3.1 View

作为创建 UI 时最基础的组件，`View` 是一个支持 Flexbox 布局、样式、一些触摸处理、和一些无障碍功能的容

器，不论在什么平台上，View 都会直接对应平台的原生视图，无论它是 `UIView`、`<div>`、`android.view` 等等

`View` 可以放到其它的视图里，也可以有任意多个任意类型的子视图。

下面的例子创建了一个 `View`，包含了两个有颜色的方块和一个自定义的组件，并且设置了一个内边距：

```
class ViewColoredBoxesWithText extends Component {
  render() {
    return (
      <View style={{flexDirection: 'row', height: 1, padding:
0.2}}>
        <View style={{backgroundColor: 'blue', flex: 0.3}} />
        <View style={{backgroundColor: 'red', flex: 0.5}} />
        <Text>Hello World!</Text>
      </View>
    );
  }
}
```

`View` 设计初衷是要和 `StyleSheet` 搭配使用，这样可以是代码更清晰，性能更高，内联的样式也是可

以使用的。

属性

hitSlop PropTypes.oneOfType([

PropTypes.number,

EdgeInsetsPropType,

])

定义了触摸事件离 View 有多远的距离,一般触摸目标的参考值是 30-40 点/像素密度，决定于像素值。

举个例子，如果一个触摸的 View 有 20 的触摸高度，可以用

`hitSlop={{top: 10, bottom: 10, left: 0,}}`

`right: 0}}`扩展到 40。

触摸区域从来不会超出父组件的边界，如果触摸到两个重叠的 View 时，相邻组件的 Z-idnex 总是

会优先考虑的，也就是说先根据 Z-index 判断哪个 View 在上面

onLayout

PropTypes.func

加载完之后回调，可以用下面的代码改变：

`{nativeEvent: { layout: {x, y, width, height}}}`

这个事件会在布局计算完成后立即调用一次，不过收到此事件时新的布局可能还没有在屏幕上呈现，尤

其是一个布局动画正在进行中的时候。

pointerEvents PropTypes.oneOf([

'box-none',

'none',

'box-only',

'auto',

])

控制 View 是否成为触摸事件的目标。

- `'auto'`：View 可以成为触摸事件的目标。
- `'none'`：View 从不成为触摸事件的目标。
- `'box-none'`：View 不成为触摸事件的目标，但子 View 可以，就跟下面的 CSS 样式一样：

```
.box-none {
  pointer-events: none;
}
.box-none * {
  pointer-events: all;
}
```

- `'box-only'`：View 可以成为触摸事件的目标，但子 View 不可以，就跟下面的 CSS 样式一样：

```
.box-only {
```

```
  pointer-events: all;
}
.box-only * {
  pointer-events: none;
}
```

因为 `pointerEvents` 不影响布局，所以我们单独提供了这个方法，并没有把 `pointerEvents` 放在

`style` 里面．在有些平台上，我们需要把它声明成一个 `className`．用不用 `style` 取决于平台．

style style

Layout Props... 查看左侧导航的"布局属性"

Shadow Props...

Transforms... 查看左侧导航的"三维坐标、变换"

backgroundColor color

borderColor color

borderRadius number

borderWidth number

opacity number

vr **enableGazeCollision** PropTypes.bool

Allows the control of gaze collision for this view and all child views, when not enabled no collision checks will be performed within the system. It would be as if that view was not there