#### Package

Package 包。只是在我们的app中这个Package是唯一的，就像你身份证号码一样。在我们做app自动化时，我们就需要知道他的Package，我们知道了Package那么也就知道我们需要对哪个app做自动化。 注意和.apk文件包名不同。

通过UIAuto演示不同app的包名--见视频

#### Activity

Android中，activity是所有程序的根本，所有程序的流程都运行在activity之中，activity可以算是开发者遇到的最频繁，也是android当中最基本的模块之一。在android的程序中，activity一般代表手机屏幕的一屏。如果把手机比作一个浏览器，那么activity就相当于一个网页。在activity当中可以添加一些Button、Checkbox等控件，可以看到activity概念和网页的概念相当类似。

一般一个android应用是由多个activity组成的，这多个activity之间可以进行相互跳转。例如，按下一个Button按钮后，可能会跳转到其他的activity，与网页跳转稍微有点不一样的是，activity之间的跳转有可能返回值。

Tips：activity的生命周期：即“产生、运行、销毁”，但是这其中会调用许多方法onCreate（创建） 、onStart（激活） 、onResume（恢复） 、onPause（暂停） 、onStop（停止） 、onDestroy（销毁） 、onRestart（重启）。

#### Activity获取

研发提供

##### aapt

aapt即Android Asset Packaging Tool，在SDK的build-tools目录下。该工具可以查看，创建， 更新ZIP格式的文档附件(zip, jar, apk)。也可将资源文件编译成二进制文件。获取命令如下：

aapt **dump** badging xxxx.apk

aapt **dump** badging xxxx.apk | **find** "launchable-activity"

可以把appt配置到环境变量（系统变量中的Path），这样运行便捷一些，appt路径：\Andriod\_SDK\build-tools{version}

#### Activity页面布局元素

##### FrameLayout

FrameLayout是最简单的布局了。所有放在布局里的控件，都按照层次堆叠在屏幕的左上角。后加进来的控件覆盖前面的控件。

##### LinearLayout

LinearLayout按照垂直或者水平的顺序依次排列子元素，每一个子元素都位于前一个元素之后。如果是垂直排列，那么将是一个N行单列的结构，每一行只会有一个元素，而不论这个元素的宽度为多少；如果是水平排列，那么将是一个单行N列的结构。如果搭建两行两列的结构，通常的方式是先垂直排列两个元素，每一个元素里再包含一个LinearLayout进行水平排列。

##### RelativeLayout

RelativeLayout相对布局允许子元素指定它们相对于其父元素或兄弟元素的位置，这是实际布局中最常用的布局方式之一。

##### AbsoluteLayout

AbsoluteLayout是绝对位置布局。在此布局中的子元素的android:layout\_x和android:layout\_y属性将生效，用于描述该子元素的坐标位置。屏幕左上角为坐标原点（0,0），第一个0代表横坐标，向右移动此值增大，第二个0代表纵坐标，向下移动，此值增大。在此布局中的子元素可以相互重叠。在实际开发中，通常不采用此布局格式，

##### TableLayout

TableLayout 为表格布局，适用于N行N列的布局格式。一个TableLayout由许多TableRow组成，一个TableRow就代表TableLayout中的一行。

##### TextView

通常用于显示文字用的。

##### ImageView

通常用于显示图片用的。

参考资料：<http://www.xuebuyuan.com/1975042.html>