1. Learn how to create new layout and how to set it into setContentview. When you turn android it shows up bit wrong to solve this problem we have to create new folder with orientation and choose from orientation landscape and there create new content with views.Then it will shows up well in landscape view and linear view

Расположение View-элементов на экране зависит от **ViewGroup**(Layout), в которой они находятся. В этом уроке мы рассмотрим основные виды **Layout**.

**1 LinearLayout**– отображает View-элементы в виде одной строки (если он Horizontal) или одного столбца (если он Vertical). Я использовал это на прошлом уроке, когда демонстрировал использование layout-файлов при смене ориентации.

1. **TableLayout**– отображает элементы в виде таблицы, по строкам и столбцам.
2. **RelativeLayout**– для каждого элемента настраивается его положение относительно других элементов.
3. **AbsoluteLayout**– для каждого элемента указывается явная позиция на экране в системе координат (x,y)

- слово **android** в названии каждого атрибута – это [namespace](http://www.w3schools.com/xml/xml_namespaces.asp" \t "_blank), я его буду опускать при объяснениях.  
**- id** – это ID элемента,  
**- layout\_width** (ширина элемента) и **layout\_height** (высота элемента) могут задаваться в абсолютных значениях, а могут быть следующими: **fill\_parent** (максимально возможная ширина или высота в пределах родителя) и **wrap\_content** (ширина или высота определяется по содержимому элемента). В [хелпе](http://developer.android.com/reference/android/view/ViewGroup.LayoutParams.html" \t "_blank) указывается, что есть еще **match\_parent**. Это тоже самое, что и fill\_parent. По каким-то причинам, разработчики системы решили, что название match\_parent удобнее, и от fill\_parent постепенно будут отказываться. А пока его оставили для совместимости. Так что запомните, что **match\_parent = fill\_parent** и в дальнейшем будем стараться использовать **match\_parent**. Позже мы еще остановимся на этом и разберем подробнее.

Code in MainActivity

EditText text = findViewById(R.id.***editText2***);  
TextView show = findViewById(R.id.***textView***);  
Button btn = findViewById(R.id.***button***);  
Button bt = findViewById(R.id.***button5***);  
btn.setOnClickListener(b ->{  
 show.setText(text.getText());  
});  
bt.setOnClickListener(view -> {  
 text.setText(**""**);  
 show.setText(**""**);  
});

Test2.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
  
 <**LinearLayout  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:inputType="textPersonName"  
 android:text="@string/textName"  
 android:ems="10"  
 tools:layout\_editor\_absoluteY="16dp" tools:layout\_editor\_absoluteX="65dp" android:id="@+id/editText2"**/>  
 <**Button  
 android:text="@string/save"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" tools:layout\_editor\_absoluteY="105dp"  
 tools:layout\_editor\_absoluteX="38dp" android:id="@+id/button"**/>  
 <**Button  
 android:text="@string/cancel"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" tools:layout\_editor\_absoluteY="105dp"  
 tools:layout\_editor\_absoluteX="209dp" android:id="@+id/button5"**/>  
 <**TextView  
 android:text="TextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="47dp" android:id="@+id/textView"**/>  
 </**LinearLayout**>  
</**android.support.constraint.ConstraintLayout**>

**OnlickListener**

public class MainActivity extends Activity implements OnClickListener {

   TextView tvOut;

   Button btnOk;

   Button btnCancel;

   /\*\* Called when the activity is first created. \*/

   @Override

   public void onCreate(Bundle savedInstanceState) {

     super.onCreate(savedInstanceState);

     setContentView(R.layout.main);

     // найдем View-элементы

     tvOut = (TextView) findViewById(R.id.tvOut);

     btnOk = (Button) findViewById(R.id.btnOk);

     btnCancel = (Button) findViewById(R.id.btnCancel);

     // присваиваем обработчик кнопкам

     btnOk.setOnClickListener(this);

     btnCancel.setOnClickListener(this);

   }

   @Override

   public void onClick(View v) {

     // по id определеяем кнопку, вызвавшую этот обработчик

     switch (v.getId()) {

     case R.id.btnOk:

       // кнопка ОК

       tvOut.setText("Нажата кнопка ОК");

       break;

     case R.id.btnCancel:

       // кнопка Cancel

       tvOut.setText("Нажата кнопка Cancel");

       break;

     }

   }

 }