

# 《Android 开发从入门到实战》

## 高级控件

让人人都能享受到高品质的教育服务

## 目 录

下拉列表

A

列表类视图

B

翻页类视图

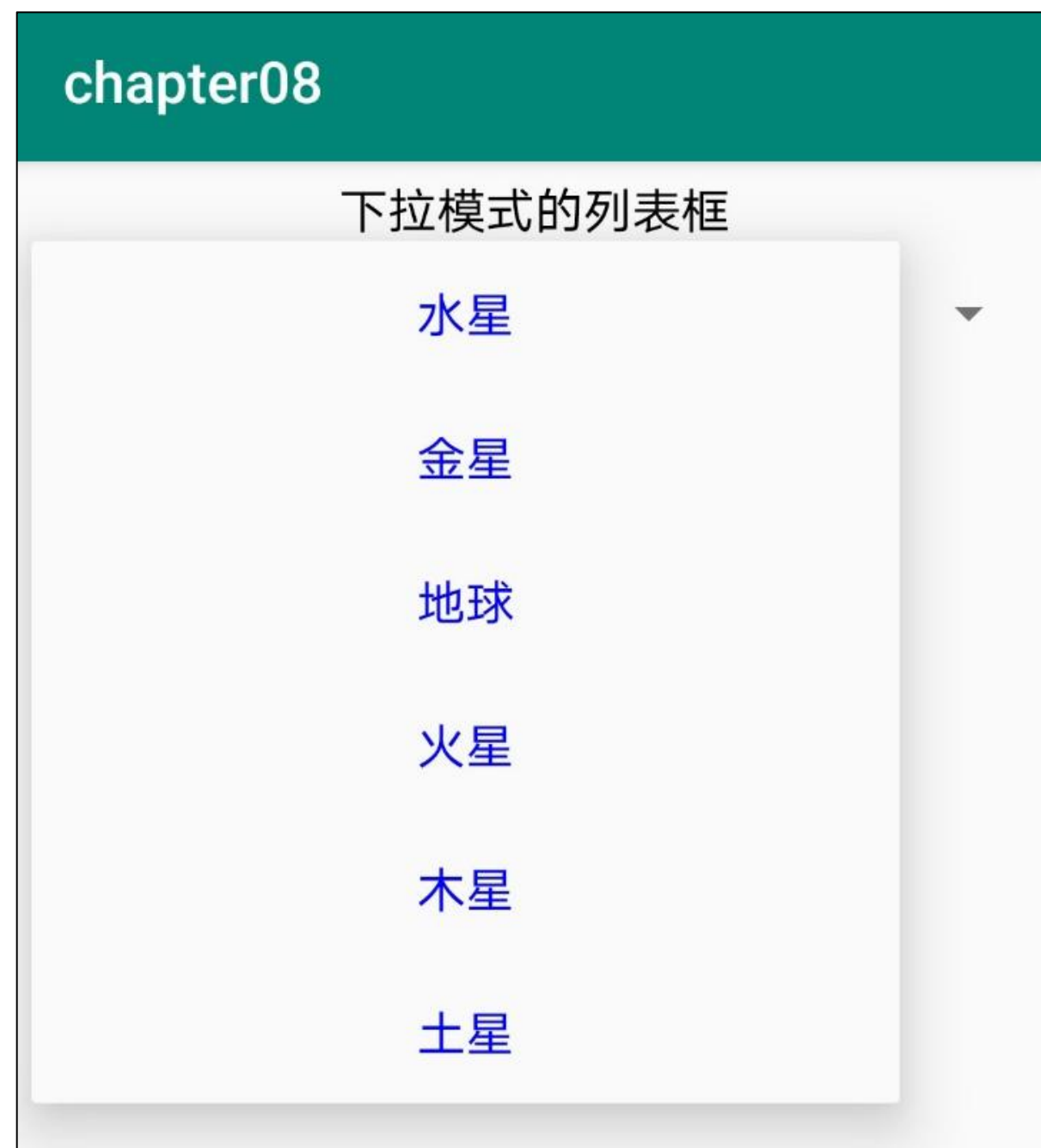
C

Fragment

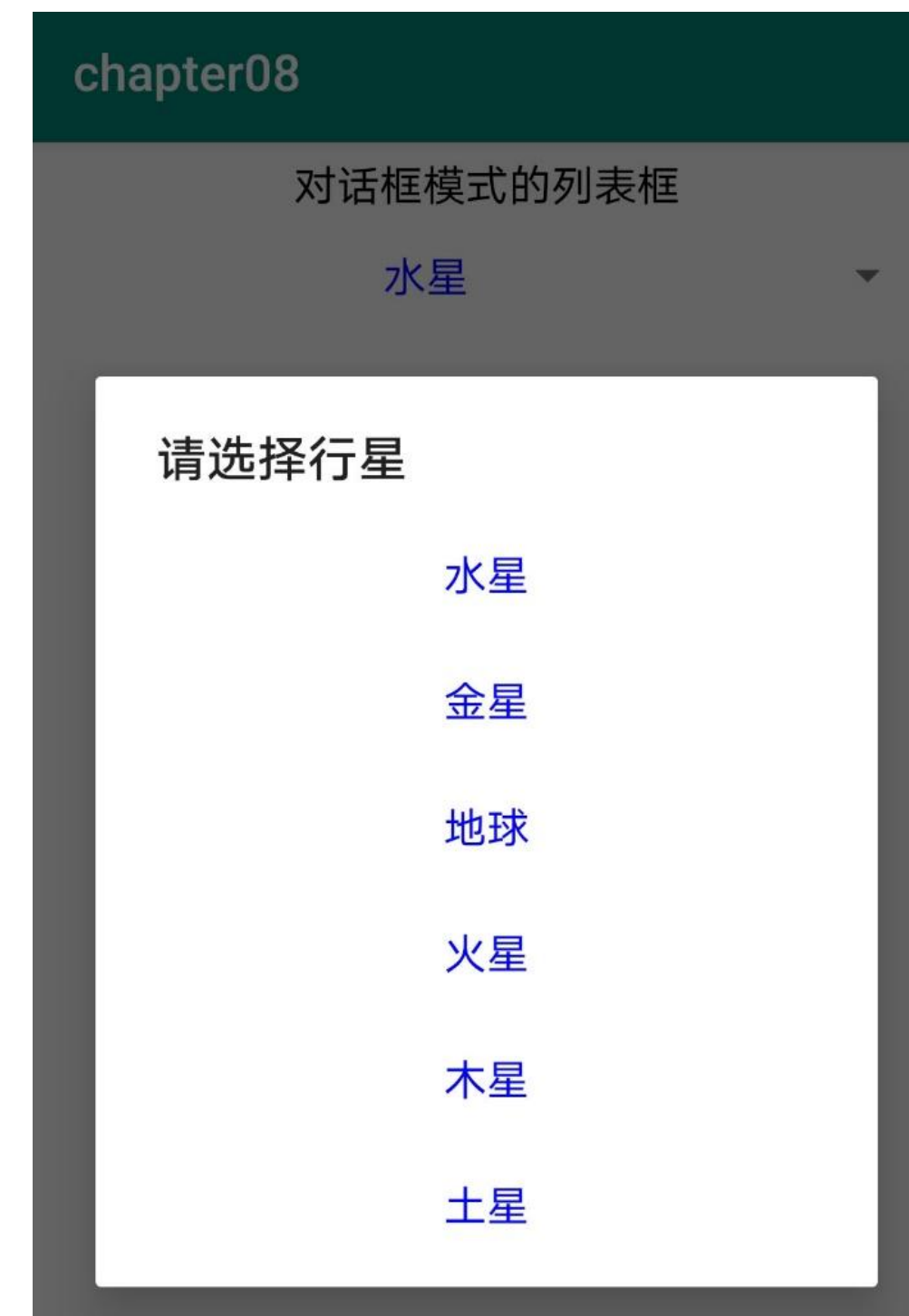
D

# 下拉列表 Spinner

- Spinner 用于从一串列表中选择某项，功能类似于单选按钮的组合。



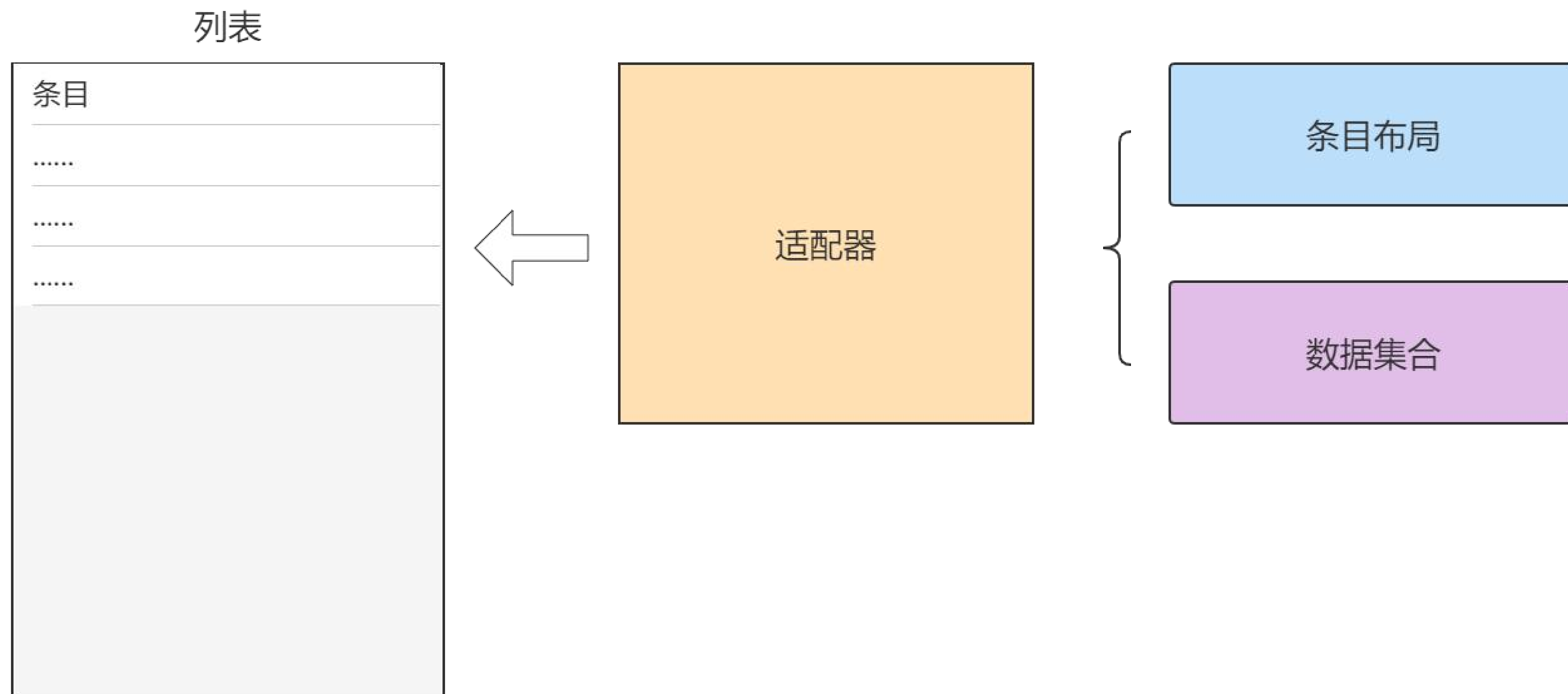
下拉列表形式



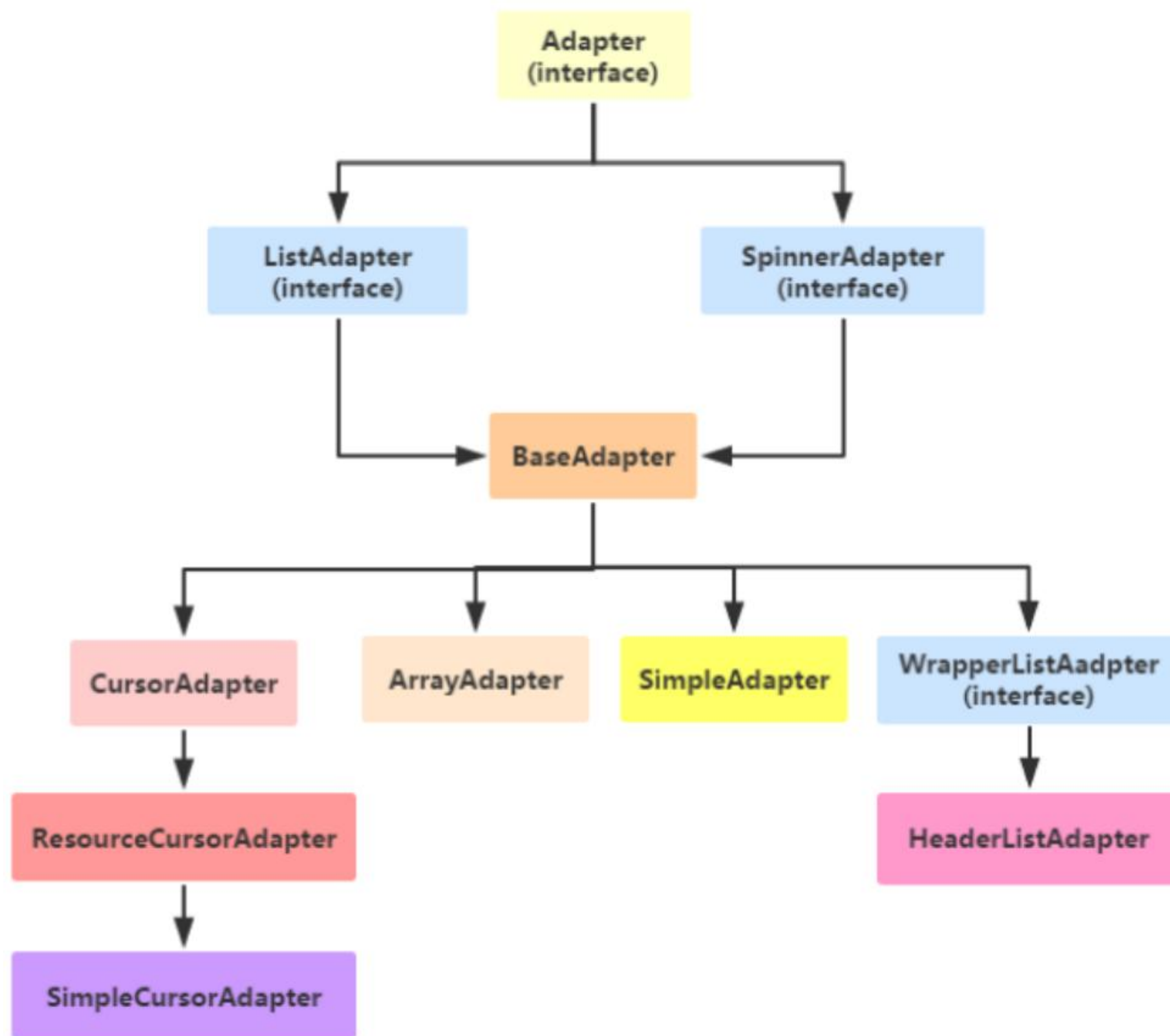
对话框形式

# 适配器 Adapter

- 适配器负责从数据集中取出对应的数据显示到条目布局上。



# Adapter 继承结构

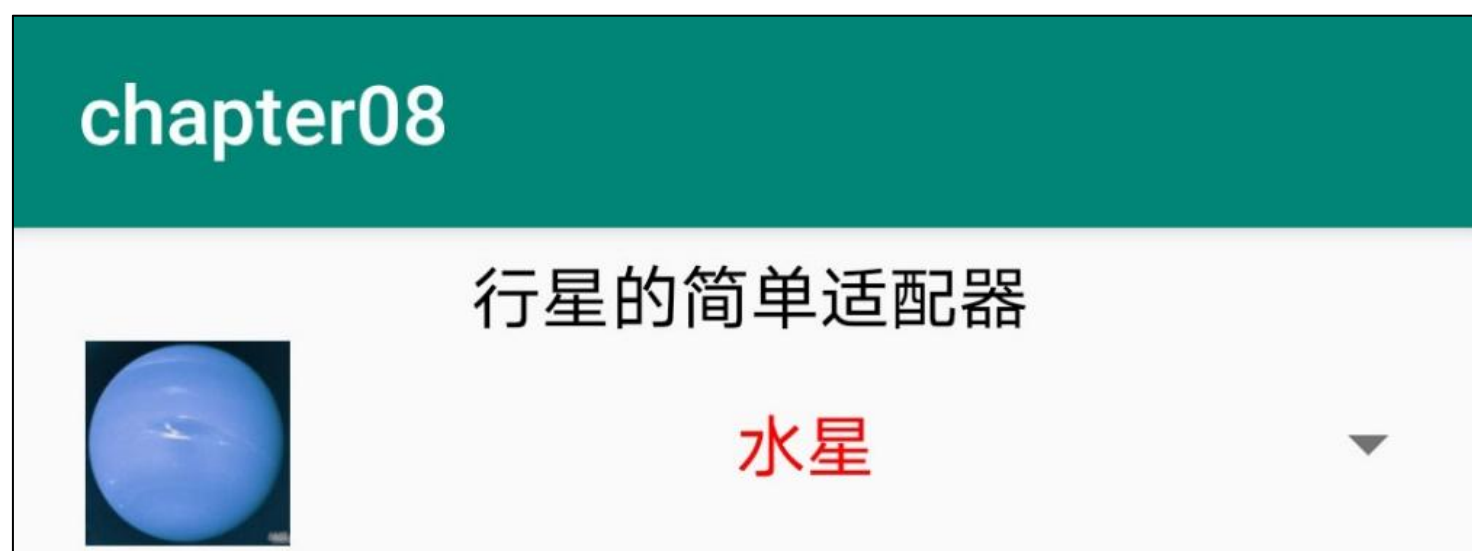


# 数组适配器 ArrayAdapter

- 最简单的适配器，只展示一行文字。
- 运用数组适配器分成下列步骤：
  - 编写列表项的XML文件，内部布局只有一个TextView标签
  - 调用ArrayAdapter的构造方法，填入待展现的字符串数组，以及列表项的XML文件（R.layout.item\_select）
  - 调用下拉框控件的setAdapter方法，传入第二步得到的适配器实例

# 简单适配器 SimpleAdapter

- SimpleAdapter 允许在列表项中同时展示文本与图片。



采用简单适配器的初始下拉框

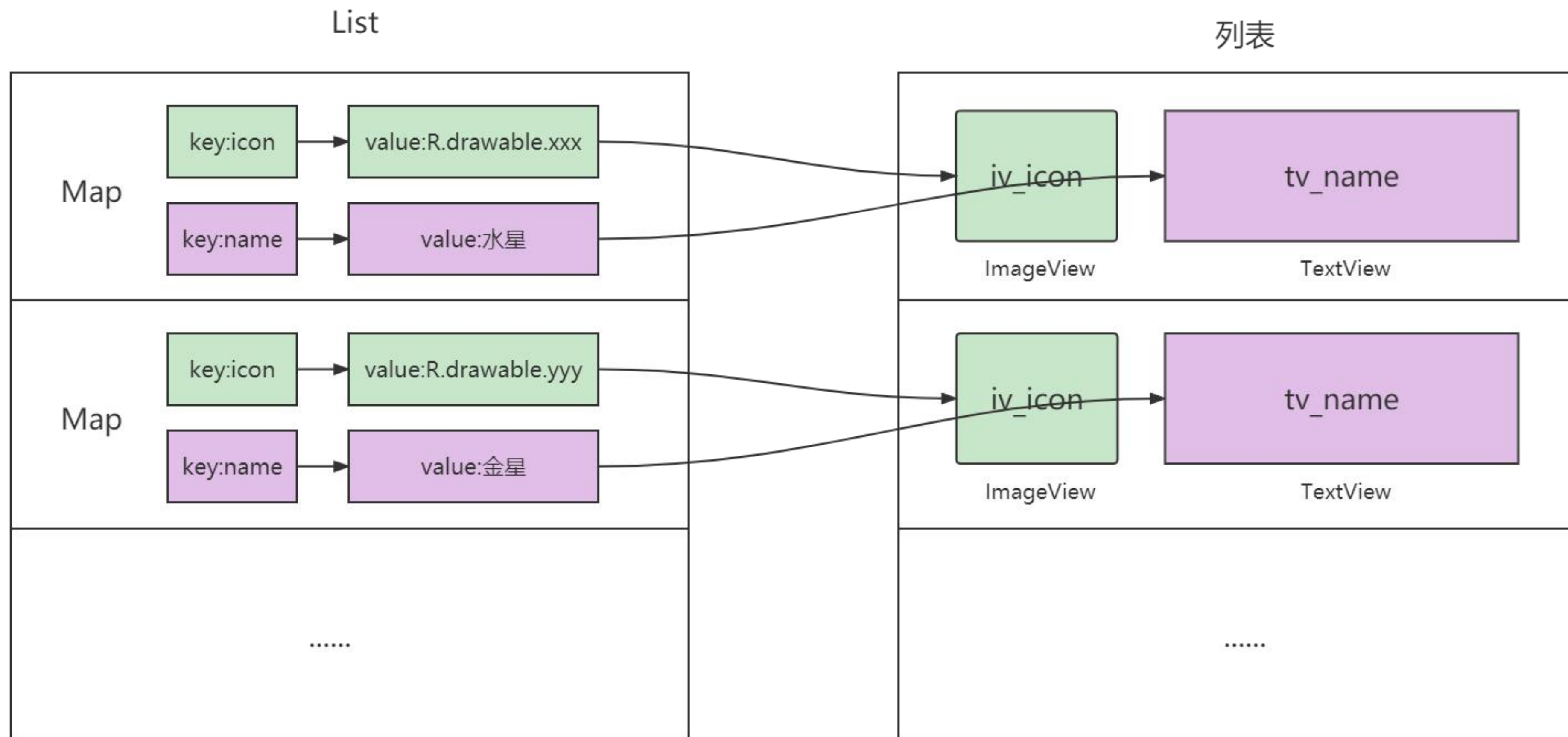


采用简单适配器的列表对话框



# 简单适配器 SimpleAdapter

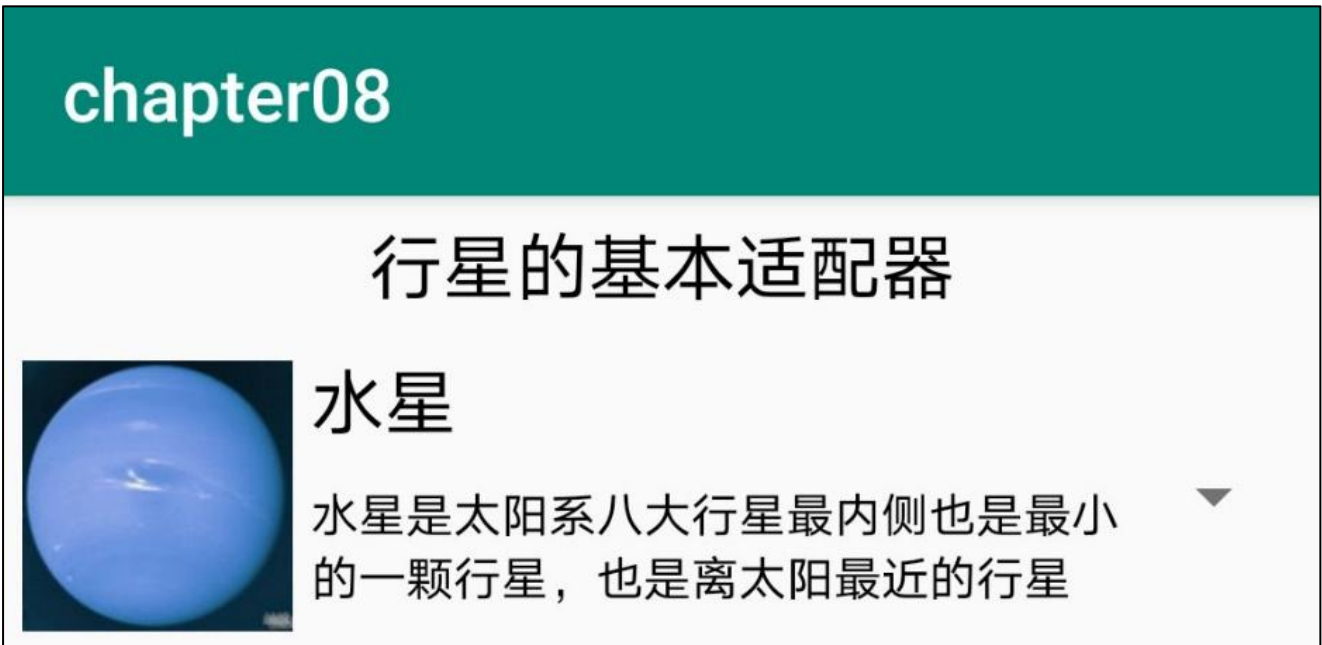
- 集合当中的数据与条目布局的对应关系。



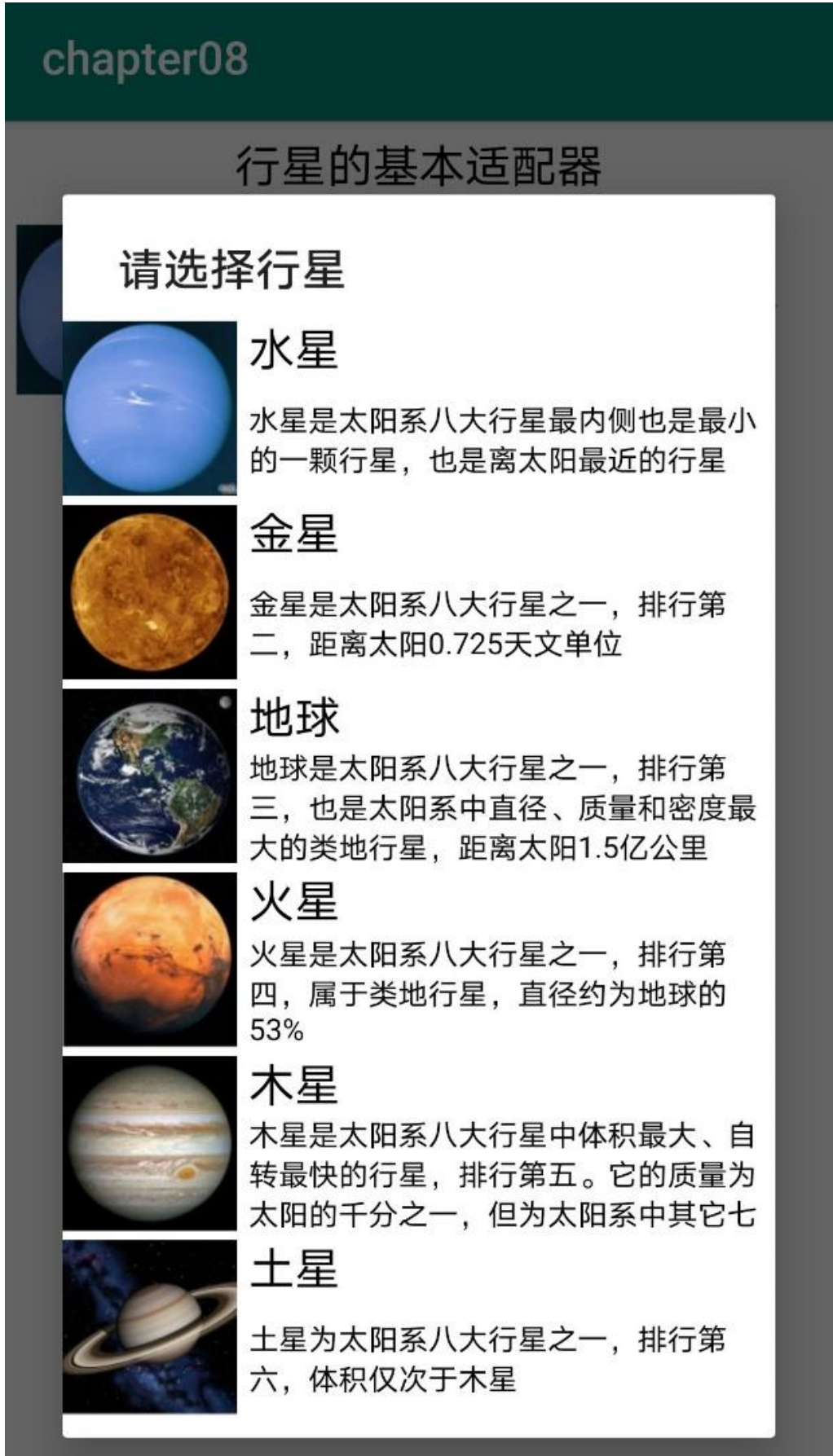


# 基本适配器 BaseAdapter

➤ BaseAdapter 是一种适应性更强的基本适配器。



采用基本适配器的初始下拉框



采用基本适配器的列表对话框

# 基本适配器 BaseAdapter

## ➤ 条目与数据集合对应关系

列表

条目 getView(position = 0)

条目 getView(position = 1)

条目 getView(position = 2)

条目 getView(position = 3)

.....

集合

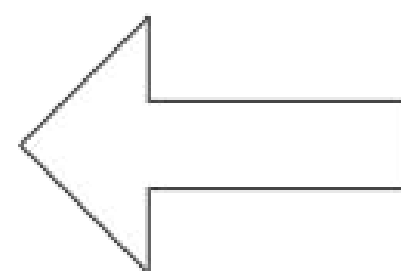
index = 0

index = 1

index = 2

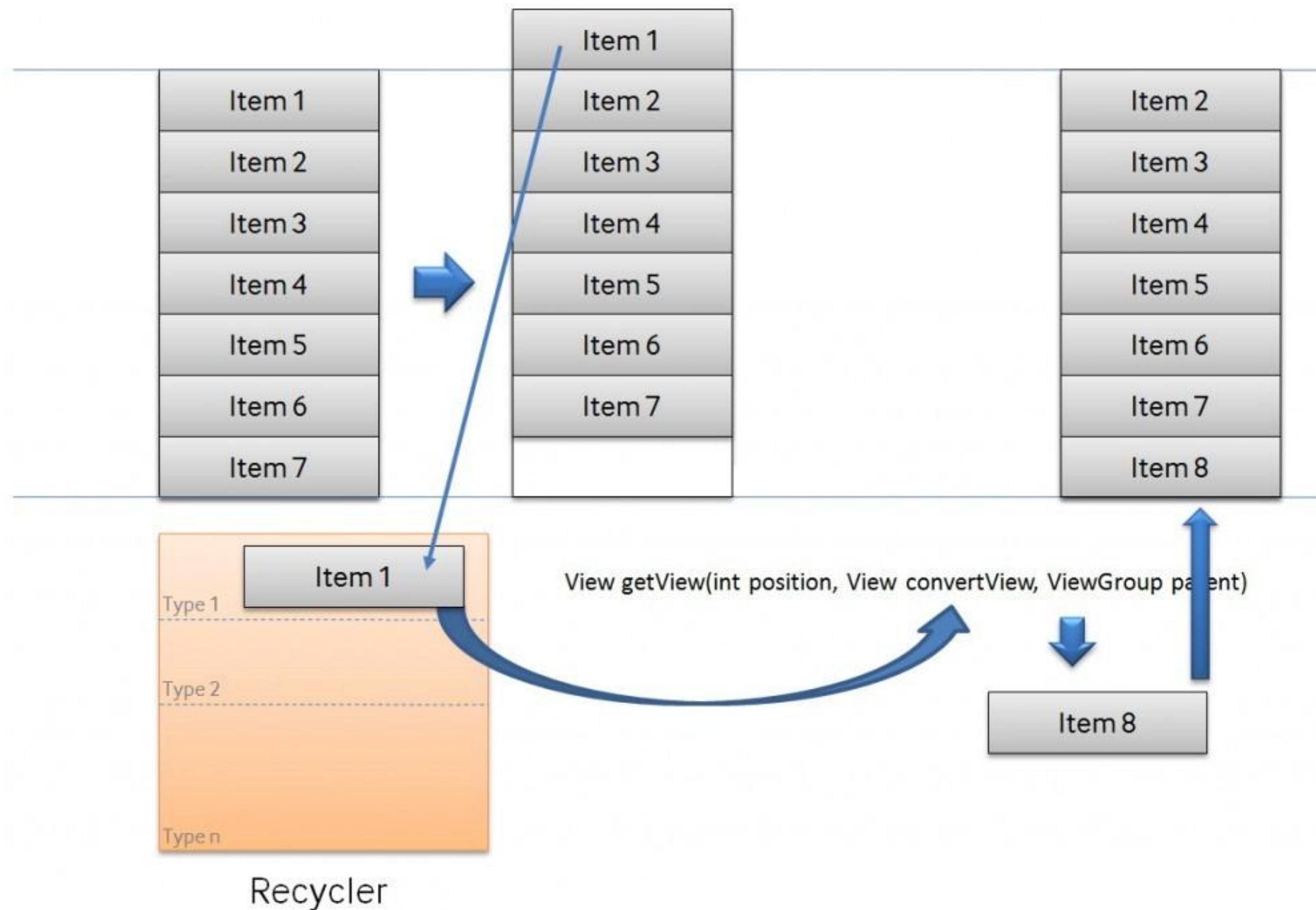
index = 3

.....



# 复用 convertView

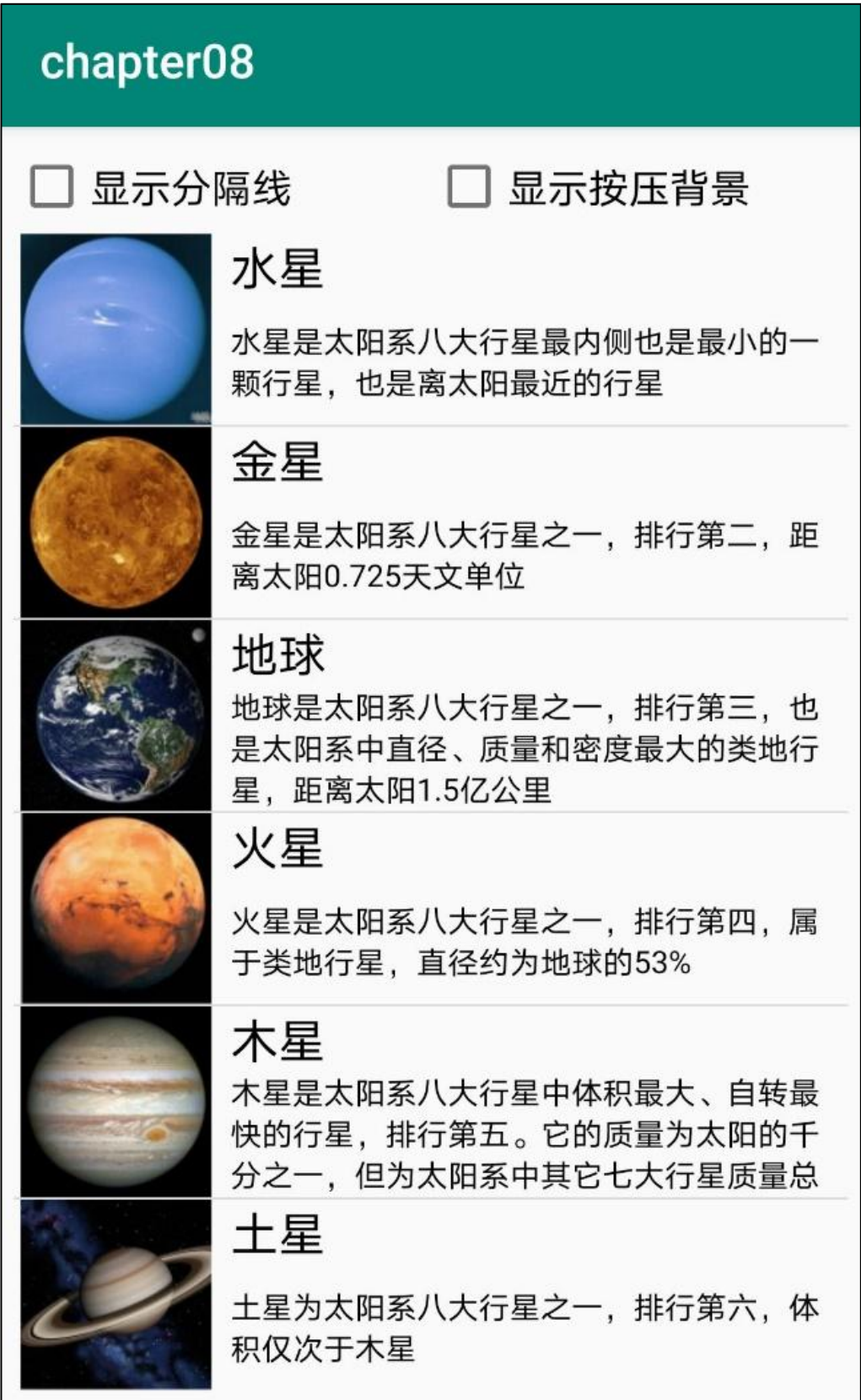
➤ 当列表的Item从上方滚出屏幕视角之外时:





# 列表视图 ListView

➤ ListView 允许在页面上分行展示数据列表，例如新闻列表、商品列表等，方便用户浏览与操作。



采用基本适配器的列表视图



利用列表视图改造购物车界面

# ListView 新增属性

- 修改分隔线样式要在XML文件中同时设置divider与dividerHeight两个属性。
- 若想取消按压列表项之时默认的水波背景，可在XML文件中设置也可在Java代码中设置。

XML中的属性	ListView类的设置方法	说明
divider	setDivider	指定分隔线的图形。如需取消分隔线，可将该属性值设为@null
dividerHeight	setDividerHeight	指定分隔线的高度
listSelector	setSelector	指定列表项的按压背景（状态图形格式）

# 网格视图 GridView

- 网格视图用于分行分列显示表格信息。



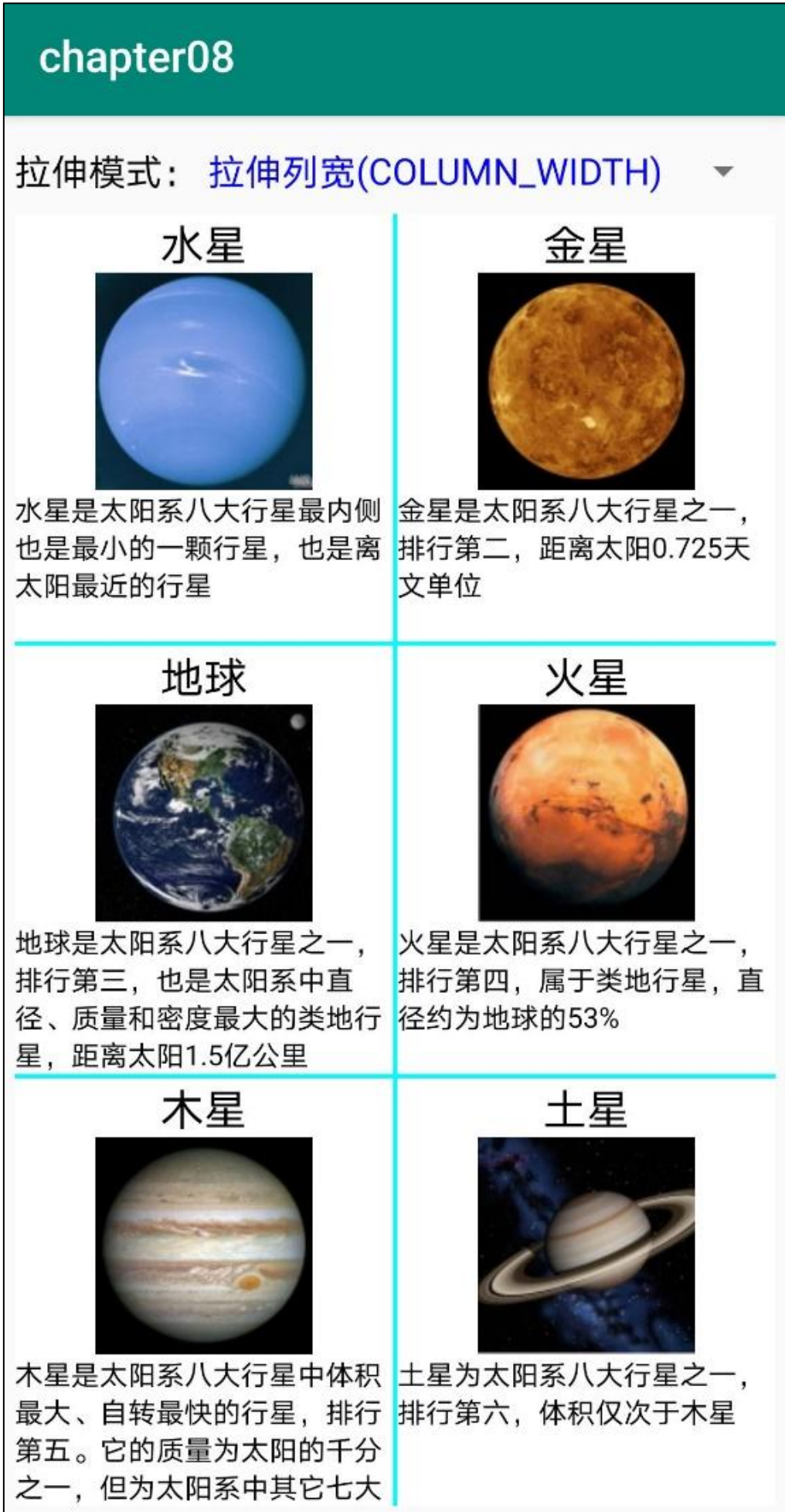


# GridView 新增属性与方法

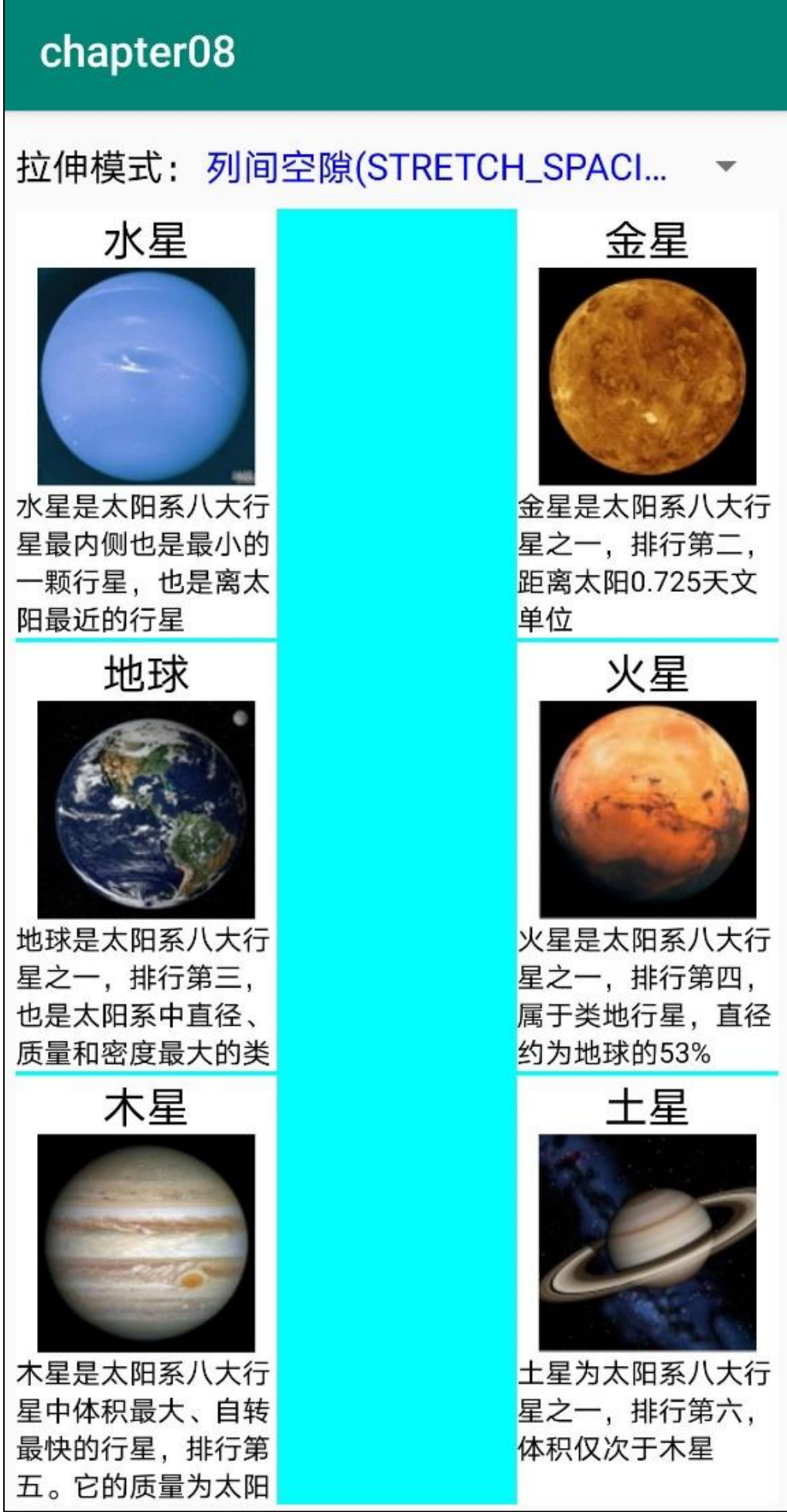
XML中的属性	GridView类的设置方法	说明
horizontalSpacing	setHorizontalSpacing	指定网格项在水平方向的间距
verticalSpacing	setVerticalSpacing	指定网格项在垂直方向的间距
numColumns	setNumColumns	指定列的数目
stretchMode	setStretchMode	指定剩余空间的拉伸模式
columnWidth	setColumnWidth	指定每列的宽度。拉伸模式为spacingWidth、spacingWidthUniform时，必须指定列宽



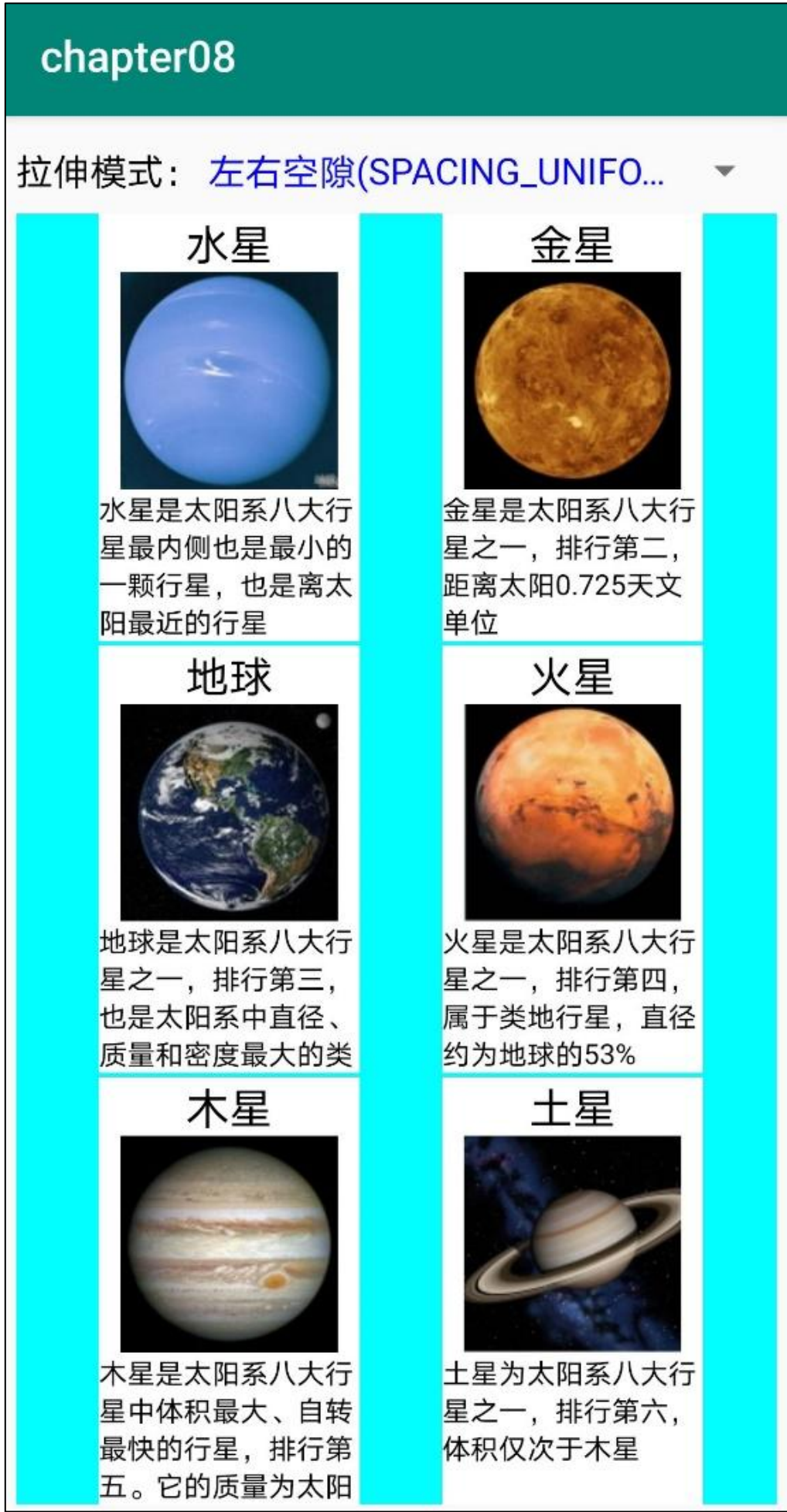
# GridView 拉伸模式效果



columnWidth的拉伸效果



spacingWidth的拉伸效果



spacingWidthUniform的拉伸效果

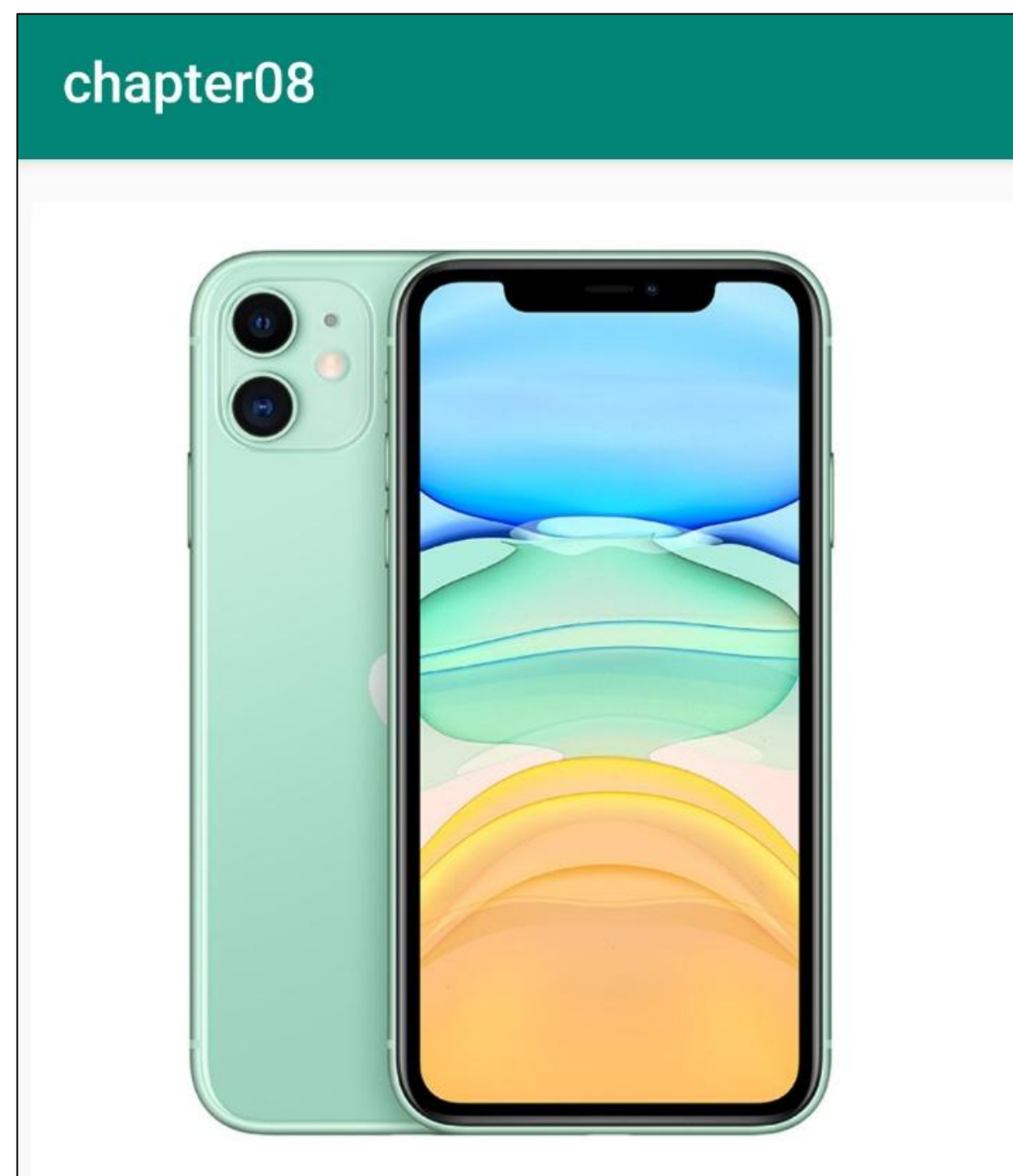
# GridView 拉伸模式取值

XML中的拉伸模式	GridView类的拉伸模式	说明
none	NO_STRETCH	不拉伸
columnWidth	STRETCH_COLUMN_WIDTH	若有剩余空间，则拉伸列宽挤掉空隙
spacingWidth	STRETCH_SPACING	若有剩余空间，则列宽不变，把空间分配到每列间的空隙
spacingWidthUniform	STRETCH_SPACING_UNIFORM	若有剩余空间，则列宽不变，把空间分配到每列左右的空隙

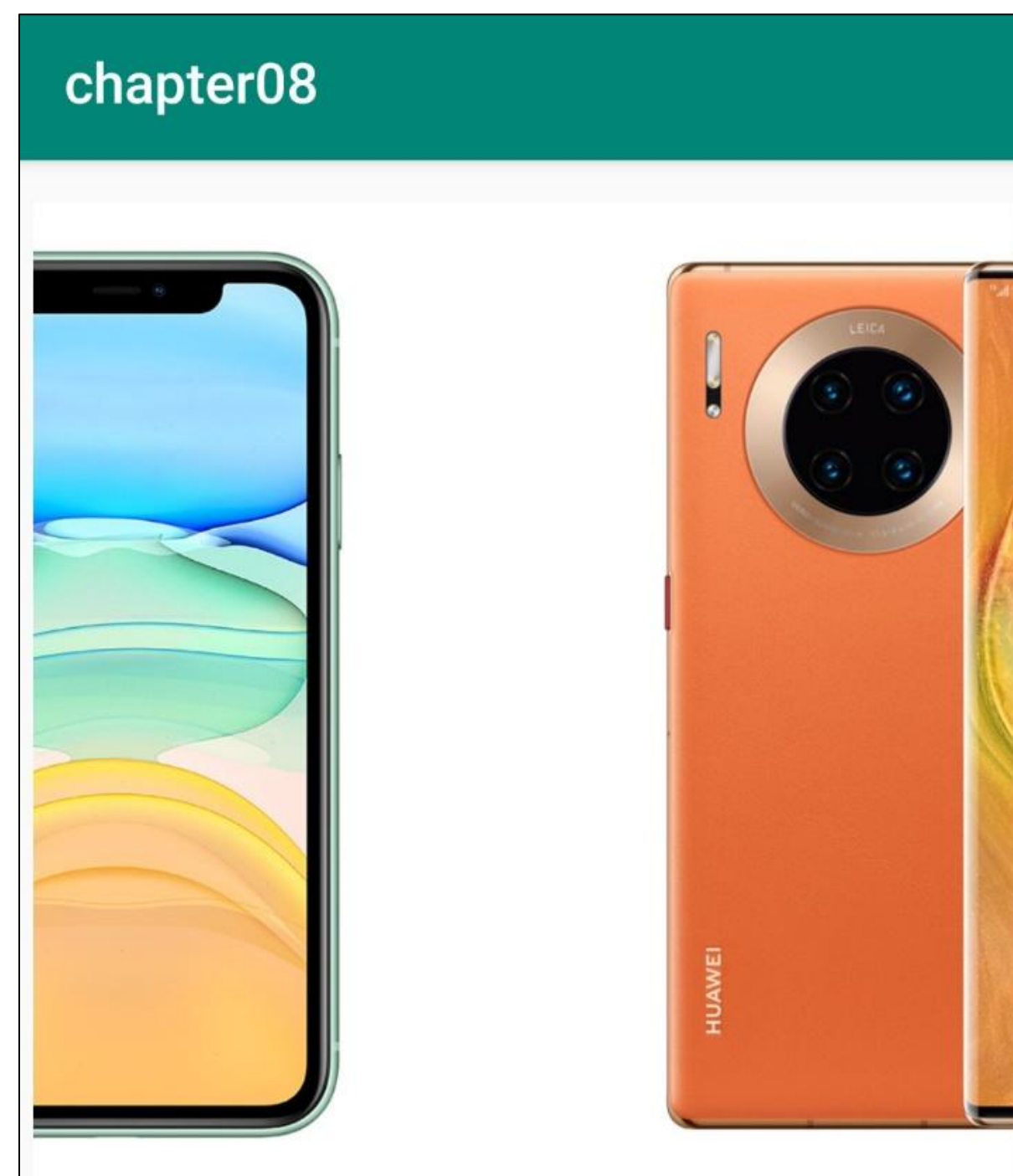


# 翻页视图 ViewPager

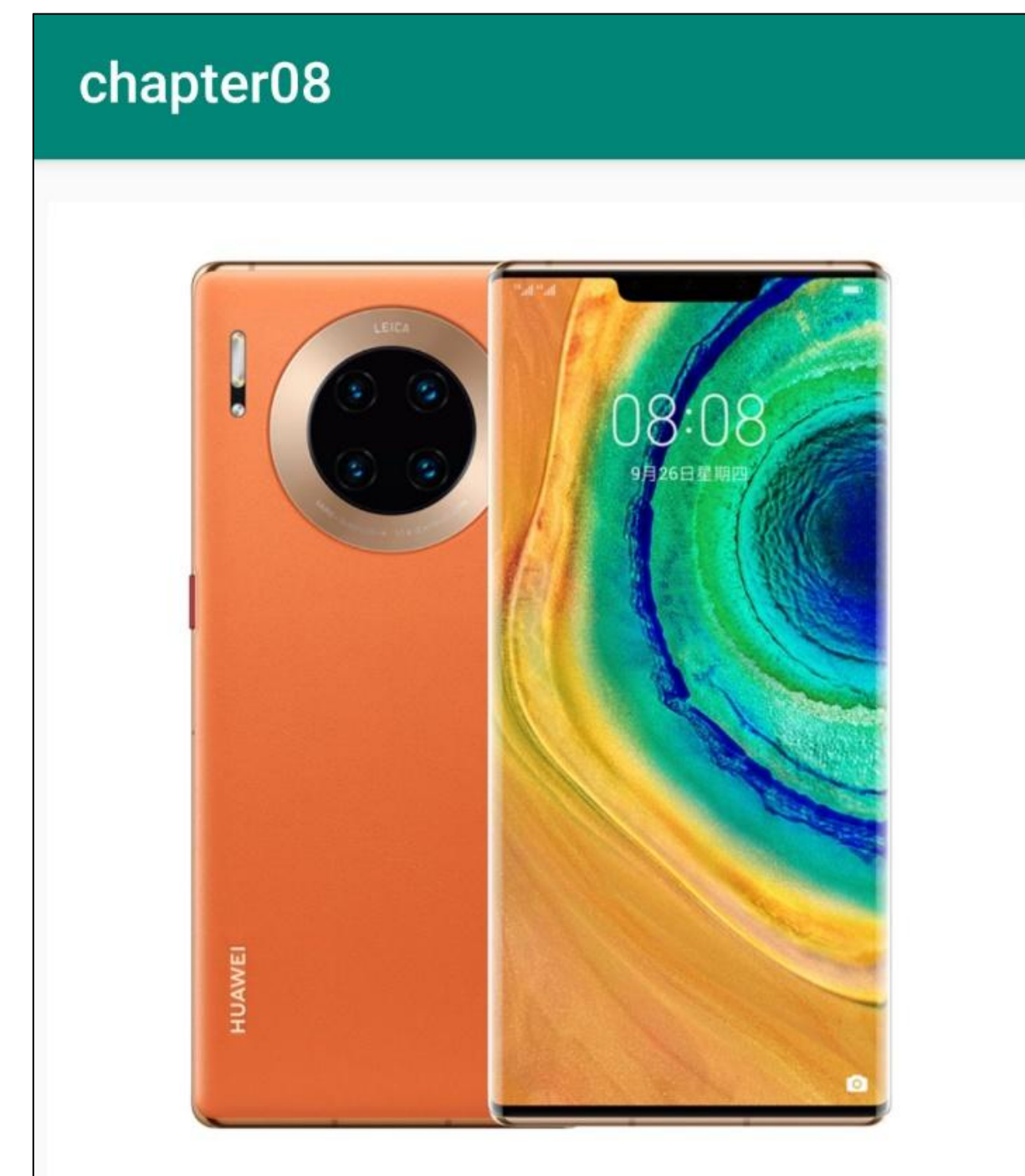
- 翻页视图允许页面在水平方向左右滑动。



初始的翻页视图



滑到一半的翻页视图



滑动结束的翻页视图

# 翻页标签栏 PagerTabStrip

- 翻页标签栏能够在翻页视图上方显示页面标题，点击页面标题即可切换到对应页面。



翻页标签栏的界面效果1



翻页标签栏的界面效果2



# 简单的启动引导页

- 当用户安装一个新应用时，首次启动大多出现欢迎页面，这个引导页要往右翻好几页，才会进入应用主页，这种启动引导页就是通过翻页视图实现的。





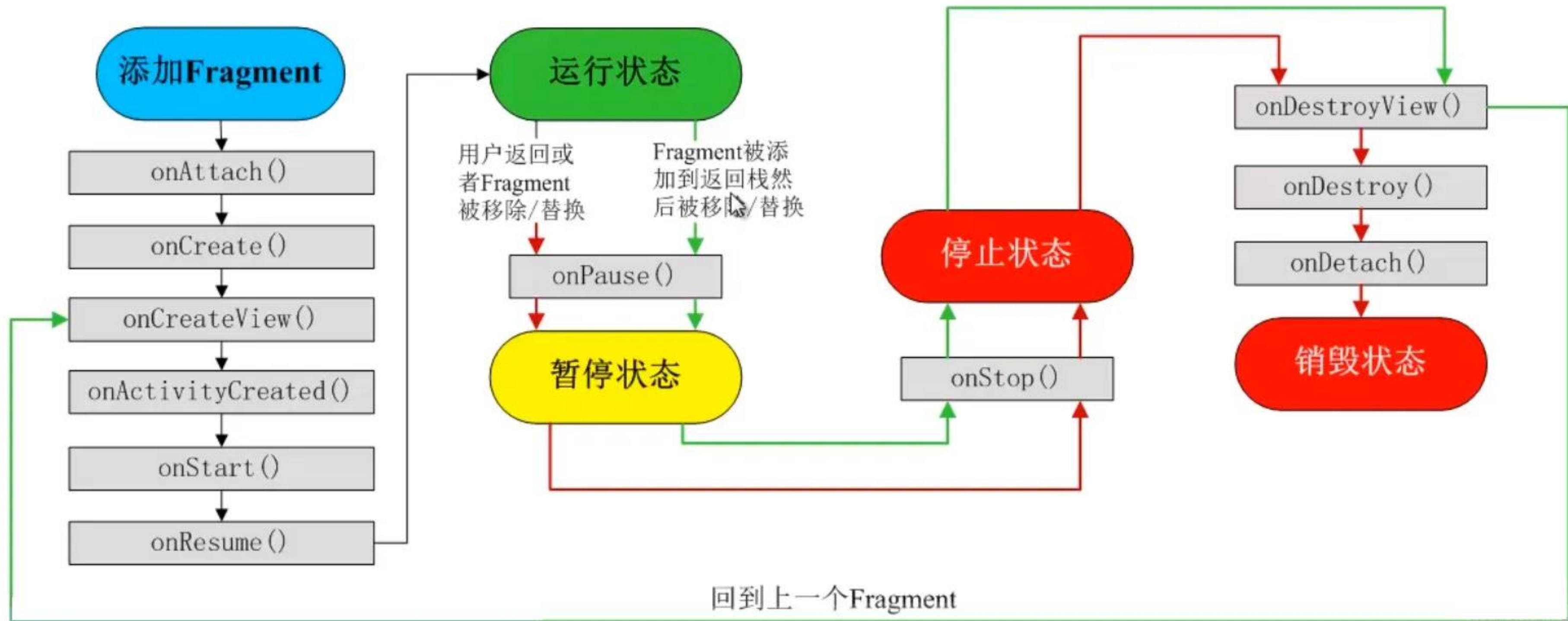
# 碎片 Fragment

- 传统的Activity并不能很好的处理大屏问题，所以急需一个碎片化的东西能够划区域的展示内容，且有属于自己的独立可操作空间，所以就出现了Fragment。



# Fragment 注册

- 静态注册在布局文件中直接指定Fragment，而动态注册直到在代码中才动态添加Fragment。





# 改进的启动引导页

- 与之前的启动引导页比较，改进后的启动引导页采用Fragment搭配ViewPager。
- 一旦发生页面切换，相邻页面就被加载，非相邻页面就被回收。只有用户正在浏览与将要浏览的Fragment才会加载，避免所有页面项一起加载造成资源浪费。
- 引入碎片之后有以下两个好处：
  - 加快启动速度
  - 降低代码耦合

# 实战项目：记账本

## ➤ 记录新账单与查看账单列表

<

请填写账单

账单列表

账单日期：

2020-07-29

▼

账单类型：

☐ 收入

☒ 支出

事项说明：

请填写说明内容

金额：

单位（元）

保

存

账单填写页面

<

账单列表

添加账单

请选择月份：

2020-06

▼

5月份

6月份

7月份

2020-06-02

吃饭

支出20元

2020-06-21

买水果

支出15元

2020-06-25

打车

支出30元

2020-06-26

手机话费充值

支出100元

合计

收入0元  
支出165元

余额-165元

账单列表页面