。
 - 使用场景: 用户的「拖起 -> 放下」操作, 重在内容的移动。可以附加拖拽数据
 - 不需要写自定义 `View`, 使用 `startDrag()` / `startDragAndDrop()` 手动开启拖拽
 - 拖拽的原理是创建一个图像在屏幕的最上层, 用户的手指拖着图像移动
- `ViewDragHelper`
 - 2015 年的 `support v4` 包中新增的工具类, 用于拖拽操作。
 - 使用场景: 用户拖动 `ViewGroup` 中的某个子 `View`
 - 需要应用在自定义 `ViewGroup` 中调用 `ViewDragHelper.shouldInterceptTouchEvent()` 和 `processTouchEvent()`, 程序会自动开启拖拽
 - 拖拽的原理是实时修改被拖拽的子 `View` 的 `mLeft`, `mTop`, `mRight`, `mBottom` 值

嵌套滑动

嵌套滑动的场景

- 不同向嵌套
 - `onInterceptTouchEvent` 父 `View` 拦截
 - `requestDisallowInterceptTouchEvent()` 子 `View` 阻止父 `View` 拦截
- 同向嵌套

- 父 View 会彻底卡住子 View
 - 原因：抢夺条件一致，但 父 View 的 `onInterceptTouchEvent()` 早于子 View 的 `dispatchTouchEvent()`
- 本质上是策略问题：嵌套状态下用户手指滑动，他是想滑谁？
 - 场景一：NestedScrollView
 - 子 View 能滑动的时候，滑子 View；滑不动的时候，滑父 View
 - 场景二：Google 的样例
 - 父 View 展开的时候：
 - 上滑：优先滑父 View
 - 下滑：滑不动（所以可以说还是优先滑父 View）
 - 父 View 半展开的时候：
 - 向上：优先滑父 View，滑到父 View 完全收齐之后开始滑子 View
 - 向下：滑父 View，滑到父 View 完全展开之后开始滑子 View
 - 父 View 收起的时候：
 - 上滑：滑子 View（所以可以说是优先滑子 View）
 - 下滑：优先滑子 View，滑到子 View 顶部的时候开始滑父 View
- 解决方案：实现策略——父 View、子 View 谁来消费事件可以实时协商
- 实现？
 - 其实大多数场景，SDK 就解决了
 - 案例 1：ScrollView 嵌套：没法滑动
 - 换成 NestedScrollView：可以滑动
 - 自己实现？
 - 按照课上代码，实现 `NestedScrollingChild2` 接口来实现自定义的嵌套滑动逻辑

问题和建议？

课上技术相关的问题，都可以在学员群里和大家讨论，我一旦有时间也都会来解答。如果我没来就 @我一下吧！

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。



- # HenCoder

微信公众号: HenCoder
 微博: 扔物线
 知乎专栏: HenCoder
 稀土掘金: 扔物线
<http://hencoder.com>

