1. 布局管理器

1．线性布局管理器

1. 不换行属性：线性布局中组件不会自动换行，如果组件一个一个排列到尽头后，剩余的组件不会显示出来。
2. LinearLayout的xml属性

* android：gravity----设置布局管理器内组件的对齐方式，该属性值可设为 top（顶部对齐）、bottom（底部对齐）、left（左对齐）、right（右对齐）、center\_vertical（垂直方向居中）、fill\_vertical（必要的时候垂直方向拉伸，以完全充满其容器）、center\_horizontal（水平方向居中）、fill\_horizontal（水平方向填充）、center（垂直与水平方向都居中）、fill （必要的时候纵横向拉伸，以完全充满其容器）、clip\_vertical（垂直方向裁剪，按照容器的边来裁剪对象的顶或底部的内容，裁剪基于其纵向对齐设置：顶部对齐时，裁剪底部；底部对齐时裁剪顶部；除此之外裁剪顶部和底部）、clip\_horizontal（水平方向裁剪）、end (放在容器的结束位置，不改变其大小) 、start (将对象放在其容器的开始位置，不改变其大小)、top (将对象放在其容器的顶部，不改变其大小)。可以同时设置多个属性的组合，如left|center\_vertical。，竖线前后不能用空格。
* android:orientation----线性布局以列或行来显示内部子元素。“vertical”表示采用纵向的布局方式，“horizontal”表示采用横向的布局方式。
* android：baselineAligned----该属性设为 false ，将会阻止该布局管理器与它的子元素的基线对其。
* android：divider----设置垂直布局时两个按钮直接的分隔条。
* android：measureWithLargestChild----该属性设为 true 时，所有带权重的子元素都会具有最大子元素的最小尺寸。
* android：layout\_gravity--指定该子元素在 LinearLayout 中的对其方式。当 android:orientation="vertical" 时， 只有水平方向的设置才起作用，垂直方向的设置不起作用。即：left，right，center\_horizontal 是生效的。当 android:orientation="horizontal" 时， 只有垂直方向的设置才起作用，水平方向的设置不起作用。即：top，bottom，center\_vertical 是生效的。
* android：layout\_weight---- 指定该子元素在 LinearLayout 中所占的权重。
* android:padding="10dp"----本元素所有子元素的与父元素边缘的距离，设置在父元素上 。
* android:layout\_marginLeft="10dp"----子元素与父元素边缘的距离，设置在子元素上。
* android:layout\_height:----表示当前线性布局的高度。"match\_parent" 表示高度占满整个屏幕、"wrap\_content" 表示高度根据其包含的控件自适应调整)、"30dp"自定义设置高度，通常单位为dp。
* android:layout\_width----表示当前线性布局的宽。"match\_parent" 表示宽度占满整个屏幕、 wrap\_content" 表示宽度根据其包含的控件自适应调整、"30dp" 自定义设置宽度，通常单位为dp。
* android:margin----表示外边距，通常表示本控件与父控件四面之间的距离。
* android:background----表示当前线性布局的背景颜色或背景图（直接加图片即可）。
* android:weightSum----权重的总比例。

注：[权重比例分配原理](权重分配原理.docx)

2.相对布局管理器

1. RelativeLayput的属性

* android:gravity="left"----空间布局位置 。
* android:ignoreGravity----指定那个子元素不受gravity影响。

// 相对于给定ID控件

* android:layout\_above----将该控件的底部置于给定ID的控件之上。
* android:layout\_below ----将该控件的底部置于给定ID的控件之下。
* android:layout\_toLeftOf----将该控件的右边缘与给定ID的控件左边缘对齐。
* android:layout\_toRightOf----将该控件的左边缘与给定ID的控件右边缘对齐。
* android:layout\_alignBaseline ----该控件内容与给定id控件的内容在一条线上（[效果](androidlayout_alignBaseline效果.docx)）
* android:layout\_alignTop ----将该控件的顶部边缘与给定ID的顶部边缘对齐。
* android:layout\_alignBottom----将该控件的底部边缘与给定ID的底部边缘对齐。
* android:layout\_alignLeft ----将该控件的左边缘与给定ID的左边缘对齐。
* android:layout\_alignRight ----将该控件的右边缘与给定ID的右边缘对齐。

// 相对于父组件

* android:layout\_alignParentTop----如果为true,将该控件的顶部与其父控件的顶部对齐。
* android:layout\_alignParentBottom----如果为true,将该控件的底部与其父控件的底部对齐。
* android:layout\_alignParentLeft----如果为true,将该控件的左部与其父控件的左部对齐。
* android:layout\_alignParentRight ----如果为true,将该控件的右部与其父控件的右部对齐。

// 居中

* android:layout\_centerHorizontal ----如果为true,将该控件的置于水平居中。
* android:layout\_centerVertical----如果为true,将该控件的置于垂直居中。
* android:layout\_centerInParent----如果为true,将该控件的置于父控件的中央。

// 指定移动像素

* android:layout\_marginTop----上偏移的值。
* android:layout\_marginBottom----下偏移的值。
* android:layout\_marginLeft ----左偏移的值。
* android:layout\_marginRight ----右偏移的值。

1. 常用的UI组件
2. RadioGroup和RadioButton
3. RadioGroup的常用方法和属性

* android:orientation 方向
* RadioGroup.getCheckedRadioButtonId ()----获取选中按钮的id。使用.getCheckedRadioButtonId()来获得它的值的时候，若没有给它设置默认的值，则会报空指针错误。解决方法是在布局文件用android：checked=”true”指定选中对象。
* RadioGroup.clearCheck ()-----清除选中状态
* RadioGroup.check (int id)-----通过参入选项id来设置该选项为选中状态如果传递-1作为指定的选择标识符来清除单选按钮组的勾选状态，相当于调用clearCheck()操作
* setOnCheckedChangeListener (RadioGroup.OnCheckedChangeListener listener)----一个当该单选按钮组中的单选按钮勾选状态发生改变时所要调用的回调函数
* addView (View child, int index, ViewGroup.LayoutParams params)-----使用指定的布局参数添加一个子视图。参数 child 所要添加的子视图 index 将要添加子视图的位置 params 所要添加的子视图的布局参数

2）RadioButton的常用属性和方法

* android:text 文字
* android:checked 是否选中
* boolean isChecked() 返回是否被选中
* getText() 返回文字
* setChecked(boolean checked) 设置是否选中