TRAINING GUIDE FOR USING UIAUTOMATOR/WEDITOR IN AUTOMATION

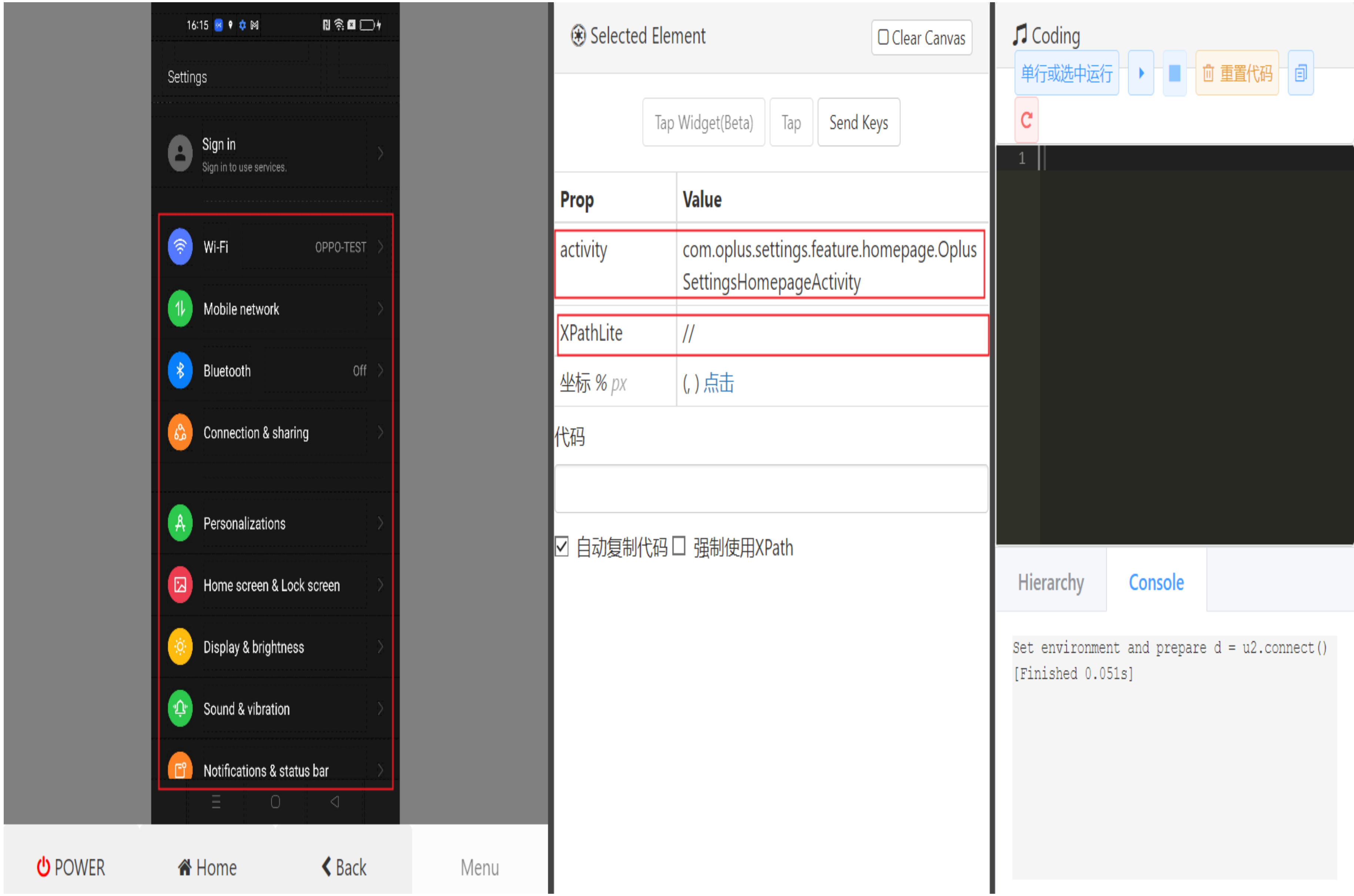
PREPARED BY:SHUBHAM PAL SINGH,ENGINEER 2

INDEX

1)What is UIAutomator?
2)How we are using UIAutomator for writing scripts.
3)What all functions can be used in UIAutomator?
4)How we are importing modules and the functions through olib(By example through script).

What is UIAutomator?

- UI Automator is a UI testing framework suitable for cross-app functional UI testing across system and installed apps.
- UI Automator API lets us interact with visible elements in device and helps in opening setting menu, launch an app atc.



Prop	Value
activity	com.oplus.settings.feature.homepage.OplusSettingsHomepageActivity
XPathLite	//
坐标 % *px*	(,) 点击

代码

☑ 自动复制代码 ☐ 强制使用XPath

```
Set environment and prepare d = u2.connect()
[Finished 0.051s]
```

UIAutomator elements

The following are the UIAutomator elements:

1) activity

2) XPathLite

3) className

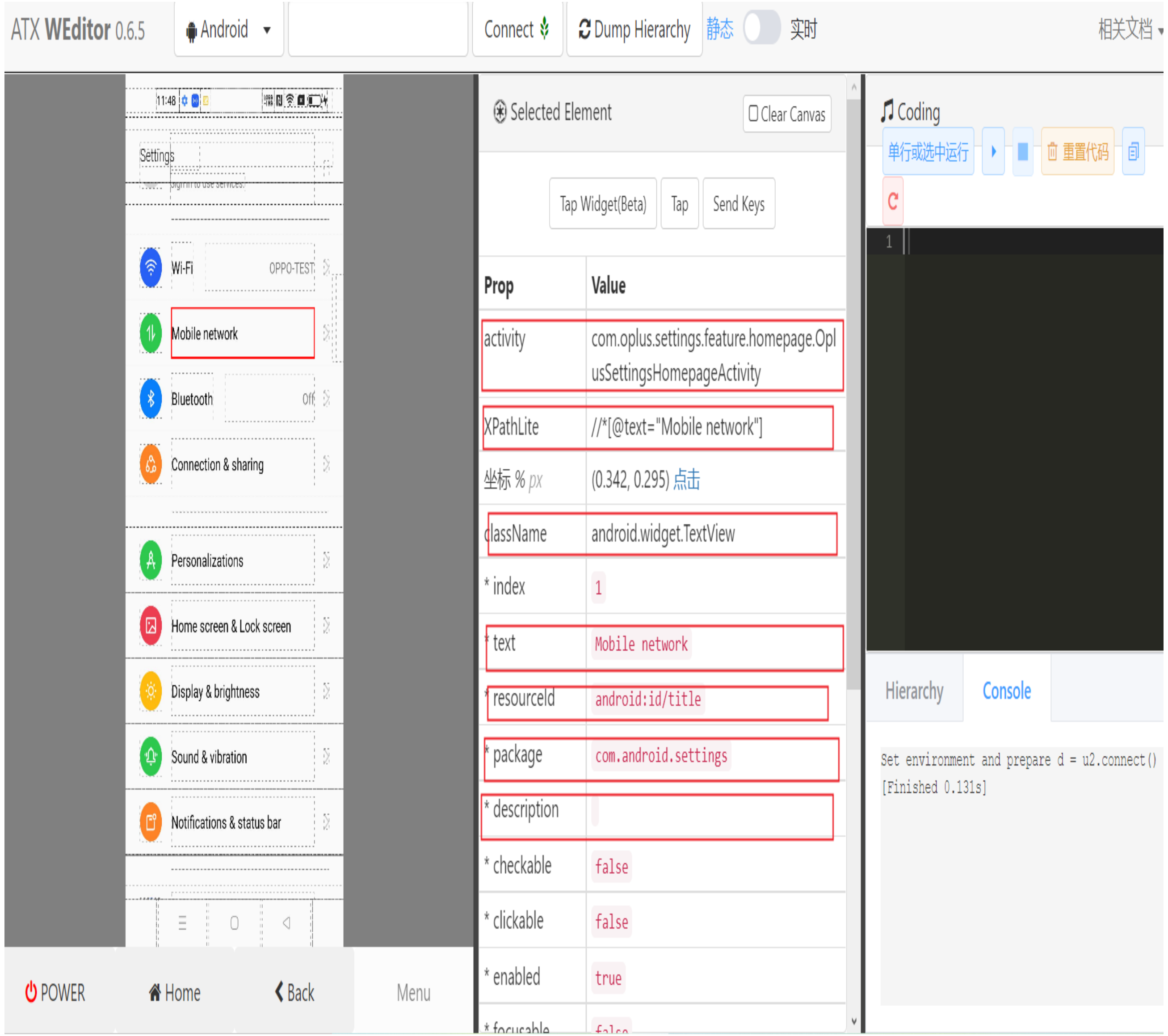
4) text

5) resourceId

6) package

7) description

Using the above elements we can use olib libraries for writing scripts.



Prop	Value
activity	com.oplus.settings.feature.homepage.OplusSettingsHomepageActivity
XPathLite	//*[@text="Mobile network"]
坐标 % *px*	(0.342, 0.295) 点击
className	android.widget.TextView
* index	1
* text	Mobile network
* resourceId	android:id/title
* package	com.android.settings
* description	
* checkable	false
* clickable	false
* enabled	true
* focusable	false

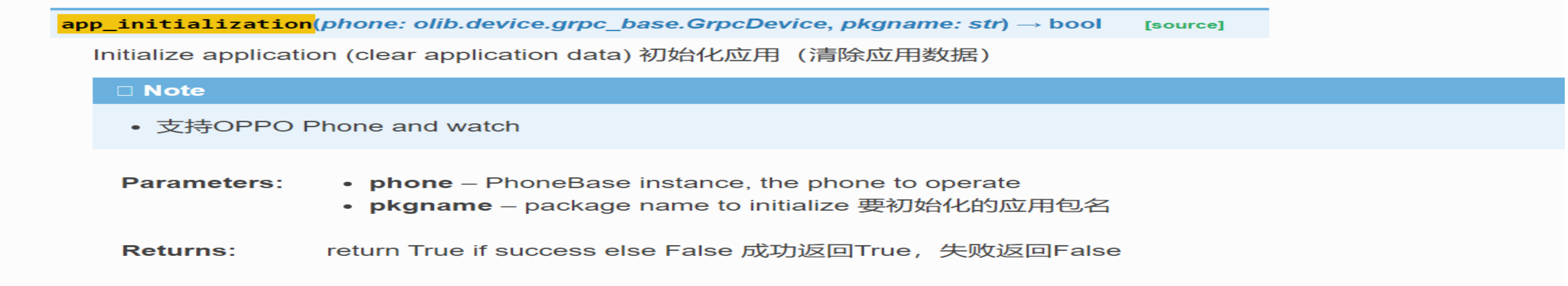
```
Set environment and prepare d = u2.connect()
[Finished 0.131s]
```

Applications of elements using olib in scripts

1)Activity element can be used to start any activity of our device, for example,

- common.app_initialization(dut, “com.coloros.relax”) , we can initialize app

-common.start_app_activity(dut, “com.coloros.ui.MainActivity”, action = None), we can enter into orelax app



app_initialization(*phone: olib.device.grpc_base.GrpcDevice, pkgname: str*) → bool [source]

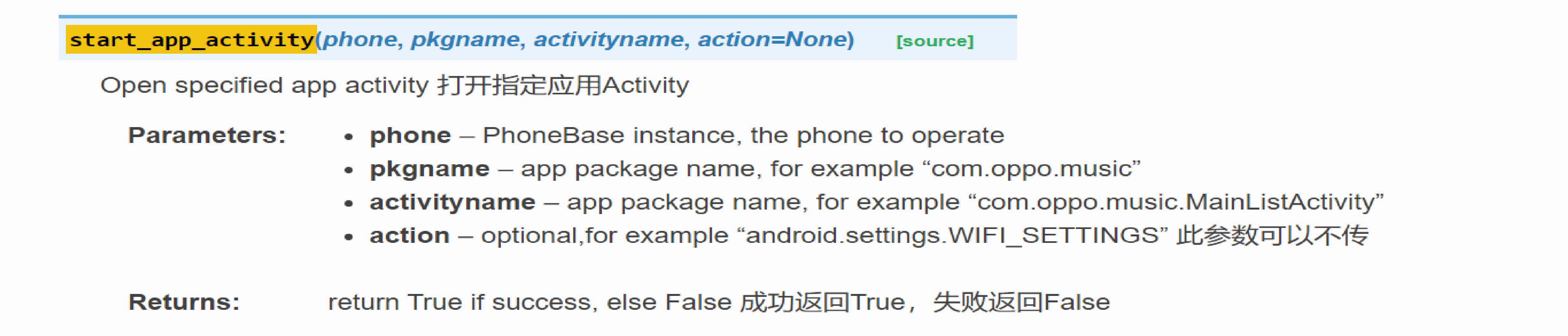
Initialize application (clear application data) 初始化应用 （清除应用数据）

□ **Note**

- 支持OPPO Phone and watch

Parameters:
- **phone** – PhoneBase instance, the phone to operate
- **pkgname** – package name to initialize 要初始化的应用包名

Returns: return True if success else False 成功返回True， 失败返回False



start_app_activity(*phone, pkgname, activityname, action=None*) [source]

Open specified app activity 打开指定应用Activity

Parameters:
- **phone** – PhoneBase instance, the phone to operate
- **pkgname** – app package name, for example “com.oppo.music”
- **activityname** – app package name, for example “com.oppo.music.MainListActivity”
- **action** – optional,for example “android.settings.WIFI_SETTINGS” 此参数可以不传

Returns: return True if success, else False 成功返回True， 失败返回False

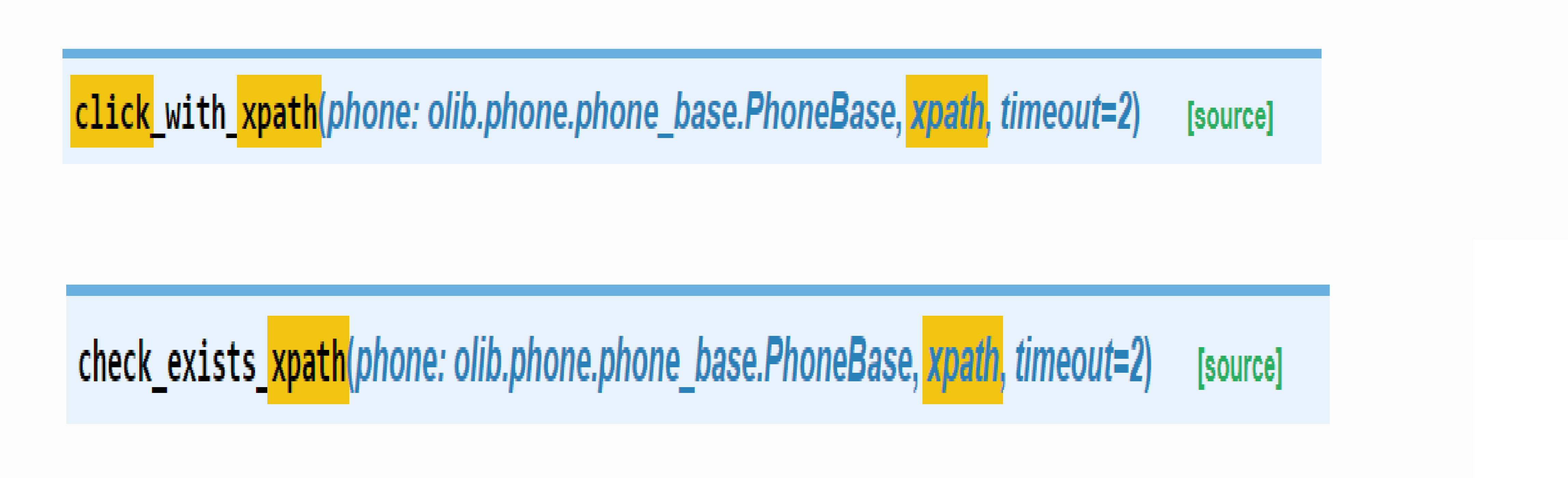
Applications of elements using olib in scripts

2)Xpath-we can use xpath to reach to defined location or path
Xpath can be declared in two ways:

1)Relative xpath- used with // ,it doesnot identify element from root node Eg- (//*[@attribute='value’])

2)Absolute xpath- used with /, it identify element from beginning to root node Eg-/html/body/div[1]/div/div[1]/a.

Which xpath is generally used?

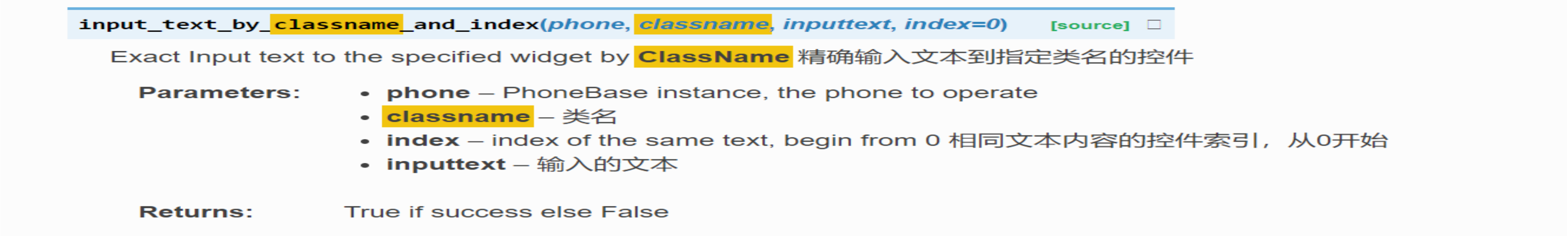


```
click_with_xpath(phone: olib.phone.phone_base.PhoneBase, xpath, timeout=2)    [source]
```

```
check_exists_xpath(phone: olib.phone.phone_base.PhoneBase, xpath, timeout=2)    [source]
```

Application of elements using olib:

We can pick up elements using classname,text, resource id and description.



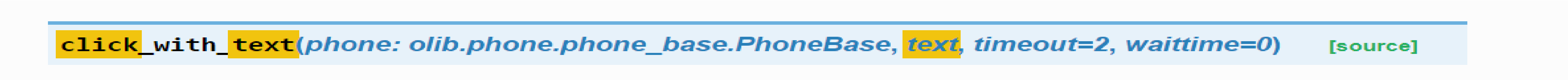
input_text_by_classname_and_index(*phone, classname, inputtext, index=0*) [source] □

Exact Input text to the specified widget by **ClassName** 精确输入文本到指定类名的控件

Parameters:
- **phone** – PhoneBase instance, the phone to operate
- **classname** – 类名
- **index** – index of the same text, begin from 0 相同文本内容的控件索引， 从0开始
- **inputtext** – 输入的文本

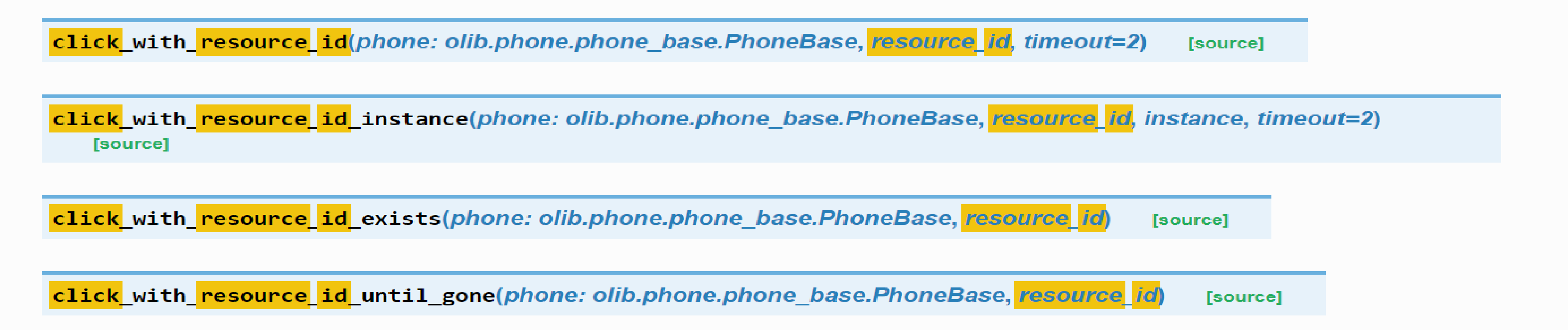
Returns: True if success else False

Click with text:



click_with_text(*phone: olib.phone.phone_base.PhoneBase, text, timeout=2, waittime=0*) [source]

Click with resource_id:



click_with_resource_id(*phone: olib.phone.phone_base.PhoneBase, resource_id, timeout=2*) [source]

click_with_resource_id_instance(*phone: olib.phone.phone_base.PhoneBase, resource_id, instance, timeout=2*) [source]

click_with_resource_id_exists(*phone: olib.phone.phone_base.PhoneBase, resource_id*) [source]

click_with_resource_id_until_gone(*phone: olib.phone.phone_base.PhoneBase, resource_id*) [source]

Thank you

Any questions?