Shape

可以帮我绘制一些简单的背景，替代png图片资源，减少apk大小

<shape xmlns:android="http://schemas.android.com/apk/res/android" android:shape=["rectangle" | "oval" | "line" | "ring"] >

//画图形 主要是四种 矩形 椭圆 线 圆环

/圆角

<corners

android:radius="integer"

android:topLeftRadius="integer"

android:topRightRadius="integer"

android:bottomLeftRadius="integer" android:bottomRightRadius="integer" />

//渐变

<gradient android:angle="integer"

android:centerX="integer"

android:centerY="integer"

android:centerColor="integer"

android:endColor="color"

android:gradientRadius="integer"

android:startColor="color"

android:type=["linear" | "radial" | "sweep"]

android:useLevel=["true" | "false"] />

//内边距

<padding android:left="integer"

android:top="integer"

android:right="integer"

android:bottom="integer" />

//大小

<size android:width="integer"

android:height="integer" />

//填充色

<solid android:color="color" />

//边宽度

<stroke android:width="integer"

android:color="color"

android:dashWidth="integer"

android:dashGap="integer" />

</shape>

Android ArrayMap与HashMap

HashMap的内存使用率，Google 开发了ArrayMap在一定条件下可以取代HashMap.

ArrayMap主要依靠

mHashes 存放HashCode

mArray 存放key 与Value

**HashMap**的存储结构是一个数据加单向链表的形式。HashMap将每隔节点信息存储在Entry<K,V>结构中。Entry<K,V>中存储了节点对应的key、value、hash信息，同时存储了当前节点的下一个节点的引用。因此，Entry<K,V>是一个单向链表。每一个key对应的hashCode，在HashMap数组中都可以找到一个位置，而如果多个key对应了相同的hashCode，那么他们在数组中对应在相同的位置上，这是HashMap将把对应的信息放到Entry<K,V>中，并使用链表连接这些Entry<K,V>。

特点：链表支持，速度快；非基本数据对象创建，内存消耗严重

**ArrayMap**它使用了两个数组来存储数据——一个整型数组存储键的hashCode，另一个对象数组存储键/值对。这样既能避免为每个存入map中的键创建额外的对象，又能更积极的控制这些数据的长度的增加。因为增加长度只需要拷贝数组中的键，而不是重新构建一个哈希表。

特点：二分查找，废时间；内存创建少

总结：在几百的数量级的时候 推荐使用ArrayMap

其他：

• ArrayMap<K,V> 替代 HashMap<K,V>

• ArraySet<K,V> 替代 HashSet<K,V>

• SparseArray<V> 替代 HashMap<Integer,V>

• SparseBooleanArray 替代 HashMap<Integer,Boolean>

• SparseIntArray 替代 HashMap<Integer,Integer>

• SparseLongArray 替代 HashMap<Integer,Long>

• LongSparseArray<V> 替代 HashMap<Long,V>