

# UiAutomator UiObject类API (三)

By Zhang Jingwei

20160222

# 概要

- UiObject类介绍
- 点击与长按对象
- 拖拽与滑动对象
- 输入文本与清除文本
- 获取对象属性与属性判断
- 手势操作
- 判断对象是否存在

# UIObject类介绍

- UIObject类说明



- UIObject功能

代表一个组件对象，对象有许多模拟操作手机的方法和属性



# 点击与长按

## 组件区域

- Rect 对象代表一个区域[left,top][right,bottom]

# 点击与长按API

返回值	方法名	描述
boolean	click()	点击对象
boolean	clickAndWaitForNewWindow()	点击对象，等待新窗口打开，默认超时5500ms
boolean	clickAndWaitForNewWindow( <u>timeout</u> )	点击对象，等待新窗口打开，自定义超时时间
boolean	clickBottomRight()	点击对象右下角
boolean	clickTopLeft()	点击对象左上角
boolean	longClick()	长按对象，执行长按操作
boolean	longClickBottomRight()	长按对象右下角
boolean	longClickTopLeft()	长按对象左上角

# 示例

```
public void testClick() throws UiObjectNotFoundException{  
    //Click,点击Chrome  
    UiObject chromelcon=new UiObject(new UiSelector().description("Chrome"));  
    chromelcon.click();  
    sleep(1500);  
    UiDevice.getInstance().pressHome();  
    chromelcon.clickAndWaitForNewWindow(); //默认超时5500ms  
    sleep(1500);  
    UiDevice.getInstance().pressHome();  
    chromelcon.clickAndWaitForNewWindow(3000);  
    sleep(1500);  
    UiDevice.getInstance().pressHome();  
    chromelcon.clickBottomRight();  
    sleep(1500);  
    UiDevice.getInstance().pressHome();  
    chromelcon.clickTopLeft();  
    sleep(1500);  
    UiDevice.getInstance().pressHome();  
}
```

# 示例

```
public void testLongClick() throws UiObjectNotFoundException{  
//longClick,长按  
UiObject chromelcon=new UiObject(new UiSelector().description("Chrome"));  
chromelcon.longClick();  
sleep(1500);  
chromelcon.longClickBottomRight();  
sleep(1500);  
chromelcon.longClickTopLeft();  
}
```



# 拖拽与滑动

- 拖拽到指定位置（点或另一个组件）
- 滑动一小段距离

# 拖拽与滑动API

返回值	方法名	描述
boolean	<code>dragTo(<u>destObj</u>, <u>steps</u>)</code>	拖拽对象到另一个对象位置上，步长可设置拖动速度
boolean	<code>dragTo(<u>destX</u>, <u>destY</u>, <u>steps</u>)</code>	拖拽对象到屏幕某个坐标位置上，步长可设置拖动速度
boolean	<code>swipeDown(<u>steps</u>)</code>	拖动对象往下滑动
boolean	<code>swipeLeft(<u>steps</u>)</code>	拖动对象往左滑动
boolean	<code>swipeRight(<u>steps</u>)</code>	拖动对象往右滑动
boolean	<code>swipeUp(<u>steps</u>)</code>	拖动对象往上滑动

# 示例

```
public void testDrag() throws UiObjectNotFoundException{  
//dragto,拖拽chrome  
UiObject chromelcon=new UiObject(new UiSelector().description("Chrome"));  
chromelcon.dragTo(883, 460, 50);  
sleep(1500);  
UiObject weChatIcon=new UiObject(new UiSelector().description("微信"));  
chromelcon.dragTo(weChatIcon, 50);  
}
```

# 示例

```
public void testSwipe() throws UiObjectNotFoundException{  
//swipe,滑动recentapp  
UiObject chromelcon=new UiObject(new  
UiSelector().resourceId("com.android.systemui:id/task_view_thumbnail"));  
chromelcon.swipeUp(50);  
sleep(1500);  
chromelcon.swipeDown(50);  
sleep(1500);  
chromelcon.swipeLeft(10);  
sleep(1500);  
chromelcon.swipeRight(10);  
}
```

# 输入文本与清除文本

## 1.相关API

返回值	方法名	描述
boolean	setText( <u>text</u> )	在对象中输入文本
void	clearTextField()	清除编辑框中的文本

## 2.实现步骤

- 1) 输入文本：清除文本->输入文本
- 2) 清除文本：长按->清除

# 示例

```
public void testSetTextAndClearText() throws UiObjectNotFoundException{  
//输入清除网址  
UiObject urlRect=new UiObject(new  
UiSelector().resourceId("com.android.chrome:id/url_bar"));  
urlRect.setText("www.163.com");  
sleep(2000);  
urlRect.clearTextField();  
}
```

# 获取对象的属性与属性判断

## 1. 获取对象属性API

返回值	方法名	描述
Rect	<code>getBounds()</code>	获取对象矩形坐标
int	<code>getChildCount()</code>	获得下一级子类数量
String	<code>getClassName()</code>	获取对象类名属性的类名文本
String	<code>getContentDescription()</code>	获取对象描述属性的描述文本
String	<code>getPackageName()</code>	获取对象包名属性的包名文本
String	<code>getText()</code>	获取对象文本属性中的文本
Rect	<code>getVisibleBounds()</code>	返回可见视图的范围，如果部分可见，只返回可见范围

# 示例

```
public void testGetProp() throws UiObjectNotFoundException{  
    UiObject chromelcon=new UiObject(new UiSelector().description("Chrome"));  
int rectX=chromelcon.getBounds().left;  
int childCount=chromelcon.getChildCount();  
    String className=chromelcon.getClassName();  
    String packageName=chromelcon.getPackageName();  
    String description=chromelcon.getContentDescription();  
    System.out.println("Rect.x:"+rectX+" childCount:"+childCount+" className:"  
+className+" packageName:"+packageName+" description:"+description);  
}
```



## 2.获取父类与子类节点API

返回值	方法名	描述
UiObject	getChild( <u>selector</u> )	获取对象的子类对象，可以递归获取子孙当中某个对象
UiObject	getFromParent( <u>selector</u> )	从父类获取子类，按照UiSelector获取兄弟类（递归）

# 示例

```
public void testNode() throws UiObjectNotFoundException{  
//re文件管理器  
UiObject dir_d=new UiObject(new UiSelector()  
.className("android.widget.RelativeLayout").index(4));  
UiObject letter_d=dir_d.getChild(new UiSelector().text("d"));  
letter_d.click();  
sleep(1500);  
UiDevice.getInstance().pressBack();  
sleep(500);  
UiObject dir_data=dir_d.getFromParent(new UiSelector()  
.className("android.widget.RelativeLayout").index(5));  
dir_data.click();  
}
```

### 3.属性判断API

返回值	方法名	描述
boolean	isCheckable()	检查对象checkable属性是否为true
boolean	isChecked()	检查对象checked属性是否为true
boolean	isClickable()	检查对象clickable属性是否为true
boolean	isEnabled()	检查对象enabled属性是否为true
boolean	isFocusable()	检查对象focusable属性是否为true
boolean	isFocused()	检查对象focused属性是否为true
boolean	isLongClickable()	检查对象longclickable属性是否为true
boolean	isScrollable()	检查对象scrollable属性是否为true
boolean	isSelected()	检查对象selected属性是否为true

# 示例

```
public void testIs() throws UiObjectNotFoundException{  
//文件管理器  
UiObject dirDataCheckbox=new UiObject(new UiSelector()  
.className("android.widget.CheckBox").instance(5));  
if(!dirDataCheckbox.isChecked()){  
dirDataCheckbox.click();  
}  
}
```

# 手势操作

## 手势相关操作

- 两指手势
- 多指手势
- 两指合拢
- 两指张开

# 手势操作相关API

返回值	方法名	描述
boolean	<code>performMultiPointerGesture(<u>touches</u>)</code>	执行单手指触控手势，可定义任意手势与形状
boolean	<code>performTwoPointerGesture(<u>startPoint1</u>, <u>startPoint2</u>, <u>endPoint1</u>, <u>endPoint2</u>, <u>steps</u>)</code>	执行任意两个手指触控手势，模拟两个手指手势
boolean	<code>pinchIn(<u>percent</u>, <u>steps</u>)</code>	手势操作，两点向内收缩
boolean	<code>pinchOut(<u>percent</u>, <u>steps</u>)</code>	手势操作，两点向外张开

# 示例

```
public void testGesture() throws UiObjectNotFoundException{  
//百度地图  
UiObject picSec=new UiObject(new  
UiSelector().className("android.webkit.WebView"));  
picSec.pinchIn(80, 20);  
sleep(2800);  
picSec.pinchOut(80, 20);  
sleep(2800);  
Point startPoint1, startPoint2, endPoint1, endPoint2;  
startPoint1=new Point(259,788);  
startPoint2=new Point(276,898);  
endPoint1=new Point(572,711);  
endPoint2=new Point(276,1099);  
picSec.performTwoPointerGesture(startPoint1, startPoint2, endPoint1, endPoint2,50);  
}
```

# 判断对象是否存在

## 相关API

返回值	方法名	描述
boolean	<code>waitForExists(<u>timeout</u>)</code>	等待对象出现
boolean	<code>waitUntilGone(<u>timeout</u>)</code>	等待对象消失
boolean	<code>exists()</code>	检查对象是否存在



# 示例

```
public void testExist() throws UiObjectNotFoundException  
UiObject chromelcon=new UiObject(new UiSelector().description("Chrome"));  
if(chromelcon.exists()){  
chromelcon.clickAndWaitForNewWindow();  
}else{  
UiDevice.getInstance().pressHome();  
chromelcon.clickAndWaitForNewWindow();  
}  
UiObject urlRect=new UiObject(new  
UiSelector().resourceId("com.android.chrome:id/url_bar"));  
String url=urlRect.getText();  
assertEquals("https://m.baidu.com",url);  
}
```

# 实例演示

示例：打开浏览器，输入百度网址

- 步骤

灭屏->亮屏->解锁->单击浏览器->单击网址输入框->输入[www.baidu.com](http://www.baidu.com)->按回车键->旋转屏幕->截图

# 示例

**public void testDemo() throws RemoteException, UiObjectNotFoundException{**

//灭屏->亮屏->解锁->单击浏览器->单击网址输入框->输入www.baidu.com->按回车键->旋转屏幕->截图

**UiDevice.getInstance().sleep();**

**sleep(1500);**

**if(!UiDevice.getInstance().isScreenOn()){**

**UiDevice.getInstance().wakeUp();**

**}**

**sleep(1500);**

//向上滑动解锁

**UiDevice.getInstance().swipe(500, 1100, 500, 160, 20);**

**UiDevice.getInstance().pressHome();**

**sleep(1500);**

**UiObject chromelcon=new UiObject(new UiSelector().description("Chrome"));**

**chromelcon.clickAndWaitForNewWindow();**

**sleep(1500);**

**UiObject urlRect=new UiObject(new UiSelector().resourceId("com.android.chrome:id/url\_bar"));**

**urlRect.setText("www.baidu.com");**

**sleep(500);**

**UiDevice.getInstance().pressEnter();**

**sleep(2000);**

**UiDevice.getInstance().setOrientationLeft();**

**sleep(1500);**

**UiDevice.getInstance().takeScreenshot(new File("/sdcard/3.png"));**

**}**

Q&A