**Обзор TableLayout**

Диспетчер компоновки TableLayout является расширением компоновки LinearLayout. Этот диспетчер выстраивает вложенные элементы в строки и столбцы, как и в любой таблице.  
Для использования этой компоновки, нужно в контейнере TableLayout создать элементы TableRow, которые будут являться строками и выстраиваться вертикально. Внутри TableRow можно размещать другие элементы и они будут выстраиваться горизонтально.  
  
Начнём с простого примера:

**Пример 1**

<?xml version="1.0" encoding="utf-8"?>

<**TableLayout** **xmlns:android**=**"http://schemas.android.com/apk/res/android"**

**android:background**=**"#eeeeee"**

**android:stretchColumns**=**"\*"**

**android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"fill\_parent"**>

<**TableRow** **android:background**=**"#cccccc"**

**android:padding**=**"5dp"**>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"Дата"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceLarge"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"Кол-во"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceLarge"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"Сумма"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceLarge"**

/>

</**TableRow**>

<**TableRow** **android:padding**=**"5dp"**>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"2012-11-22"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"397"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"19039.21"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

</**TableRow**>

<**TableRow** **android:padding**=**"5dp"**>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"2012-11-23"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"348"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"16759.28"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

</**TableRow**>

<**TableRow** **android:padding**=**"5dp"**>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"2012-11-24"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"305"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"14527.19"**

**android:gravity**=**"center\_horizontal"**

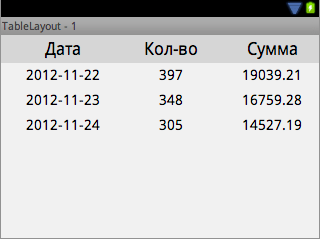
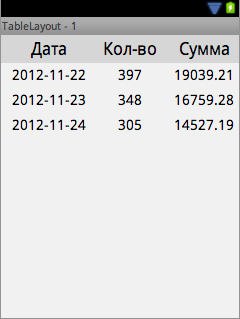
**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

/>

</**TableRow**>

</**TableLayout**>

Из кода мы получили такую вот простую табличку



Разберём как это работает. Вначале создаём контейнер разметки TableLayout и вкладываем в него 3 элемента TableRow.

<**TableLayout** **xmlns:android**=**"http://schemas.android.com/apk/res/android"**

**android:background**=**"#eeeeee"**

**android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"fill\_parent"**>

<**TableRow**></**TableRow**>

<**TableRow**></**TableRow**>

<**TableRow**></**TableRow**>

<**TableRow**></**TableRow**>

</**TableLayout**>

  У нас получилось 4 строки в таблице. Далее создаём столбцы таблицы и для этого в каждый элемент TableRowвкладываем по 3 элемента TextView. В первом напишем названия столбцов, а во вторых названия. Для того, чтобы визуально отделить заголовки таблицы, мы зададим фоновый цвет для первого TableRow.

<**TableRow** **android:background**=**"#cccccc"**></**TableRow**>

Чтобы содержимое элементов TableRow растянулось на всю ширину макета, нужно указать у TableLayout значение атрибуте android:stretchColumns равным звёздочке.

android:stretchColumns=**"\*"**

Вообще, через запятую можно указать порядковые номера столбцов, которые нужно растянуть. Номера столбцов указываем начиная с нуля. Остальные займут минимум места.  
Давайте рассмотрим подробнее эту функциональность на примере.

**Атрибуты stretchColumn**

**Пример 2**

<?xml version="1.0" encoding="utf-8"?>

<**LinearLayout** **xmlns:android**=**"http://schemas.android.com/apk/res/android"**

**android:orientation**=**"vertical"**

**android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"fill\_parent"**>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:textStyle**=**"bold"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceLarge"**

**android:text**=**"stretchColumns=0,3,5"**/>

<**TableLayout**

**android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:stretchColumns**=**"0,3,5"**

>

<**TableRow**>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"0"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"1"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"2"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"3"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"4"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"5"**

/>

</**TableRow**>

</**TableLayout**>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:textStyle**=**"bold"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceLarge"**

**android:text**=**"stretchColumns=\*"**/>

<**TableLayout**

**android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:stretchColumns**=**"\*"**

>

<**TableRow**>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"0"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"1"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"2"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"3"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"4"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"5"**

/>

</**TableRow**>

<**TableRow**>

<**TextView** **android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:textStyle**=**"bold"**

**android:gravity**=**"center\_horizontal"**

**android:textAppearance**=**"?android:attr/textAppearanceLarge"**

**android:layout\_span**=**"6"**

**android:text**=**"layout\_span"**/>

</**TableRow**>

<**TableRow**>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"layout\_span=3"**

**android:layout\_span**=**"3"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"1"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"layout\_span=2"**

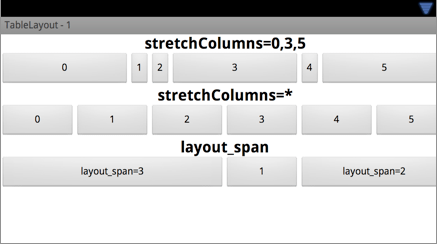
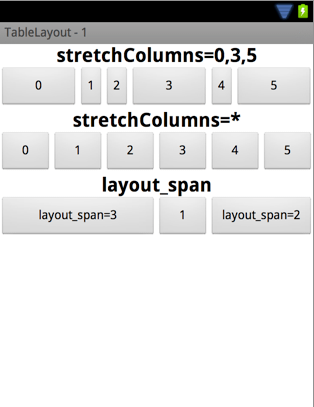
**android:layout\_span**=**"2"**

/>

</**TableRow**>

</**TableLayout**>

</**LinearLayout**>



В этом примере, в первой таблице мы задали атрибут strechColumn равным "0,3,5"

<**TableLayout**

**android:layout\_width**=**"fill\_parent"**

**android:layout\_height**=**"wrap\_content"**

**android:stretchColumns**=**"0,3,5"**

>

И как видно из картинки, столбцы с индексами 0, 3 и 5 растянулись равномерно, а остальные заняли минимум места. Как я уже писал выше, индекс считается с нуля.   
Для сравнения, у следующей таблицы мы указали, чтобы растянулись все столбцы, значением звёздочкой android:stretchColumns="\*".  
  
На JAVA это задаётся методом setStretchAllColumns(true)(для растягивания всех столбцов).   
Чтобы задать определённые столбцы, нужно воспользоваться методом setColumnStretchable(int index, boolean isStretchable). Для каждого индекса нужно вызывать этот метод отдельно, например:

*// эквивалент XML атрибута android:stretchColumns="0,3,5"*

tableLayout.setColumnStretchable(0, **true**);

tableLayout.setColumnStretchable(3, **true**);

tableLayout.setColumnStretchable(5, **true**);

**Атрибут layout\_span**

Если Вы знакомы с построением таблиц, на HTML, то наверняка знаете атрибут colspan у тега , который отвечает за объединение нескольких столбцов в один. Так вот, мы тоже можем это сделать в нашей компоновке при помощи атрибута android:layout\_span у вложенного элемента в TableRow и указав значение в виде целого числа, означающего количество объединённых столбцов.

<**TableRow**>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"layout\_span=3"**

**android:layout\_span**=**"3"**

**android:id**=**"@+id/btn1"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"1"**

**android:id**=**"@+id/btn2"**

/>

<**Button** **android:layout\_width**=**"wrap\_content"**

**android:layout\_height**=**"wrap\_content"**

**android:text**=**"layout\_span=2"**

**android:layout\_span**=**"2"**

**android:id**=**"@+id/btn3"**

/>

</**TableRow**>

Как видите, первая кнопка растянулась на 3 столбца, а третья на 2.  
  
Программно это можно реализовать при помощи вложенного в TableLayout класса LayoutParams.  
Пример:

Button button1 = (Button) findViewById(R.id.btn1);

TableRow.LayoutParams params = (TableRow.LayoutParams) button1.getLayoutParams();

params.span = 3;

button1.setLayoutParams(params);

Button button3 = (Button) findViewById(R.id.btn3);

TableRow.LayoutParams params3 = (TableRow.LayoutParams) button3.getLayoutParams();

params3.span = 2;

button3.setLayoutParams(params3);

**Рамка в таблице**

К сожалению, наш контейнер не может создать рамку для таблицы. Разработчики Android почему то не удосужились добавить эту функциональность. Ну что же,  попробуем её создать сами.   
Схема такая – задаём у корневого элемента TableLayout атрибут background в виде цвета, например

<**TableLayout** **xmlns:android**=**"http://schemas.android.com/apk/res/android"**

**android:layout\_height**=**"wrap\_content"**

**android:layout\_width**=**"fill\_parent"**

**android:stretchColumns**=**"\*"**

**android:background**=**"#000"**>

Мы установили фоновый цвет в чёрный и это будет цвет нашей рамки.  
Далее просто задаём у элементов, вложенных в TableRow другой фоновый цвет. Он будет перекрывать фоновый цвет элемента TableLayout и тем самым у нас получается подобие рамки между элементами.  
Рассмотрим полный код:

**Пример 3**

<**TableLayout** **xmlns:android**=**"http://schemas.android.com/apk/res/android"**

**android:layout\_height**=**"wrap\_content"**

**android:layout\_width**=**"fill\_parent"**

**android:stretchColumns**=**"\*"**

**android:background**=**"#000"**>

<**TableRow**>

<**TextView** **android:textColor**=**"#fff"**

**android:textStyle**=**"bold"**

**android:gravity**=**"center\_horizontal"**

**android:background**=**"#7fc342"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

**android:layout\_span**=**"3"**

**android:text**=**"Статистика"**/>

</**TableRow**>

<**TableRow**>

<**TextView** **android:background**=**"#fff"**

**android:layout\_margin**=**"1dip"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

**android:textColor**=**"#36a300"**

**android:gravity**=**"center\_horizontal"**

**android:text**=**"1 столбец"**/>

<**TextView** **android:background**=**"#fff"**

**android:layout\_margin**=**"1dip"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

**android:textColor**=**"#36a300"**

**android:gravity**=**"center\_horizontal"**

**android:text**=**"2 столбец"**/>

<**TextView** **android:background**=**"#fff"**

**android:layout\_margin**=**"1dip"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

**android:textColor**=**"#36a300"**

**android:gravity**=**"center\_horizontal"**

**android:text**=**"3 столбец"**/>

</**TableRow**>

<**TableRow**>

<**TextView** **android:background**=**"#ccc"**

**android:layout\_margin**=**"1dip"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

**android:textColor**=**"#36a300"**

**android:gravity**=**"center\_horizontal"**

**android:text**=**"1 столбец"**/>

<**TextView** **android:background**=**"#ccc"**

**android:layout\_margin**=**"1dip"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

**android:textColor**=**"#36a300"**

**android:gravity**=**"center\_horizontal"**

**android:text**=**"2 столбец"**/>

<**TextView** **android:background**=**"#ccc"**

**android:layout\_margin**=**"1dip"**

**android:textAppearance**=**"?android:attr/textAppearanceMedium"**

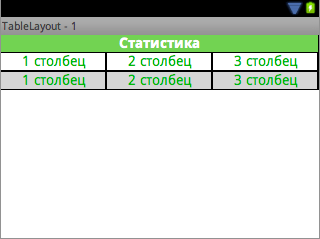
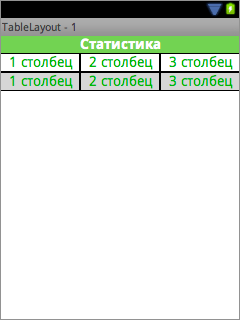
**android:textColor**=**"#36a300"**

**android:gravity**=**"center\_horizontal"**

**android:text**=**"3 столбец"**/>

</**TableRow**>

</**TableLayout**>



Как видно у 2 и 3 строки мы задали у текста фоновый цвет в белый и серый, он перекрыл чёрный фоновый у таблицы и от неё остались одни ~~лишь рожки, да ножки~~ чёрточки.

**Программное создание таблицы**

**Пример 4**

TableLayout tableLayout = **new** TableLayout(**this**);

tableLayout.setLayoutParams(**new** TableLayout.LayoutParams(

ViewGroup.LayoutParams.FILL\_PARENT,

ViewGroup.LayoutParams.WRAP\_CONTENT

));

tableLayout.setStretchAllColumns(**true**);

TextView textView1 = **new** TextView(**this**);

textView1.setText(**"Column 1"**);

TextView textView2 = **new** TextView(**this**);

textView2.setText(**"Column 2"**);

TextView textView3 = **new** TextView(**this**);

textView3.setText(**"Column 3"**);

TextView textView4 = **new** TextView(**this**);

textView4.setText(**"Column 4"**);

TextView textView5 = **new** TextView(**this**);

textView5.setText(**"Column 5"**);

TextView textView6 = **new** TextView(**this**);

textView6.setText(**"Column 6"**);

TextView textView7 = **new** TextView(**this**);

textView7.setText(**"Column 7"**);

TextView textView8 = **new** TextView(**this**);

textView8.setText(**"Column 8"**);

TextView textView9 = **new** TextView(**this**);

textView9.setText(**"Column 9"**);

TableRow tableRow1 = **new** TableRow(**this**);

TableRow tableRow2 = **new** TableRow(**this**);

TableRow tableRow3 = **new** TableRow(**this**);

tableRow1.addView(textView1);

tableRow1.addView(textView2);

tableRow1.addView(textView3);

tableRow2.setBackgroundColor(0xffcccccc);

tableRow2.addView(textView4);

tableRow2.addView(textView5);

tableRow2.addView(textView6);

tableRow3.addView(textView7);

tableRow3.addView(textView8);

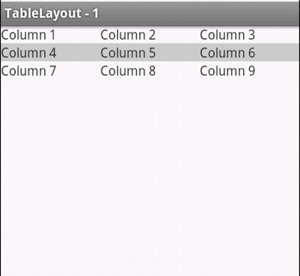
tableRow3.addView(textView9);

tableLayout.addView(tableRow1);

tableLayout.addView(tableRow2);

tableLayout.addView(tableRow3);

setContentView(tableLayout);



Исходный код примеров на гитхабе - <https://github.com/devpad/Android-TableLayout-1>