

GridLayout: GridLayout使用简介(转)

1 GridLayout简介

GridLayout是Android4.0新提供的网格矩阵形式的布局控件。

GridLayout的继承关系如下：

```
java.lang.Object
--> android.view.View
--> android.view.ViewGroup
--> android.widget.GridLayout
```

GridLayout包含的属性如下：

android:alignmentMode

属性说明：当设置alignMargins，使视图的外边界之间进行校准。可以取以下值：

alignBounds -- 对齐子视图边界。

alignMargins -- 对齐子视图边距。

android:columnCount

属性说明：GridLayout的最大列数

android:rowCount

属性说明：GridLayout的最大行数

android:columnOrderPreserved

属性说明：当设置为true，使列边界显示的顺序和列索引的顺序相同。默认是true。

android:orientation

属性说明：GridLayout中子元素的布局方向。有以下取值：

horizontal -- 水平布局。

vertical -- 垂直布局。

android:rowOrderPreserved

属性说明：当设置为true，使行边界显示的顺序和行索引的顺序相同。默认是true。

android:useDefaultMargins

属性说明：当设置ture，当没有指定视图的布局参数时，告诉GridLayout使用默认的边距。默认值是false。

这些是GridLayout布局本身的属性。

2 GridLayout子元素属性

上面描述的 GridLayout 的属性，是 GridLayout 布局本身的属性；下面 GridLayout 布局中的元素所支持的属性。GridLayout 布局中的元素的属性，定义在 **GridLayout.LayoutParams** 中。取值如下：

2.1 android:layout_column

属性说明：显示该空间的列。例如，android:layout_column="0"，表示在第1列显示该控件；android:layout_column="1"，表示在第2列显示该控件。

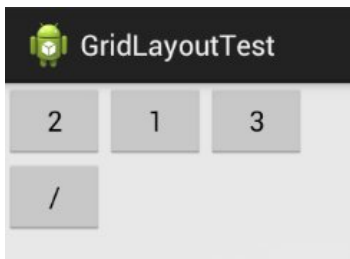
layout文件示例，



```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:rowCount="2"
    android:columnCount="3" >
    <Button
        android:id="@+id/one"
        android:layout_column="1"
        android:text="1"/>
    <Button
        android:id="@+id/two"
        android:layout_column="0"
        android:text="2"/>
    <Button
        android:id="@+id/three"
        android:text="3"/>
    <Button
        android:id="@+id/devide"
        android:text="/">
</GridLayout>
```



对应的显示效果图:



layout文件说明:

android:orientation="horizontal" -- GridLayout中控件的布局方向是水平布局。

android:rowCount="2" -- GridLayout最大的行数为2行。

android:columnCount="3" -- GridLayout最大的列数为3列。

android:layout_column="1" -- 定义控件one的位于第2列。

android:layout_column="0" -- 定义该控two件的位于第1列。

2.2 android:layout_columnSpan

属性说明: 该控件所占的列数。例如, android:layout_columnSpan="2", 表示该控件占2列。

layout文件示例:



```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:rowCount="2"
    android:columnCount="3" >
    <Button
        android:id="@+id/one"
        android:layout_column="0"
        android:layout_columnSpan="2"
        android:text="1"/>
    <Button
        android:id="@+id/two"
        android:layout_column="2"
        android:text="2"/>
</GridLayout>
```

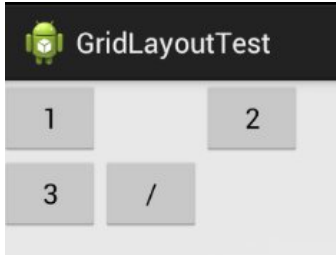
```
        android:text="2"/>
    <Button
        android:id="@+id/three"
        android:text="3"/>

    <Button
        android:id="@+id/devide"
        android:text="/">

</GridLayout>
```



对应的显示效果图：



layout文件说明：

数字"1"实际上占据的空间大小是2列，但是第2列显示为空白。若要第2列不显示空白，需要设置 android:layout_gravity属性，参考下例。

2.3 android:layout_row

属性说明： 该控件所在行。例如， android:layout_row="0"，表示在第1行显示该控件； android:layout_row="1"，表示在第2行显示该控件。它和 android:layout_column类似。

2.4 android:layout_rowSpan

属性说明： 该控件所占的行数。例如， android:layout_rowSpan="2"，表示该控件占2行。它和 android:layout_columnSpan类似。

2.5 android:layout_gravity

属性说明：

该控件的布局方式。可以取以下值：

- top -- 控件置于容器顶部，不改变控件的大小。
- bottom -- 控件置于容器底部，不改变控件的大小。
- left -- 控件置于容器左边，不改变控件的大小。
- right -- 控件置于容器右边，不改变控件的大小。
- center_vertical -- 控件置于容器竖直方向中间，不改变控件的大小。
- fill_vertical -- 如果需要，则往竖直方向延伸该控件。
- center_horizontal -- 控件置于容器水平方向中间，不改变控件的大小。
- fill_horizontal -- 如果需要，则往水平方向延伸该控件。
- center -- 控件置于容器中间，不改变控件的大小。
- fill -- 如果需要，则往水平、竖直方向延伸该控件。
- clip_vertical -- 垂直剪切，剪切的方向基于该控件的top/bottom布局属性。若该控件的gravity是竖直的：若它的gravity是top的话，则剪切该控件的底部；若该控件的gravity是bottom的，则剪切该控件的顶部。
- clip_horizontal -- 水平剪切，剪切的方向基于该控件的left/right布局属性。若该控件的gravity是水平的：若它的gravity是left的话，则剪切该控件的右边；若该控件的gravity是 right的，则剪切该控件的左边。
- start -- 控件置于容器的起始处，不改变控件的大小。
- end -- 控件置于容器的结束处，不改变控件的大小。

对应函数： setGravity(int)

layout文件示例：

```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:rowCount="2"
    android:columnCount="3" >
    <Button
        android:id="@+id/one"
        android:layout_column="0"
        android:layout_columnSpan="2"
        android:layout_gravity="fill"
        android:text="1"/>
    <Button
        android:id="@+id/two"
        android:text="2"/>
    <Button
        android:id="@+id/three"
        android:text="3"/>
    <Button
        android:id="@+id/devide"
        android:text="/">
</GridLayout>
```

对应的显示效果图：



3 应用示例

定义一个简单的计算器界面，包含“0-9、.、+、-、*、/、=、”。用GridLayout实现。

layout文件

```
<?xml version="1.0" encoding="utf-8"?>
<!-- GridLayout: 5行 4列 水平布局 -->
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:rowCount="5"
    android:columnCount="4" >
    <Button
        android:id="@+id/one"
        android:text="1"/>
    <Button
        android:id="@+id/two"
```

```
        android:text="2"/>
    <Button
        android:id="@+id/three"
        android:text="3"/>
    <Button
        android:id="@+id/devide"
        android:text="/" />
    <Button
        android:id="@+id/four"
        android:text="4"/>
    <Button
        android:id="@+id/five"
        android:text="5"/>
    <Button
        android:id="@+id/six"
        android:text="6"/>
    <Button
        android:id="@+id/multiply"
        android:text="x"/>
    <Button
        android:id="@+id/seven"
        android:text="7"/>
    <Button
        android:id="@+id/eight"
        android:text="8"/>
    <Button
        android:id="@+id/nine"
        android:text="9"/>
    <Button
        android:id="@+id/minus"
        android:text="-"/>
    <Button
        android:id="@+id/zero"
        android:layout_columnSpan="2"
        android:layout_gravity="fill"
        android:text="0"/>
    <Button
        android:id="@+id/point"
        android:text="."/>
    <Button
        android:id="@+id/plus"
        android:layout_rowSpan="2"
        android:layout_gravity="fill"
        android:text="+"/>
    <Button
        android:id="@+id/equal"
        android:layout_columnSpan="3"
        android:layout_gravity="fill"
        android:text="="/>
</GridLayout>
```



点击下载: [源代码](#)

效果图:

