

# Android XML布局

---

## 帧布局

---

1. 放置在帧布局中的控件，在不设置方位和边距的情况下，默认放在左上角。
2. 帧布局中的控件可以重叠放置。
3. 帧布局控件设置屏幕中方位使用`layout_gravity`属性（四方位）。属性值`center/center_vertical/center_horizontal`对应居中类型。
4. 可以链接居中位置和对齐位置，用`|`链接
5. 添加`margin`控制边距

## 线性布局

---

1. 方向属性：Orientation(**vertical/horizontal**)。[horizontal水平、vertical垂直]
2. 当线性布局为 垂直/水平 方向时，线性布局中的控件以 从上到下/从左到右 的方式堆放。堆放方式是无空隙的。
3. 设置`layout_gravity`时只能实现与布局方向垂直的对齐方向和对齐方向。（期中）
4. 可以通过垂直方向的相同方向的`margin`控制控件之间的距离。
5. 比重属性`weight`：可以通过`weight`属性分配布局占比比例。

## 网格布局

---

在Android 5.0 后推出

1. 网格布局有方向`orientation`属性[horizontal/vertical]，代表不同方向的网格布局，当一列/行放满后换行放置。
2. 网格布局中的组件不用设置`layout_height`和`layout_width`，但不代表高度和宽度属性不存在，网格布局中的组件预先已经被Grid Layout的宽高进行行数和列数的等比例划分。
3. `columnCount/rowCount`设置网格布局中组件的列数/行数
4. `layout_row`设置网格布局中组件的行数（从0开始），`layout_column`设置网格布局组件的列数，被设置行数和列数的组件，排在该组件之后的组件，紧接着该组件位置按序放置。排在该组件之前的位置不变。
5. `layout_rowweight` 和 `layout_columnweight` 可以按照网格布局中的比重属性值平均分配手机屏幕中的宽度/高度空间。
6. 网格布局使用`layout_columnspan/layout_rowspan`属性来设置跨列/跨行的网格，其属性值为要跨越的列数。

## 相对布局

---

1. 相对布局Relative Layout中没有`layout_gravity`属性，在相对布局中替代此属性的是一些方位属性：

属性名	内容	备注
android:layout_centerHorizontal	水平居中	android:layout_gravity="center_horizontal"
android:layout_centerVertical	垂直居中	android:layout_gravity="center_vertical"
android:layout_centerInParent	完全居中	android:layout_gravity="center"
android:layout_alignParentBottom	下边缘	android:layout_gravity="bottom"
android:layout_alignParentLeft	左边缘	android:layout_gravity="left"
android:layout_alignParentRight	右边缘	android:layout_gravity="right"
android:layout_alignParentTop	上边缘	android:layout_gravity="top"

2. 相对布局如果只有一个控件，在没有参照物的情况下使用android:layout\_centerHorizontal 水平居中。
3. 相对布局中控件默认放置在左上角，相对布局中的控件可以重叠放置。
4. 在相对布局中使用layout\_above/below/toLeftOf/toRightOf="参照物ID"来确定控件与参照控件的位置，并且通过align\*来与参照控件进行边缘对齐。
5. 如果参照与对齐都无法使控件放置在要求的位置，可以使用外边距margin\_\*来进行微调。

## 绝对布局

1. 以屏幕左上角为锚点，使用 android:layout\_x="[x]dp"、 android:layout\_y="[x]dp" 设置控件在屏幕中的精确位置