```
项目中添加的任何资源都会在R文件中生成一个相应的资源id
                                                               我们创建的first_layout.xml在R文件中已经生成了一个id
                                 setContentView(R. layout.first_layout)
                                                               在代码中引用布局文件的方法
                                 在AndroidManifest文件中注册
                                             <activity android:name=".FirstActivity" android:label="This is FirstActivity"><intent-filter>
                                 指定主活动
                                             <action android:name="android.intent.action.MAIN"/><category android:name="android.intent.category.LAUNCHER"/></intent-filter></activity>
                     Activity
                                 在活动中使用Toast
                                               Toast是一种非常好的提醒方式,通知用户后一段时间消失  要定义一个触发点
                                                                                                    Button button1=(Button)findViewById(R.id.button_1);
                                                                                                        <item android:id="@+id/add_item"指定id,android=:title="@Add"/>
                                               res目录下创建要给menu文件夹
                                                                     创建一个menu resource file
                                                                                           文件名是main
                                                                                                       title指定名称,id指定标识符
                                 在活动中使用menu
                                                                                                                                   通过指定的资源文件创建菜单
                                                                                                 \verb"getMenuInflater"(). inflate"(R. menu. main, menu)";
                                                在FirstActivity中重写onCreateOptionMenu()方法,可以用ctrl+o快捷键
                                                                                                                                   将菜单项添加到哪一个Menu对象
                                                重写onOptionItemSelected()方法
                                 销毁一个活动 Back键就可以销毁了,但是如果不想按键方式销毁,那么可以通过Activity()方法
                                             intent是android程序中各组件之间进行交互的一种重要方式,可以用来启动活动,启动服务和发送广播
                                              显示intent
                                                          intent意图非常明显Intent intent = new Intent(FirstActivity.this, SecondActivity.class);
                                                        不明确指出要启动哪个活动,指定了抽象的action和category,交给系统去分析Intent
                                                                                                               只有action和category同时匹配,才能响应
                                              Intent (Context packageContext, Class<?>cls) 第一个参数是上下文,第二个参数是要启动的目标活动
                                                            Intent intent = new Intent(Intent.ACTION_VIEW);
                                              更多隐式Intent
                                                            intent. setData(Uri. parse("http://www.baidu.com"));
                                                           intent-filter里面可以再配一个<data>
                                                                                    android:schema, host, port, path, mimeType
                     使用Intent在活动之间穿梭
                                              向下一个活动传递数据 putExtra()方法来重载,可以把我们想要传递数据暂存intent中,启动另一个活动,取出来就可以了
                                                                                                                 第一个参数Intent
                                                                                                                 第二个参数是请求码
                                                                                                                                回调中判断数据的来源
                                                               Activity中有一个startActivityForResult()方法用于启动活动期待返回结果
                                                                                                                               第一个参数向上一个活动返回处理结果
                                                                                                                                                         RESULT_OK/RESULT_CANCELED
                                              返回数据给上一个活动
                                                                                                                 调用getResult()
                                                                                                                               第二个参数把带有数据的Intent传递回去
                                                                                                                 在SecondActivity活动销毁后,会回调FirstActivity上一个活动的onActivityResult()方法
                                                                                                                                                                            三个参数,请求码,返回处理结果,写代intent的data
                                                               如果用户按back键就没法返回了
                                                                                     重写SecondActivity中的onBackPressed()方法
2 探究活动
                                             使用task来管理活动的
                                                              一个任务就是一组存放在栈里的集合
                                                      运行状态
                                                               返回栈的顶端
                                                      暂停状态
                                                               不在栈顶,但是仍然可见,内存低时系统回收
                                             活动状态
                                                               不在栈顶, 不可见
                                                      停止状态
                                                               从返回栈移除
                                                      销毁状态
                                                         onCreate() 活动第一次创建时被调用
                                                                   活动由不可见变为可见时候调用
                                                         onStart()
                                                                   活动准备好和用户进行交互的时候调用,活动位于栈顶,准备运行
                                                         onResume()
                                                         onPause()
                                                                   在系统准备去启动或恢复另一个活动的时候调用,将一些消耗CPU的资源释放掉
                                     返回栈
                                                                  在活动完全不可见的时候调用 如果新活动是一个对话框式的,那么onPause()会执行,但是onStop()不会
                                                         onStop()
                                             活动的生存期
                                                                    活动销毁前调用,活动的状态变为销毁状态
                                                         onDestroy()
                                                         onRestart()
                                                                    停止状态转为运行状态前调用
                                                         完整生存期
                                                                   onCreate()到onDestroy()方法之间所经历的,初始化+释放内存
                                                         可见生存期
                                                                   onStart()和onStop()之间经历的
                                                         前台生存期
                                                                   在onResume()和onPause()之间经历的,可以和用户进行交互的
                     活动的生命周期
                                                         启动活动B的时候活动A停止了,那么按back键的时候不是onRestart()-A,而是onCreate()A 实际上刚刚输入的临时数据全没有了
                                                                                                                                            第一个参数是键
                                             活动被回收了
                                                                                                                            可以用put保存数据
                                                                                                          会携带Bundle类型的参数
                                                                                                                                            第二个参数是真正保存的内容
                                                         onSaveInstanceState()回调方法
                                                                                可以保证活动被会收前一定会被调用
                                                                                                          其实intent也是一样,而且intent可以和Bundle一起用于传递数据
                                                            默认的启动模式  每次一个新的活动,在返回栈中入栈,处于栈顶。不会在乎是否存在,会创建该活动的一个新的实例  返回要多次才可以
                                                   standard
                                                   singleTop
                                                             只有一个相同活动在栈顶 如果不在栈顶,可以创建新的活动
                                                            启动时首先检查返回栈是否存在该活动的实例,存在则出栈,没有再创建
                                                   singleTask
                                     活动的启动模式
                                                               启动一个新的返回栈来管理这个活动,实现不同程序访问这个活动的时候,共用同一个返回栈
                                                               每个程序都会有自己的返回栈  同一个活动在不同的程序里创建了新的实例
                                                   singleInstance
                                                                不太明白,是用两个返回栈来管理就可以实现对同一个活动共用一个返回栈的意思吗
                                                   可以在AndroidManifest.xml里面通过给〈activity〉标签指定android:launchMode属性来指定启动模式
                                                  知道哪一个活动是当前活动 创建一个java类,继承AppCompatActivity
                                                  随时随地退出程序
                                     活动的最佳实践
                                                                  写出intent然后附上关键字的值  合作过程对接困难,不知道有什么参数
                                                   启动活动的最佳写法
                                                                   在SecondActivity中添加一个actionStart()方法,完成了intent的构建。所有SecondActivity需要的数据都用actionStart()方法的参数传递过来
                                                                  这样关键字全在方法的参数上, 比较好查看
```

可以包含用户界面的组件,主要用于和用户进行交互