

使用Python语言实现 Appium自动化测试

APPIUM WITH PYTHON

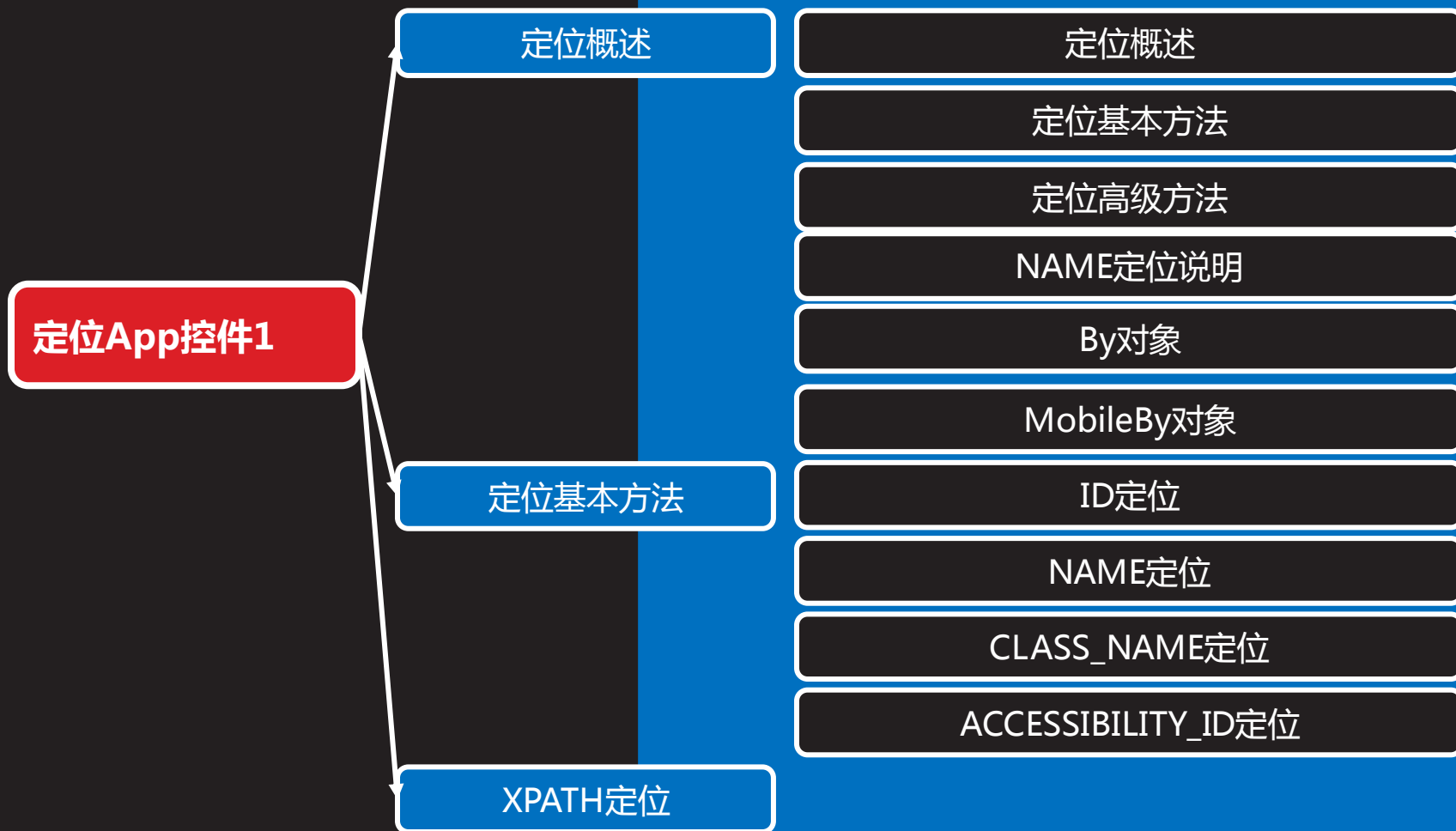
DAY05

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	定位App控件1
	10:30 ~ 11:20	
	11:30 ~ 12:20	
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	
	16:00 ~ 16:50	
	17:00 ~ 17:30	总结和答疑



定位App控件1



定位App控件1（续1）

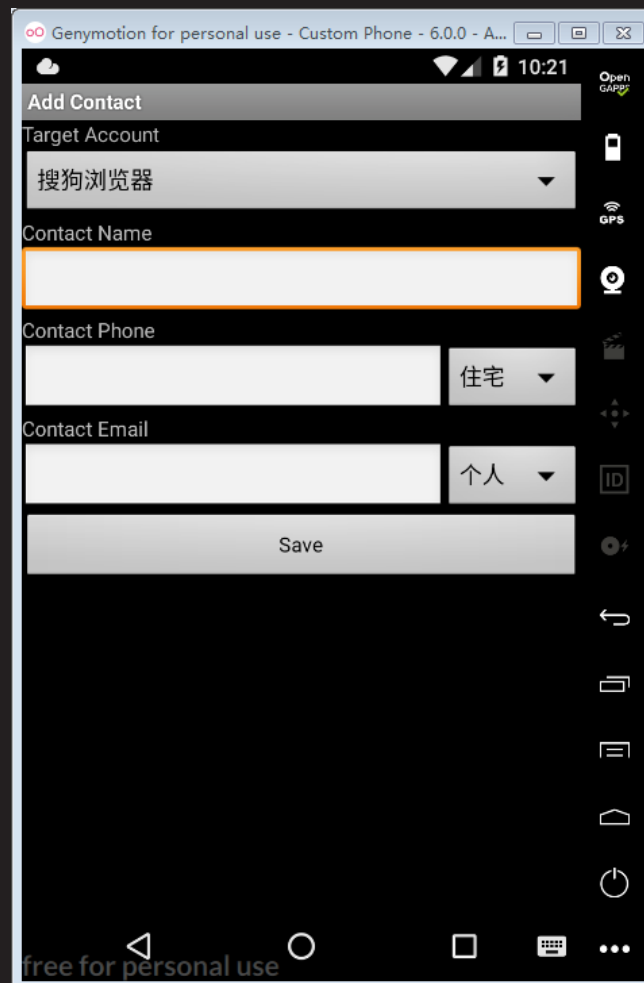
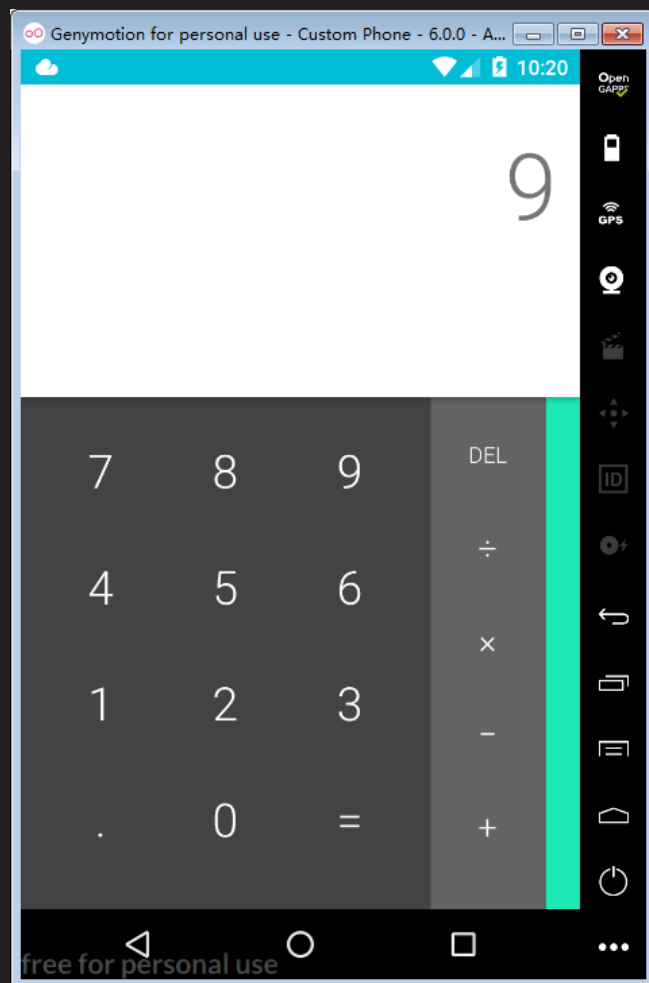


定位概述



定位概述

- 为了操作和断言，需要定位控件，例如一个按钮、一个文本框等。



定位基本方法

- 定位基本方法：
 - ID定位
 - NAME定位
 - CLASS_NAME定位
 - ACCESSIBILITY_ID定位
 - LINK_TEXT定位
 - PARTIAL_LINK_TEXT定位



定位高级方法

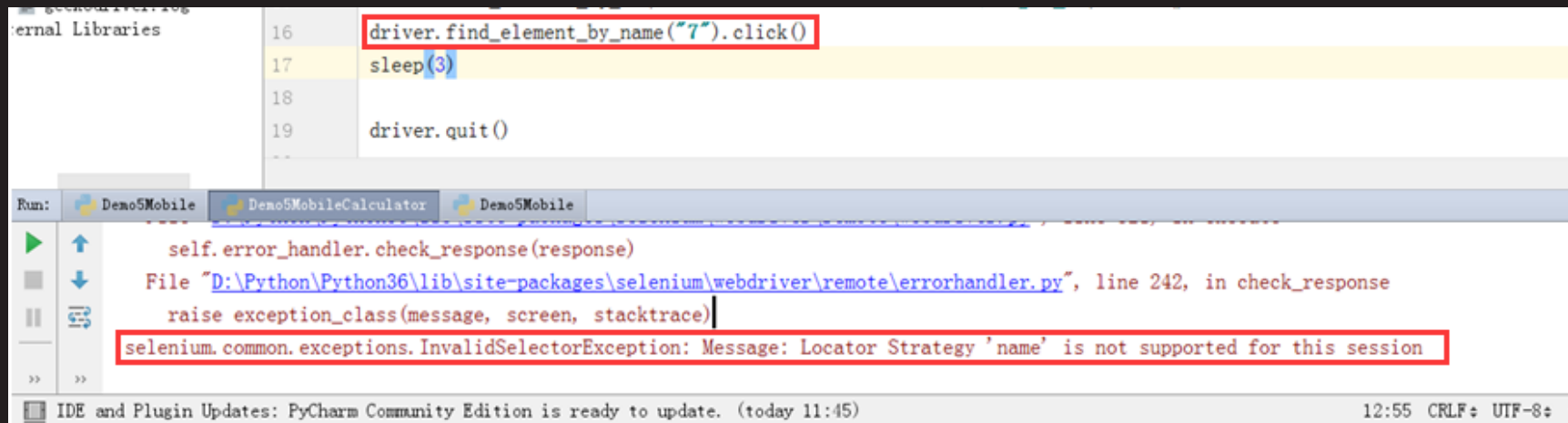
- 定位高级方法：
 - XPATH定位
 - ANDROID_UIAUTOMATOR定位
 - CSS_SELECTOR定位
 - 层级定位
 - 按编号定位



NAME定位说明

- 注意：appium版本在1.5以后就不再支持ByName的定位，如果使用那么定位会报错提示不支持。

org.openqa.selenium.InvalidSelectorException: Locator Strategy 'name' is not supported for this session



The screenshot shows a PyCharm IDE window with a Selenium script and its execution error. The script is as follows:

```
16 driver.find_element_by_name("7").click()
17 sleep(3)
18
19 driver.quit()
```

The error message is displayed in the console:

```
self.error_handler.check_response(response)
File "D:\Python\Python36\lib\site-packages\selenium\webdriver\remote\errorhandler.py", line 242, in check_response
raise exception_class(message, screen, stacktrace)
selenium.common.exceptions.InvalidSelectorException: Message: Locator Strategy 'name' is not supported for this session
```

The IDE status bar at the bottom indicates: IDE and Plugin Updates: PyCharm Community Edition is ready to update. (today 11:45) 12:55 CRLF UTF-8



NAME定位说明（续1）

- 解决方案：修改C:\Program Files\Appium\resources\app\node_modules\appium\node_modules\appium-android-driver\build\lib\driver.js中定位策略的配置中增加name，保存文件后重启Appium Desktop

```
this.locatorStrategies = ['XPATH', 'id', 'class name', 'accessibility id', '-android uiautomator'];
```



```
this.locatorStrategies = ['XPATH', 'id', 'class name', 'accessibility id', '-android uiautomator', 'name'];
```

By对象

- selenium.webdriver.common.by中By对象用于描述控件的定位条件，例如：
 - ID
 - NAME
 - CLASS_NAME
 - XPATH
 - LINK_TEXT
 -
- 导入：from selenium.webdriver.common.by import By



By对象（续1）

- 两种定位页面元素的语法：
 - find_element_by_方式(value)
 - find_element(by,value)
- 说明：find_element_by_方式(value)实际调用的都是find_element(by,value)
- 例如：定位id为abc的元素
 - from selenium.webdriver.common.by import By
 - #方式一
 - driver.find_element_by_id("abc")
 - #方式二
 - driver.find_element(By.ID, "abc")

方式二好处是可以直接把操作的by和value放到一个元组里，然后调用通用方法来传参获得元素结果

```
eid=(By.ID, "abc")
driver.find_element(*eid).click()
```

MobileBy对象

- appium.webdriver.common.mobileby有个MobileBy对象，它是对selenium.webdriver.common.by中By对象的一个扩展。
- 专门用于描述移动App控件的定位条件，例如：
 - ACCESSIBILITY_ID
 - ANDROID_UIAUTOMATOR
 - ANDROID_VIEWTAG
 - ANDROID_DATA_MATCHER
 -
- 导入：


```
from appium.webdriver.common.mobileby
import MobileBy
```

MobileBy导入后，不需要再导入父类By，因为By中的定位方法在MobileBy中都可以使用。

MobileBy对象 (续1)

- 例 :
 - from appium.webdriver.common.mobileby import MobileBy
 - #driver.find_element(By.ID,"ag2")
 - driver.find_element(MobileBy.ID,"ag2")

MobileBy类

By类

ID、NAME、XPATH、
CLASS_NAME、
CSS_SELECTOR、
LINK_TEXT、
PARTIAL_LINK_TEXT

ACCESSIBILITY_ID、
ANDROID_UIAUTOMATOR
.....



定位基本方法



ID定位

- Android里面定位的id一般为resrouce-id
- 两种语法格式例：
 - `driver.find_element(MobileBy.ID,"digit_9").click()`
 - `driver.find_element_by_id("digit_9").click()`
 - `driver.find_element_by_id("com.android.calculator2:id/digit_9").click()`

Node Detail	
index	2
text	9
resource-id	com.android.calculator2:id/digit_9
class	android.widget.Button
package	com.android.calculator2
content-desc	



ID定位（续1）

- 说明：没显示id,有以下几种可能
 - 1.开发没添加
 - 2.android是4.4以下的版本
 - 3.目标设备的API Level低于18

Node Detail	
index	0
text	
resource-id	
class	android.widget.LinearLayout
package	com.android.calculator2
content-desc	



NAME定位

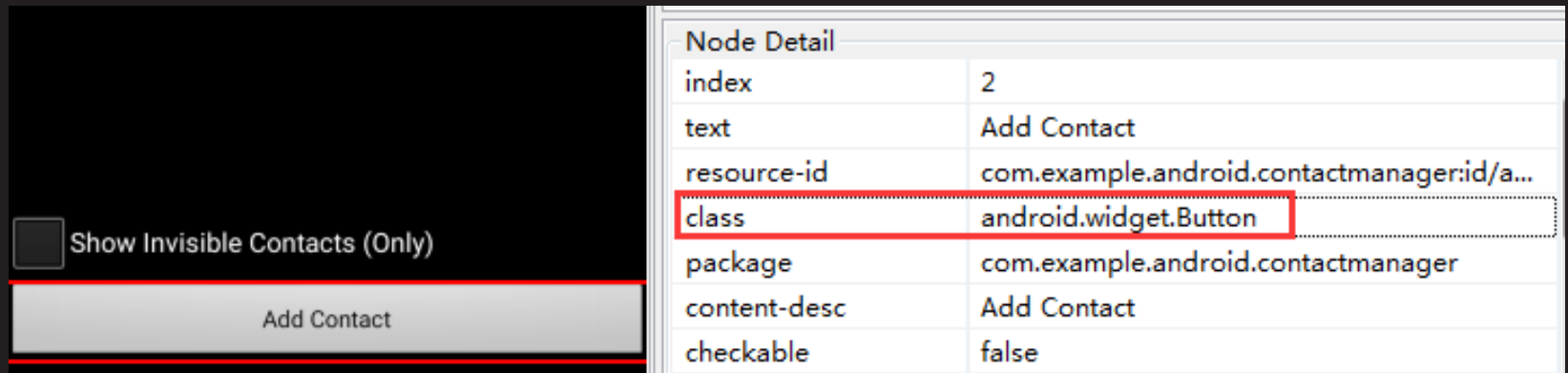
- Android里面定位的name一般为text
- 两种语法格式例：
 - `driver.find_element(MobileBy.NAME,"7").click()`
 - `driver.find_element_by_name("7").click()`

Node Detail	
index	0
text	7
resource-id	com.android.calculator2:id/digit_7
class	android.widget.Button
package	com.android.calculator2
content-desc	
checkable	false
checked	false



CLASS_NAME定位

- Android的class属性对应CLASS_NAME定位方式。
- 两种语法格式例：
 - `driver.find_element(MobileBy.CLASS_NAME, "android.widget.Button").click()`
 - `driver.find_element_by_class_name("android.widget.Button").click()`



Node Detail	
index	2
text	Add Contact
resource-id	com.example.android.contactmanager:id/a...
class	android.widget.Button
package	com.example.android.contactmanager
content-desc	Add Contact
checkable	false



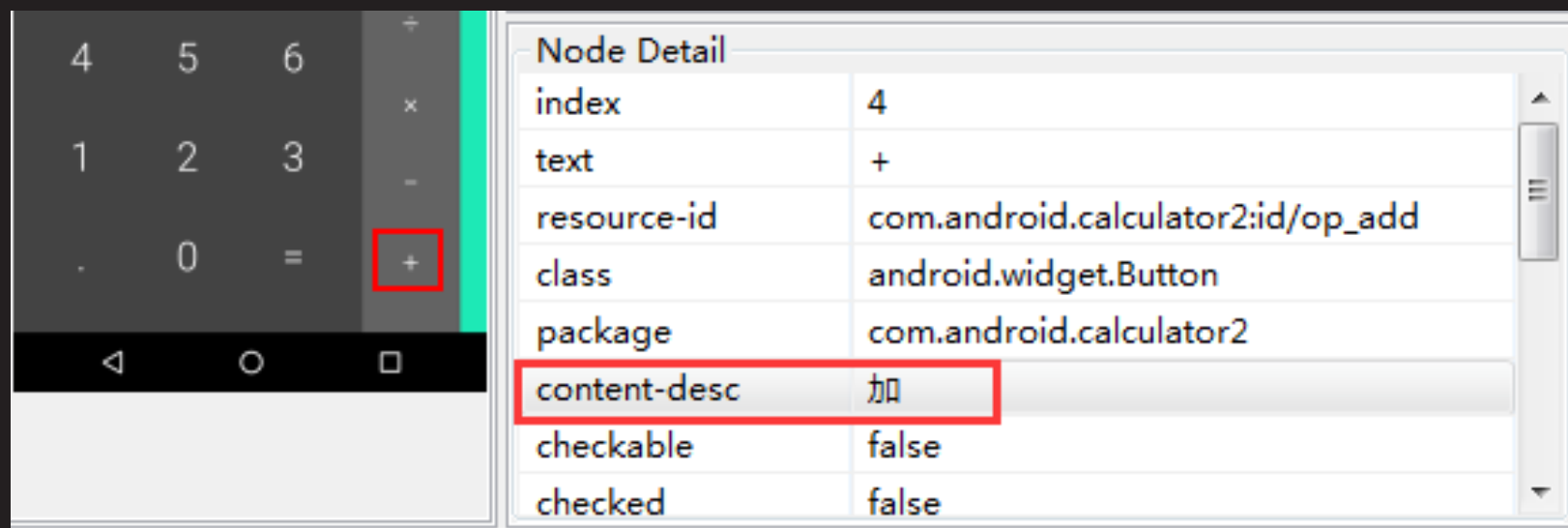
CLASS_NAME定位（续1）

- 注：一般一个页面上的class属性不唯一，如果元素不唯一定位会报错。
- 所以CLASS_NAME定位一般用于后续所讲的层级定位中或按编号定位中。
- 说明：
 - 层级定位先定位到dom树中上一级节点的元素，再定位目标元素。
 - 按编号定位可以通过index来获取需要的元素。（从0开始查找dom树中的同名class属性）



ACCESSIBILITY_ID定位

- Android的content-desc属性对应ACCESSIBILITY_ID定位方式，如果这个属性不为空则推荐使用。
- 两种语法格式例：
 - `driver.find_element(MobileBy.ACCESSIBILITY_ID,"加").click()`
 - `driver.find_element_by_accessibility_id("加").click()`



ACCESSIBILITY_ID定位（续1）

- Accessibility ID是什么？
 - 在Android上面就等同于“内容描述”（contentDescription）。
 - 这个属性的主要功能就是为视力有障碍的人增加对控件的解释。

由于有些控件没有text属性，对于一些用户来说，无法理解这个控件是做什么的，这时用户点击这个控件，android系统会自动使用人声朗读控件上contentDescription属性的内容，这样用户就可以知道这个控件是做什么用的。
 - 鉴于这是一个隐藏属性，而Android上用于查找控件的各种属性可能有所缺失或者有重复，所以跟开发团队沟通好每个view都赋予一个唯一的contentDescription。



XPATH定位



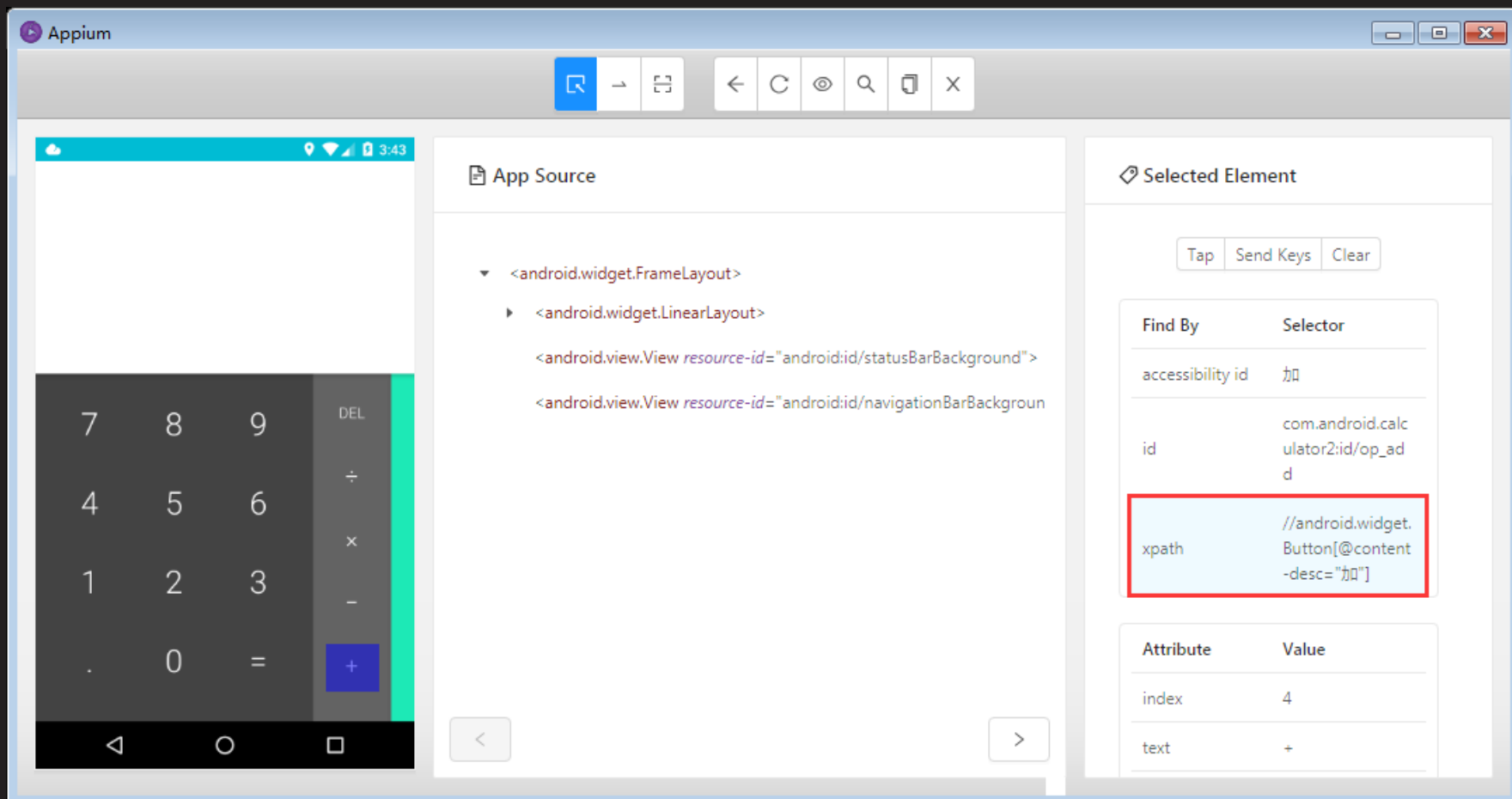
XPATH表达式

- XPATH 是一门在 XML 文档中通过路径表达式来查找节点或节点集等信息的语言。
- XPATH 用于在 XML 文档中通过元素和属性进行导航。
- 在前面课程已经简单的介绍了用工具查看目标元素的 XPATH地址,工具查看比较死板,不够灵活,有时候直接复制粘贴会定位不到。这个时候就需要自己手动的去写 XPATH表达式。
- XPATH分为：绝对路径和相对路径两种写法。
- 说明：Appium自动化中使用XPATH定位效率相对慢些。



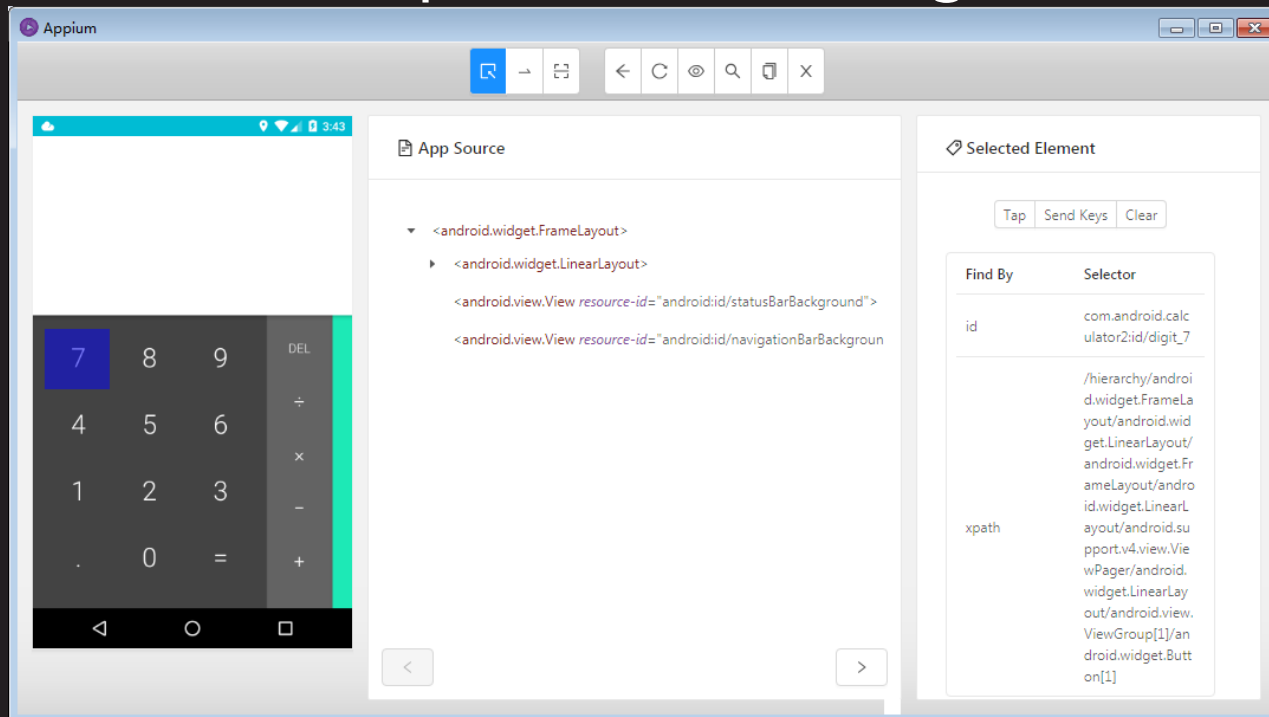
XPATH表达式 (续1)

- Inspector里有XPATH表达式，例如计算器的加号
 - `//android.widget.Button[@content-desc="加"]`



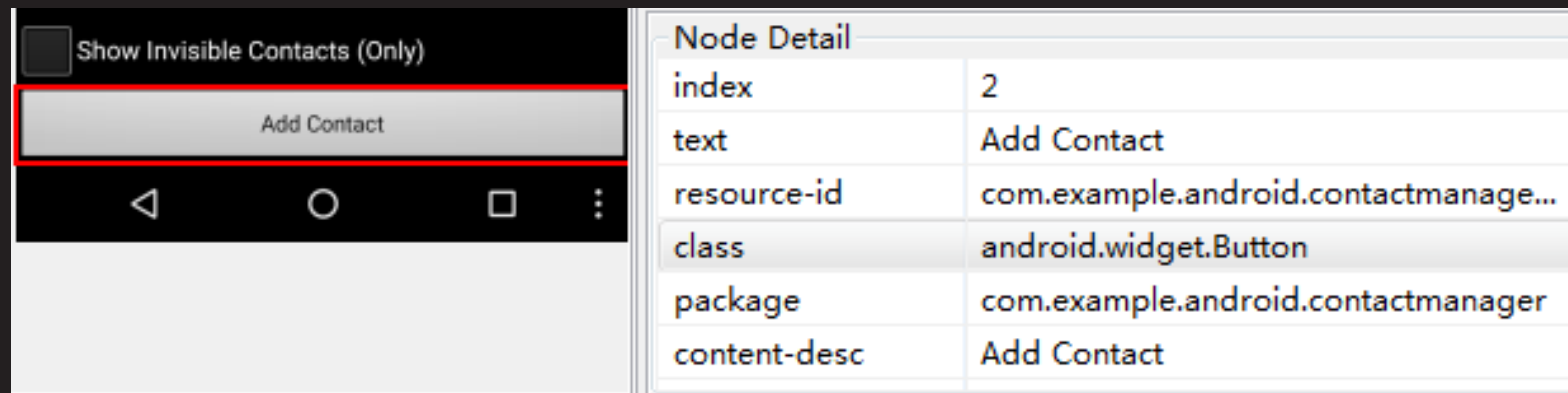
XPATH表达式 (续2)

- /hierarchy/android.widget.FrameLayout/android.widget.LinearLayout/android.widget.FrameLayout/android.widget.LinearLayout/android.support.v4.view.ViewPager/android.widget.LinearLayout/android.view.ViewGroup[1]/android.widget.Button[1]



XPATH定位

- 找到某一层特殊class属性值的祖先，从这一层开始，使用class属性值一层一层描述待定位页面元素的源码路径。
- 两种语法格式例：
 - `driver.find_element(MobileBy.XPATH,"//android.widget.Button").click()`
 - `driver.find_element_by_xpath("//android.widget.Button").click()`



Node Detail	
index	2
text	Add Contact
resource-id	com.example.android.contactmanage...
class	android.widget.Button
package	com.example.android.contactmanager
content-desc	Add Contact

XPATH定位（续1）

- XPATH定位中的标签之间的父子关系和兄弟关系：
 - 父子关系：上下级关系。
 - 兄弟关系：平级关系。

```

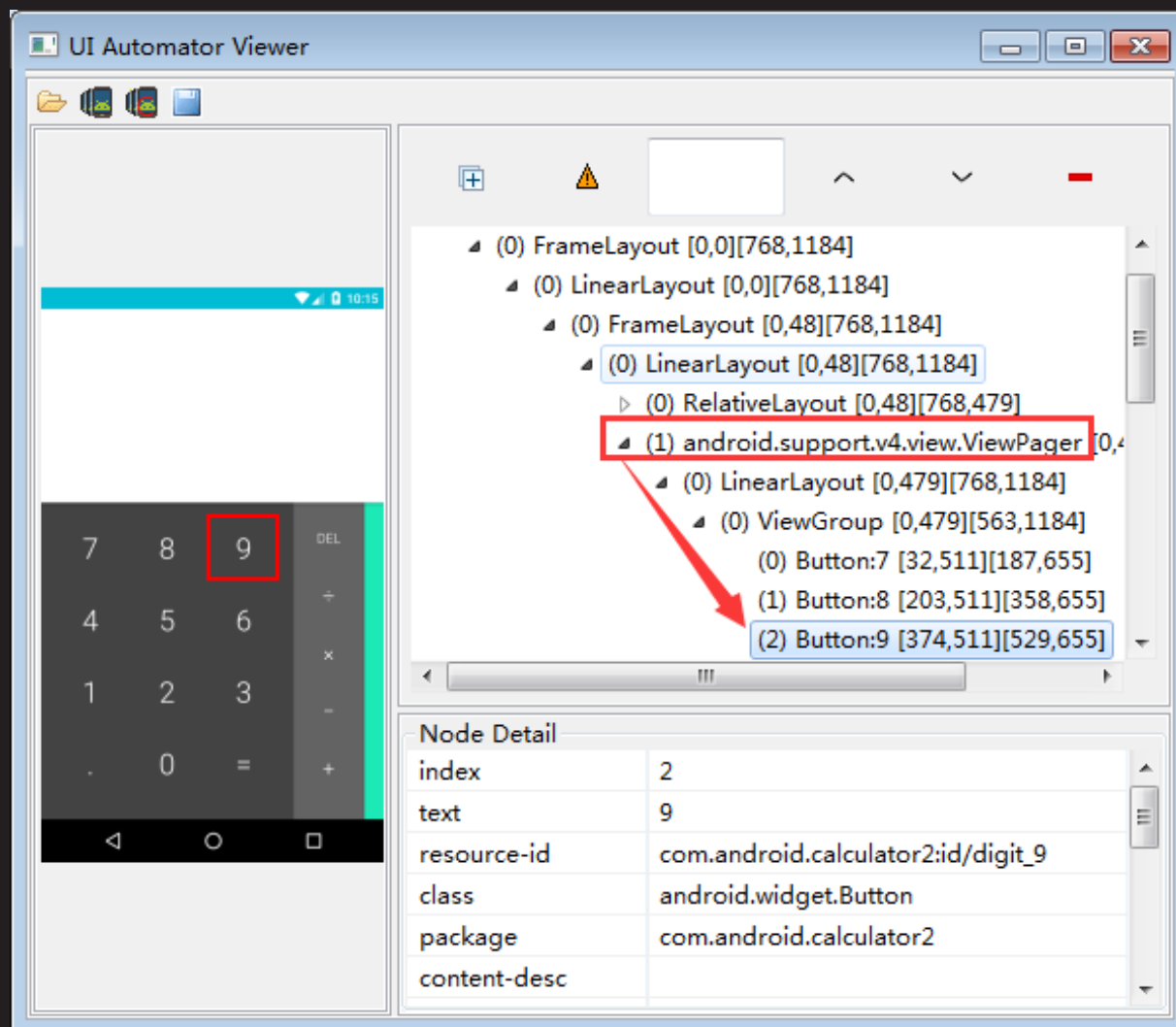
    ▲ (0) RelativeLayout [0,48][768,479]
        (0) EditText [0,48][768,283]
        (1) EditText [0,283][768,479]
    ▲ (1) android.support.v4.view.ViewPager [0,479][768,1184]
        ▲ (0) LinearLayout [0,479][768,1184]
            ▲ (0) ViewGroup [0,479][563,1184]
                (0) Button:7 [32,511][187,655]
                (1) Button:8 [203,511][358,655]
                (2) Button:9 [374,511][529,655]
                (3) Button:4 [32,671][187,815]
                (4) Button:5 [203,671][358,815]
    
```

父子关系

兄弟关系

XPATH定位（续2）

- 查看页面元素的源代码层级和各个层级的Class属性值。



注意：Appium中属性class的代表标签的名字。所以此XPATH表达式的/后只可以接Android的class属性值或*
而且：
appium1.6.3版本后，XPATH中的class属性值要使用全类名，否则会报错。

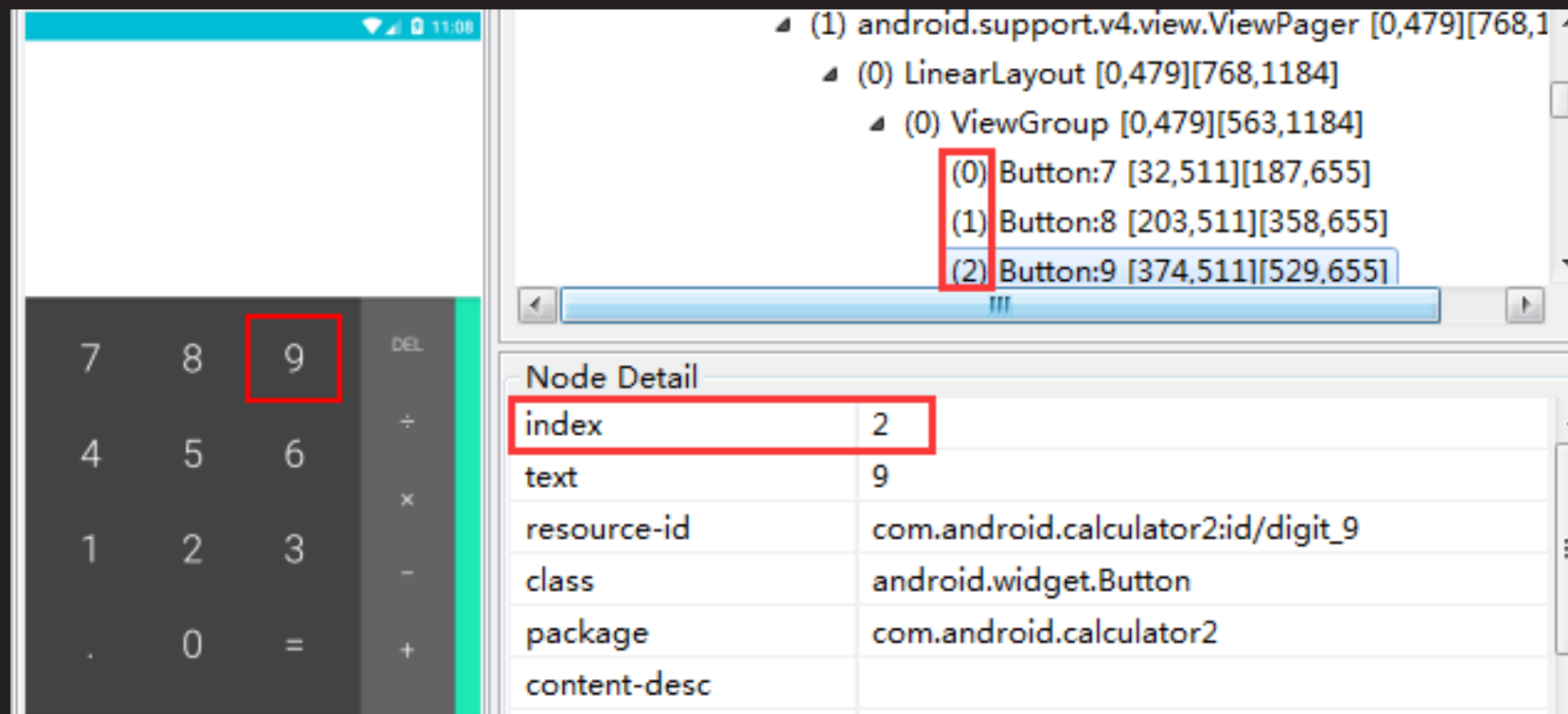
XPATH谓词

- XPATH谓词：在XPATH表达式中使用[]描述筛选条件。

- 常见XPATH谓词形式：

- 编号
- 属性值

注意：XPATH语法中编号从1开始，不是从0开始。所以不等于index属性值，因为index属性值从0开始。



The screenshot illustrates the relationship between a UI element in an application and its representation in the XPATH hierarchy. On the left, a calculator app shows the digit '9' selected. On the right, the UI hierarchy tree shows the following structure:

- (1) android.support.v4.view.ViewPager [0,479][768,1184]
 - (0) LinearLayout [0,479][768,1184]
 - (0) ViewGroup [0,479][563,1184]
 - (0) Button:7 [32,511][187,655]
 - (1) Button:8 [203,511][358,655]
 - (2) Button:9 [374,511][529,655]

The 'Node Detail' panel for the selected button (index 2) shows the following properties:

Property	Value
index	2
text	9
resource-id	com.android.calculator2:id/digit_9
class	android.widget.Button
package	com.android.calculator2
content-desc	



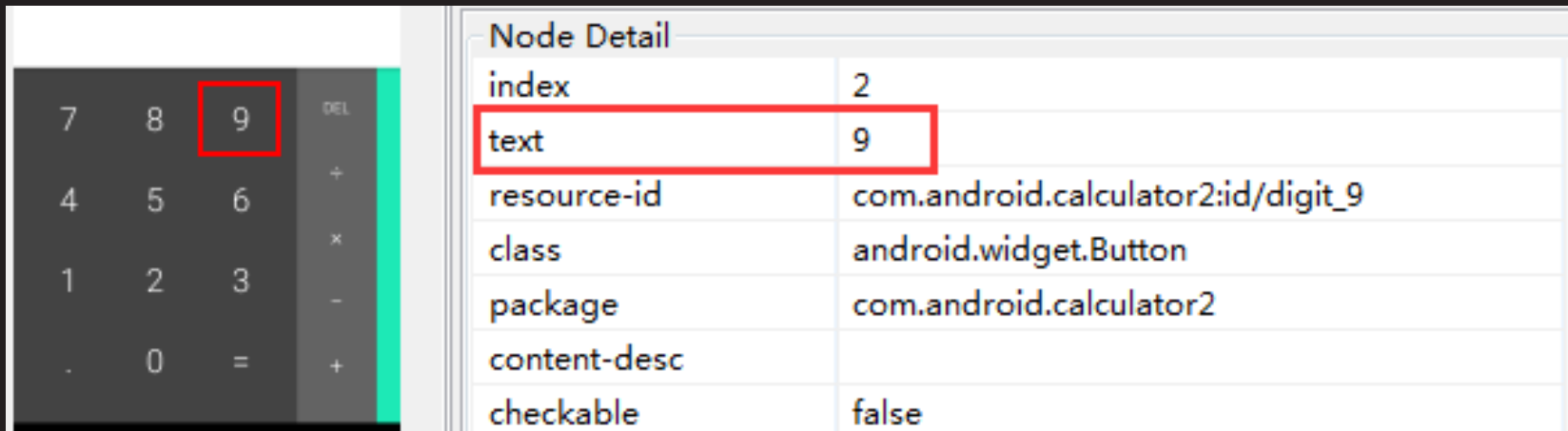
XPATH谓词（续1）

- 编号XPATH谓词例：
 - `driver.find_element(MobileBy.XPATH,"//android.support.v4.view.ViewPager/android.widget.LinearLayout/android.view.ViewGroup[1]/android.widget.Button[3]").click()`
 - `driver.find_element_by_xpath("//android.support.v4.view.ViewPager/android.widget.LinearLayout/android.view.ViewGroup[1]/android.widget.Button[3]").click()`
 - `driver.find_element_by_xpath("//android.widget.Button[3]").click()`



XPATH谓词（续2）

- 属性值XPATH谓词例1：
 - `driver.find_element(MobileBy.XPATH,"//android.widget.Button [@text='9']").click()`
 - `driver.find_element_by_xpath("//android.widget.Button [@text='9']").click()`

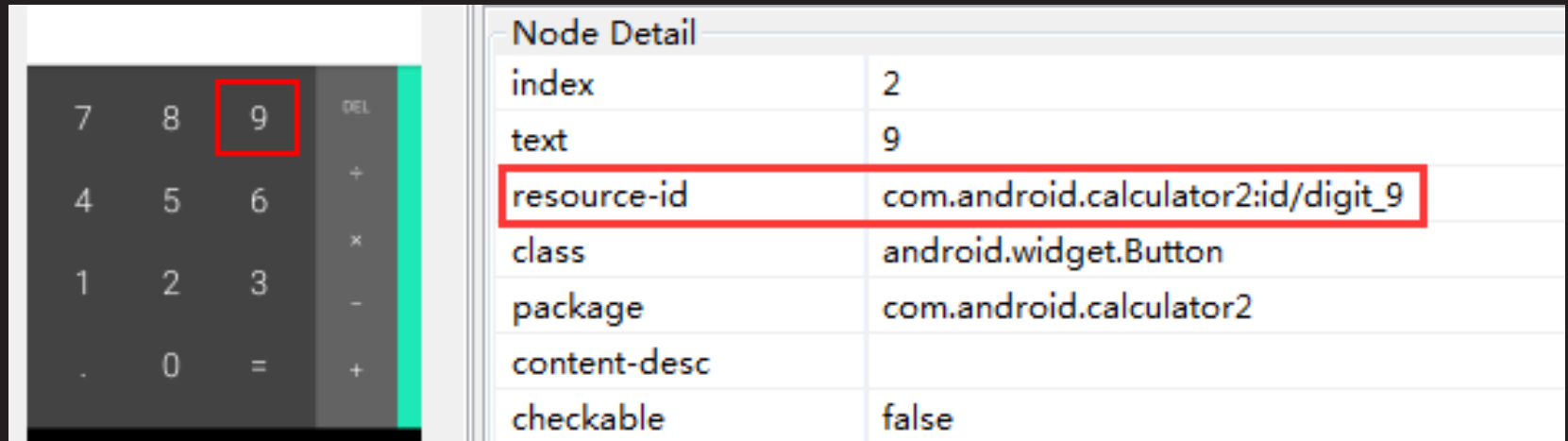


Node Detail	
index	2
text	9
resource-id	com.android.calculator2:id/digit_9
class	android.widget.Button
package	com.android.calculator2
content-desc	
checkable	false



XPATH谓词（续3）

- 属性值XPATH谓词例2：
 - driver.find_element(MobileBy.XPATH,"//android.widget.Button [@resource-id='com.android.calculator2:id/digit_9']").click()
 - driver.find_element_by_xpath("//android.widget.Button[@index='2']").click()




The image shows a mobile calculator interface on the left and a 'Node Detail' table on the right. In the calculator, the digit '9' is highlighted with a red box. In the Node Detail table, the 'resource-id' row is highlighted with a red box.

Node Detail	
index	2
text	9
resource-id	com.android.calculator2:id/digit_9
class	android.widget.Button
package	com.android.calculator2
content-desc	
checkable	false

XPATH谓词（续4）

- 属性值XPATH谓词例3：
 - driver.find_element(MobileBy.XPATH,"//android.widget.CheckBox[@checkable='true']").click()

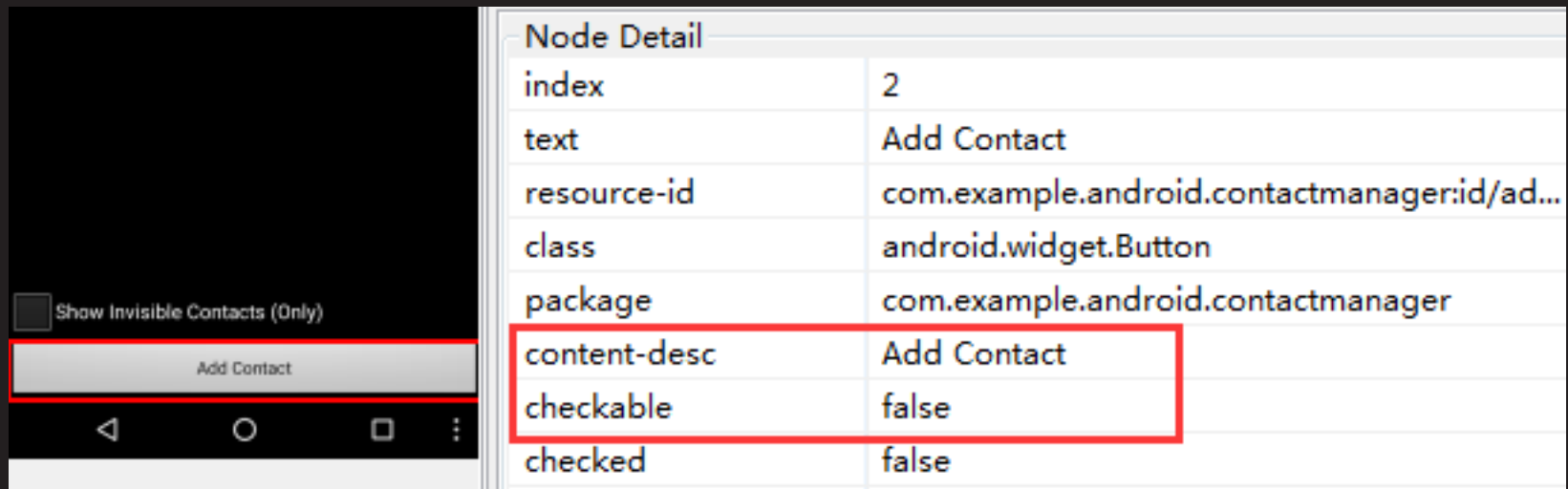


class	android.widget.CheckBox
package	com.example.android.contactmanager
content-desc	Show Invisible Contacts (Only)
checkable	true
checked	false
clickable	true
enabled	true
focusable	true
focused	false



XPATH谓词 (续5)

- 多条件筛选 :
 - `driver.find_element(MobileBy.XPATH,"//android.widget.Button[@content-desc='Add Contact'][@checkable='false']").click()`
 - `driver.find_element(MobileBy.XPATH,"//android.widget.Button[@content-desc='Add Contact'and @checkable='false']").click()`



Node Detail	
index	2
text	Add Contact
resource-id	com.example.android.contactmanager:id/ad...
class	android.widget.Button
package	com.example.android.contactmanager
content-desc	Add Contact
checkable	false
checked	false

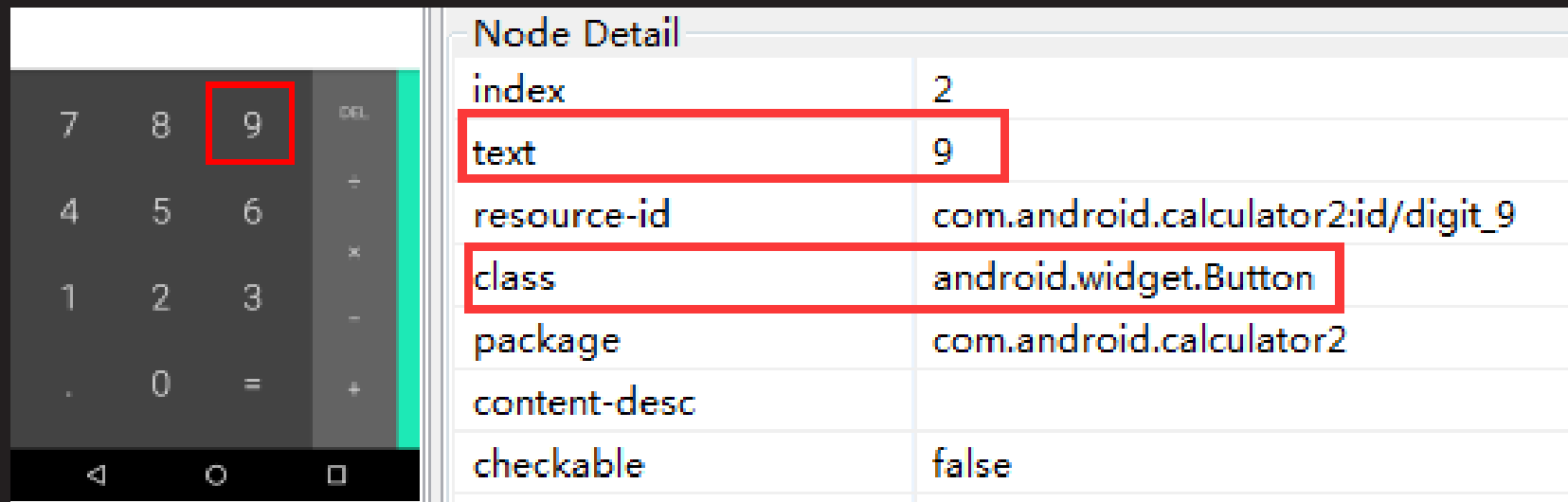
XPATH定位计算器做随机计算

- 在计算器中，使用XPATH定位实现两个随机一位数的加法计算的步骤。
- 详见【COOKBOOK】



XPATH通配符

- 通配符：*（星号）可以匹配任意个数的任意字符。
- 说明：一般用*来匹配class属性名或其他属性名
- 例1：
 - `driver.find_element(MobileBy.XPATH,"//*[@text='9']")`
`.click()`



Node Detail	
index	2
text	9
resource-id	com.android.calculator2:id/digit_9
class	android.widget.Button
package	com.android.calculator2
content-desc	
checkable	false

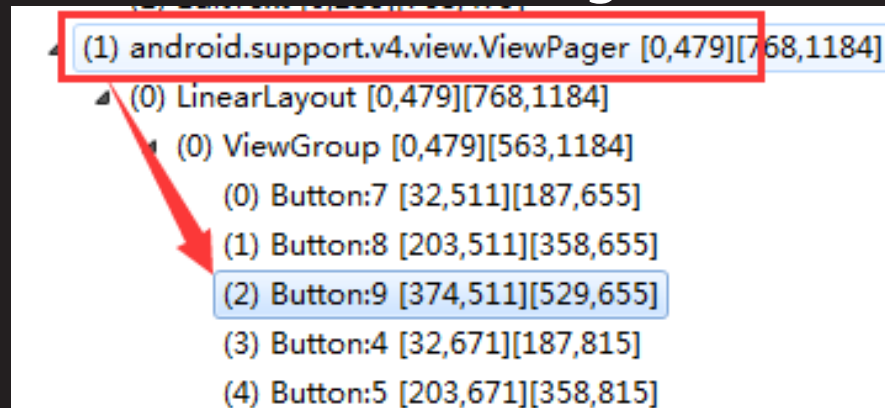
XPATH分隔符

- 常用分隔符号：

- . (一个点) 代表当前节点。
- .. (两个点) 代表上一级节点 (也就是父节点)。
- / (一个斜线) 前后的节点是父子关系。
- // (二个斜线) 前后的节点是祖先和后代关系。

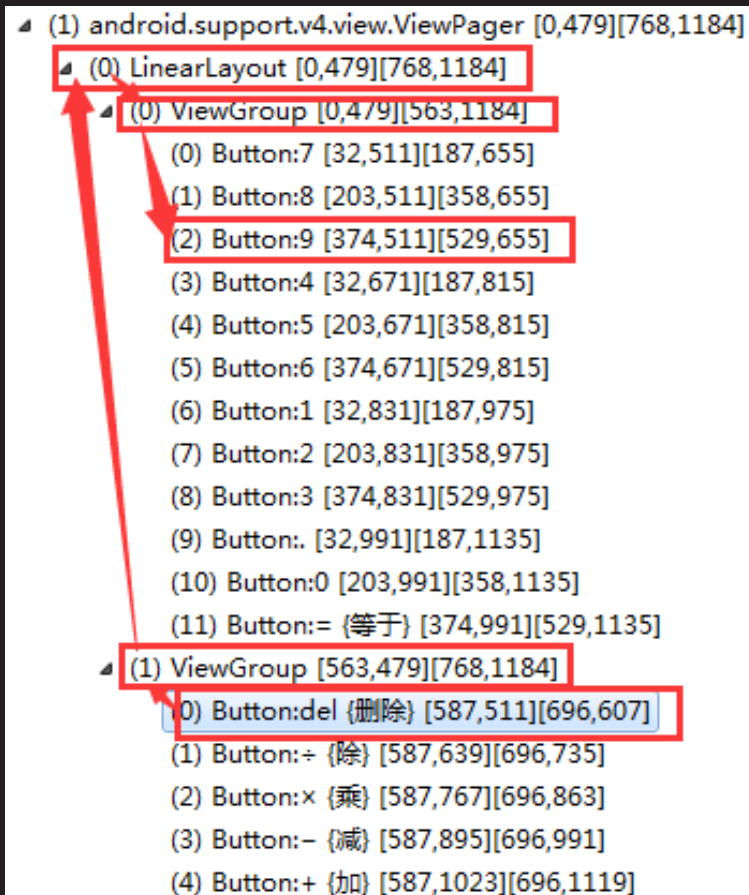
- 例1：

- driver.find_element(MobileBy.XPATH,"//android.support.v4.view.ViewPager//android.widget.Button[3]").cli



XPATH分隔符 (续1)

- 例2 :
 - driver.find_element(MobileBy.XPATH,"//*[@text='del']
/../../*[1]/*[2]").click()

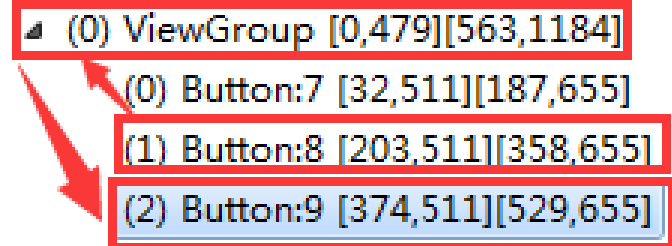


XPATH定位计算器2

- XPATH定位计算器中通过文本是8的控件找上一级再找下一级方式找到9
- 详见【COOKBOOK】

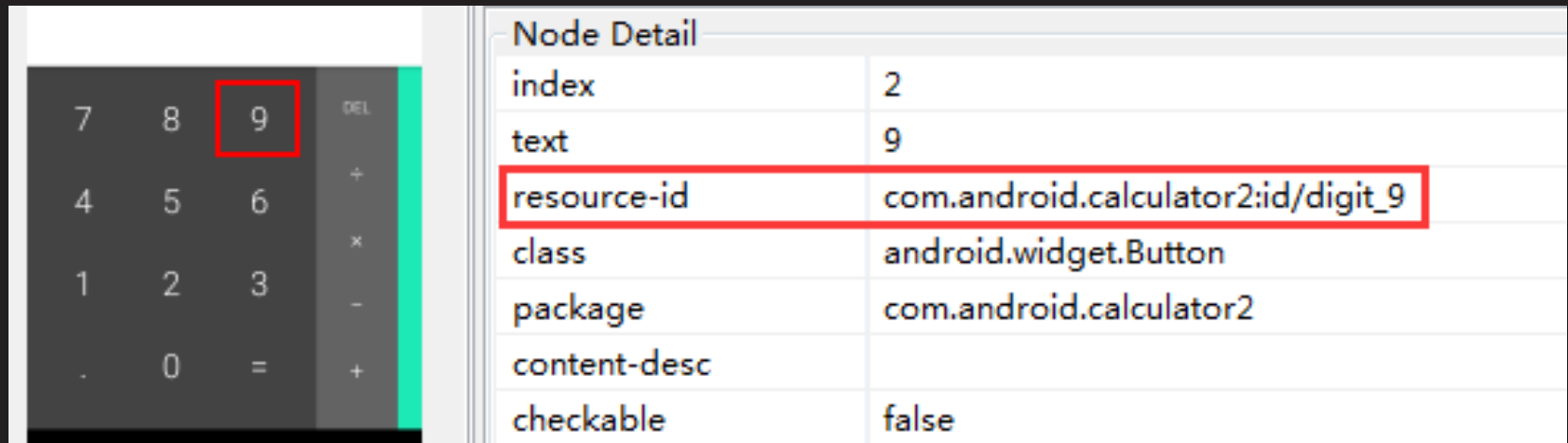
```

▲ (1) android.support.v4.view.ViewPager [0,479][768,1184]
  ▲ (0) LinearLayout [0,479][768,1184]
    ▲ (0) ViewGroup [0,479][563,1184]
      (0) Button:7 [32,511][187,655]
      (1) Button:8 [203,511][358,655]
      (2) Button:9 [374,511][529,655]
      (3) Button:4 [32,671][187,815]
      (4) Button:5 [203,671][358,815]
      (5) Button:6 [374,671][529,815]
      (6) Button:1 [32,831][187,975]
      (7) Button:2 [203,831][358,975]
      (8) Button:3 [374,831][529,975]
      (9) Button:. [32,991][187,1135]
  
```




XPATH函数

- 常用函数：
 - contains (某属性,值) 用于实现包含某值的判断。
- 例：
 - `driver.find_element(MobileBy.XPATH,"//*[contains(@resource-id,'digit_9')]").click()`

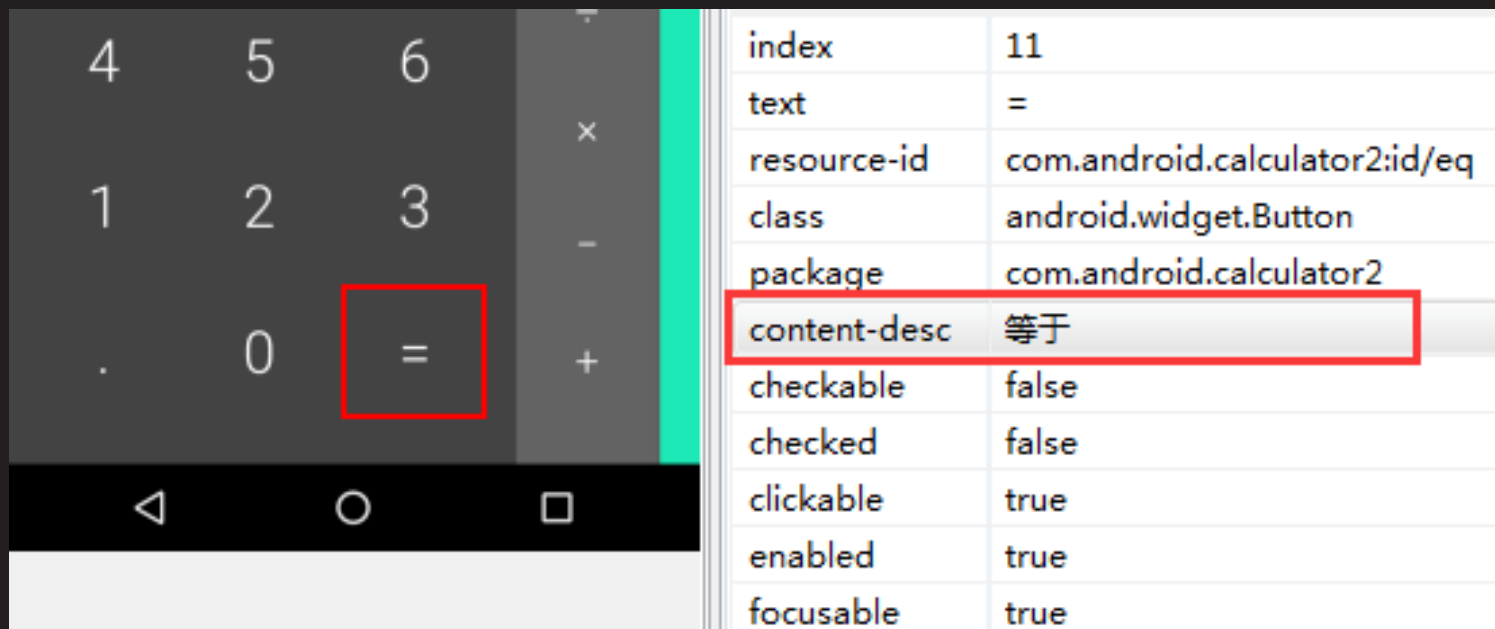


Node Detail	
index	2
text	9
resource-id	com.android.calculator2:id/digit_9
class	android.widget.Button
package	com.android.calculator2
content-desc	
checkable	false



XPATH函数（续1）

- 常用函数：
 - contains（某属性,值）用于实现包含某值的判断。
- 例：
 - `driver.find_element(MobileBy.XPATH,"//android.widget.Button[contains(@content-desc,'等')]").click()`



index	11
text	=
resource-id	com.android.calculator2:id/eq
class	android.widget.Button
package	com.android.calculator2
content-desc	等于
checkable	false
checked	false
clickable	true
enabled	true
focusable	true

XPATH轴

- 轴 (Axis) : 有些比较特殊的情况中,仅靠这些比较基本的XPATH表达式还不够灵活,所以今天继续介绍几种比较常用的基于 “轴(Axes)” 的表达式。

XPATH轴关键字	轴的定义说明
parent	选取当前节点的父节点
child	选取当前节点的子节点
ancestor	选取当前节点的所有上层节点
descendant	选取当前节点所有下层节点
following	选取当前节点之后显示的所有节点
following-sibling	选取当前节点所有的平级节点
preceding	选取当前节点前面所有的节点
preceding-sibling	选取当前节点前面所有平级的节点



XPATH轴 (续1)

- 例1：找到哥哥节点
 - driver.find_element(MobileBy.XPATH,"//android.widget.Button[@text='8']/preceding-sibling::*").click()

```

    ▲ (1) android.support.v4.view.ViewPager [0,479][768,1184]
      ▲ (0) LinearLayout [0,479][768,1184]
        ▲ (0) ViewGroup [0,479][563,1184]
          (0) Button:7 [32,511][187,655]
          (1) Button:8 [203,511][358,655]
          (2) Button:9 [374,511][529,655]
          (3) Button:4 [32,671][187,815]
          (4) Button:5 [203,671][358,815]
          (5) Button:6 [374,671][529,815]
          (6) Button:1 [32,831][187,975]
          (7) Button:2 [203,831][358,975]
          (8) Button:3 [374,831][529,975]
          (9) Button:.. [32,991][187,1135]
          (10) Button:0 [203,991][358,1135]
          (11) Button:= {等于} [374,991][529,1135]

```

Node Detail

index	1
text	8
resource-id	com.android.calculator2:id/digit_8
class	android.widget.Button



XPATH轴（续2）

• 例1：找到弟弟节点

- driver.find_element(MobileBy.XPATH,"//*[@text='5']/following-

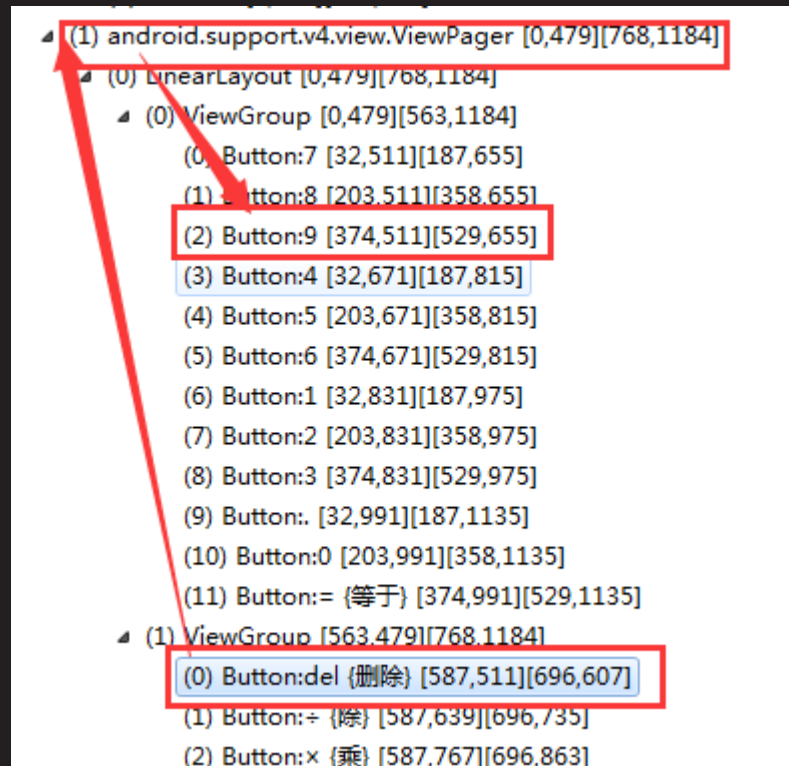
@text='3']")).click()

```

(0) ViewGroup [0,479][563,1184]
  (0) Button:7 [32,511][187,655]
  (1) Button:8 [203,511][358,655]
  (2) Button:9 [374,511][529,655]
  (3) Button:4 [32,671][187,815]
  (4) Button:5 [203,671][358,815]
  (5) Button:6 [374,671][529,815]
  (6) Button:1 [32,831][187,975]
  (7) Button:2 [203,831][358,975]
  (8) Button:3 [374,831][529,975]
  (9) Button: [32,991][187,1135]
  (10) Button:0 [203,991][358,1135]
  (11) Button:= {等于} [374,991][529,1135]
(1) ViewGroup [563,479][768,1184]
  
```

XPATH轴 (续3)

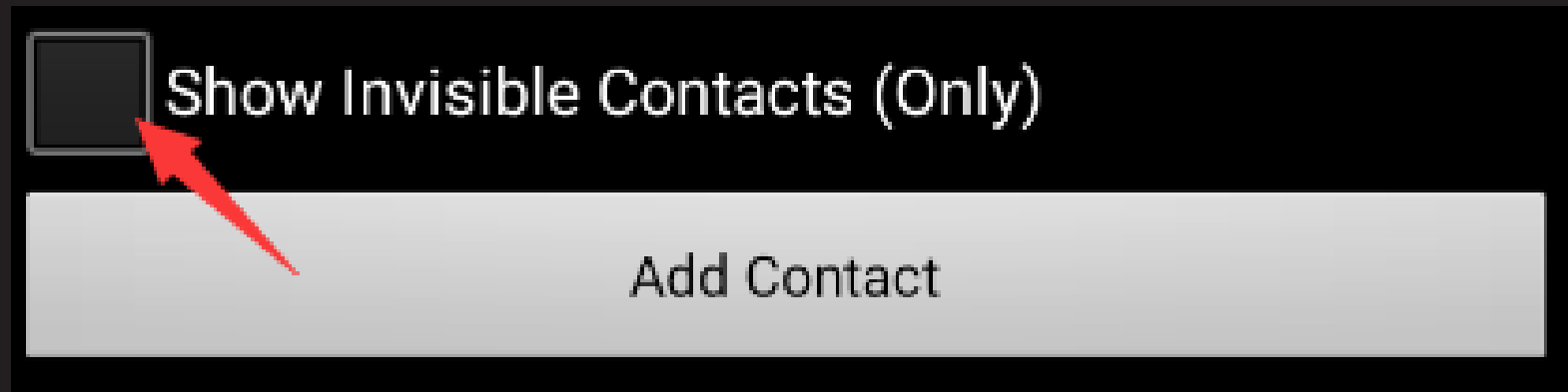
- 例3：找到祖先节点
 - driver.find_element(MobileBy.XPATH,"//*[@text='del']
/ancestor::*[contains(@resource-
id,'pager')]/android.widget.Button[3]").click()



XPATH定位ContactManager

- 使用XPATH定位ContactManager中通过Add Contact按钮查找到上面的复选框。
- 详见【COOKBOOK】

课堂练习



总结和答疑

