appiumphone

Introduction

appiumphone is a appium wrapper, it follows the idea of warping selenium into seleniumpc. appiumphone's advantage is that it is easy to learn, appiumphone's disadvantage is that by now it only support native-view but not web-view.

Develop & Test Environment

system: Windows 7 a64
runtime: Python 2.7 i386
executor: Appium 1.6.3
phone: Android 4.1

Classes in selenium

Phone

Phone represents the instance of a phone.

Element

Element represents a native-view element.

Android

Element attribute collection belonging to a Android element.

Ios

Element attribute collection belonging to a Ios element.

Net_Android

Android network type collection.

Properties & Definitions in Phone

Phone.identity

Android's device-name/Ios's UDID. Must be a string.

Phone.platform

Must be a string, 'android/ios'.

Phone.version

The phone's system version. Must be a string

Phone.app

The app's absolute path. Must be a string.

Phone.install

When set True, the app would be installed/re-installed. Must be a bool.
This property is optional.

Phone.package

Android's app-package/Ios's BundleID. Must be a string.

Phone.activity

Android's app-activity. Must be a string.
This property is only required by Android.

Phone.executor
Appium's url address. Must be a string.

Phone.log
The absolute path of where you would like to save the log. Must be a string.

Driver.delay
The global delay to slow every step down during your test run. Must be an int above 0.
The unit is millisecond, the default value is 700.

Phone.attach()
Connect the phone to Appium.

Phone.detach()
Disconnect the phone from Appium.

Phone.applaunch()
Launch the app.

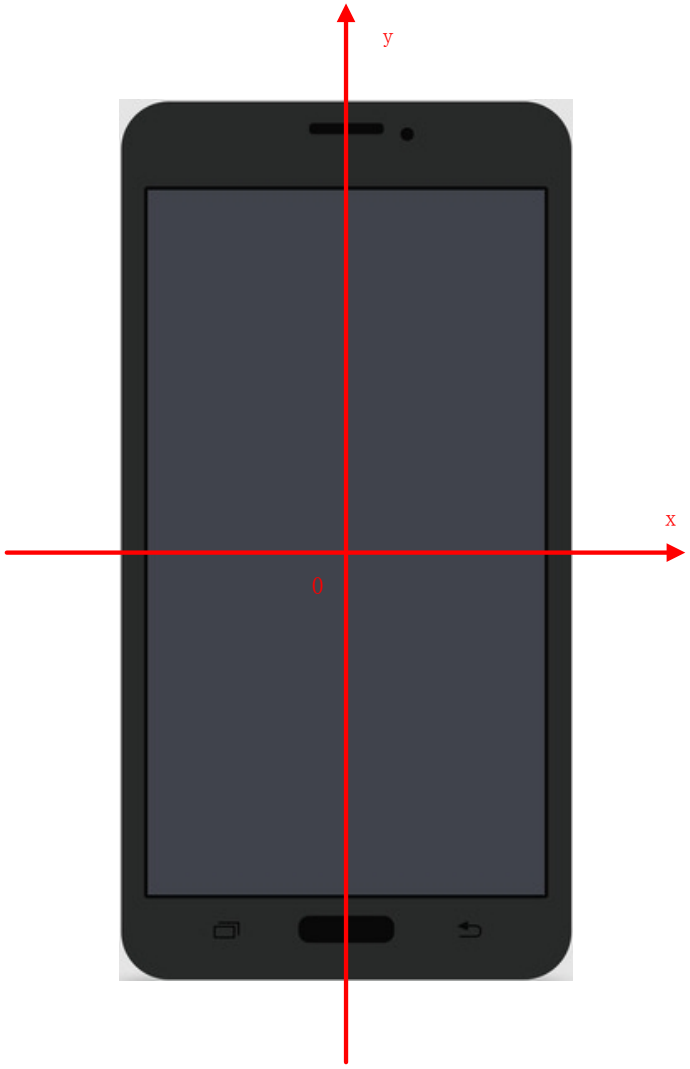
Phone.appclose()
Close the app.

Phone.appreset()
Reset the app.

Phone.apphide(duration)
Put the app into background for a period of time.
duration:
How long you would like to hide the app in the background. Must be an int above 0.
The unit is millisecond.

Driver.find(attribute)
Find all the elements who match the attributes you pass.
Return a list of found Element instances.
attribute:
The attributes that are sufficient enough to identify your desired element. Must be an Android/Ios instance, or a list of Android/Ios instances.

Phone.tap(x, y, count = 1)
Tap on the very coordinates.
x/y:
The coordinates you would like to tap on. Must be an int.
The unit is pixel.
count:
How many times of tap you would like to perform. Must be an int above 0.
The default value is 1.



phone screen's origin point & coordinate system

Phone.hold(x, y)

Press down without release on the very coordinates.

x/y:

The coordinates you would like to press on. Must be an int.

The unit is pixel.

Phone.release(x, y)

Move to the very coordinates as the finger keeps pressing down, and then release.

x/y:

The coordinates you would like to press on. Must be an int.

The unit is pixel.

Phone.press(duration, x, y)

Press down on the very coordinates for a period of time.

duration:

How long you would like to keep pressing down. Must be an int above 0.

The unit is millisecond.

x/y:

The coordinates you would like to press on. Must be an int.

The unit is pixel.

Phone.shake(count = 1)
Shake the phone.
count:
How many times of shake you would like to perform. Must be an int above 0.
The default value is 1.

Phone.locate(latitude, longitude, altitude)
Set the phone's latitude, longitude & altitude.
latitude:
Must be an int/a float, >= -90 && <= 90.
longitude:
Must be an int/a float, >= -180 && <= 180.
altitude:
Must be an int/a float.

Phone.width()
Return the width of the phone screen.

Phone.height()
Return the height of the phone screen.

Phone.topbottom()
Flick from top edge to bottom edge.

Phone.bottomtop()
Flick from bottom edge to top edge.

Phone.leftright()
Flick from left edge to right edge.

Phone.rightleft()
Flick from right edge to left edge.

Phone.imefold()
Fold the keyboard.

Phone.shoot(path)
Take a screenshot of the phone screen.
path:
The absolute path of where you would like to save the screenshot, must be a string.

Phone.type_Android(key, meta = None)
Type key-event.
This definition is Android only.
Key/meta:
key-event: https://developer.android.com/reference/android/view/KeyEvent.html
meta's default value is None, means not sending key-combine.

Phone.toast_Android(toast, strict = True, timeout = 7000)
Return True if the desired toast message were being displayed, otherwise return False.
This definition is Android only.
toast:
The toast message or partial toast message you would like to verify.
strict:
If strict were True, the desired toast message should be strictly the same as you pass. Or the toast message should be contained in that you pass. Must be a str.
The default value is True.
timeout:
The timeout of verifying the toast message. Must be an int about 0.
The unit is millisecond, the default value is 7000.

Phone.net_Android(net)
Set the phone's network type.
This definition is Android only.
net:
Must be an Net_Android instance.

push_Android(self, local, remote)

Phone.push_Android(local, remote)
Push a file into the phone.
This definition is Android only.
local:
The absolute path to the file that you would like to push, must be a string.
remote:
The absolute path in the phone where you would like to save the pushed file, must be a string.

Phone.pull_Android(local, remote)
Pull a file/folder out of the phone.
This definition is Android only.
local:
The absolute path where you would like to save the pulled file/folder, must be a string.
remote:
The absolute path in the phone to the file that you would like to pull, must be a string.

Definitions in Element
Element.find(attribute)
Under the current element, find all the elements who match the attributes you pass.
Return a list of found Element instances.
attribute:
The attributes that are sufficient enough to identify your desired element. Must be an Android/Ios instance, or a list of Android/Ios instances.

Element.parent()
Return an Element instance of the parent node of the current element.

Element.tap(x = 0, y = 0, count = 1)

Tap on the current element with offsets.

x/y:

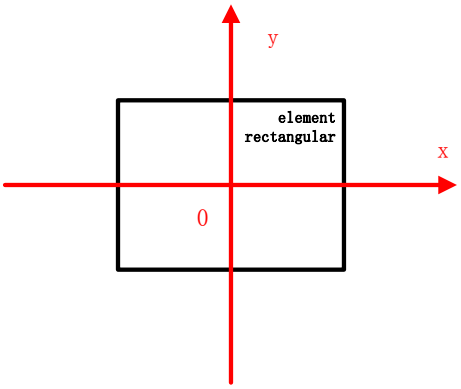
The offsets to the centre of the current element. Must be an int.

The unit is pixel, the default value is 0.

count:

How many times of tap you would like to perform. Must be an int above 0.

The default value is 1.



element's origin point & coordinate system

Element.hold(x = 0, y = 0)

Press down without release on the current element with offsets.

x/y:

The offsets to the centre of the current element. Must be an int.

The unit is pixel, the default value is 0.

Element.release(x = 0, y = 0)

Move to the current element with offsets as the finger keeps pressing down, and then release.

x/y:

The offsets to the centre of the current element. Must be an int.

The unit is pixel, the default value is 0.

Element.press(duration, x = 0, y = 0)

Press down on the current element with offsets for a period of time.

duration:

How long you would like to keep pressing down. Must be an int above 0.

The unit is millisecond.

x/y:

The offsets to the centre of the current element. Must be an int.

The unit is pixel, the default value is 0.

Element.clear()

Clear the content of the current element.

Element.send(send)

Send content into the current element.

send:

The content you would like to send into the current element. Must be a str.

Element.waitexist(attribute, timeout = 7000)
Under the current element, wait until the very element could be found.
attribute:
The attributes that are sufficient enough to identify your desired element. Must be an Android/Ios instance, or a list of Android/Ios instances.
timeout:
The timeout to wait for the element's appearing. Must be an int above 0.
The unit is millisecond, the default value is 7000.

Element.waitextinct(attribute, timeout = 7000)
Under the current element, wait until the very element could not be found.
attribute:
The attributes that are sufficient enough to identify your desired element. Must be an Android/Ios instance, or a list of Android/Ios instances.
timeout:
The timeout to wait for the element's disappearing. Must be an int above 0.
The unit is millisecond, the default value is 7000.

Element.width()
Return the width of the current element.

Element.height()
Return the height of the current element.

Element.abscissa()
Return the abscissa of the current element to the centre of the phone's screen.

Element.ordinate()
Return the ordinate of the current element to the centre of the phone's screen.

Element.isdisplay()
Return True if the current element were being displayed, otherwise return False.

Element.isselect()
Return True if the current element were being selected, otherwise return False.

Element.isenable()
Return True if the current element were enabled, otherwise return False.

Element. shoot(path, left = 0, top = 0, right = 0, bottom = 0)
Take a screenshot of the current DOM.
path:
The absolute path of where you would like to save the screenshot, must be a string.
left/top/right/bottom:
left/top/right/bottom is the offset to the edge of the current DOM, must be an int.
The unit is pixel, the default value is 0.

Definition in Android

Android.class_(class_)

Return an Android instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

class_:

The class-value of your desired Android element's attribute. Must be a str.

Android.text_(text_, strict = True)

Return an Android instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

text_:

The text-value of your desired Android element's attribute. Must be a str.

strict:

If strict were True, the desired element should have strictly the same text-value as you pass. Or the very element's text-value should be contained in that you pass. Must be a str.

The default value is True.

Android.contentdesc_(contentdesc_, strict = True)

Return an Android instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

contentdesc_:

The content-desc-value of your desired Android element's attribute. Must be a str.

strict:

If strict were True, the desired element should have strictly the same content-desc-value as you pass. Or the very element's content-desc-value should be contained in that you pass. Must be a str.

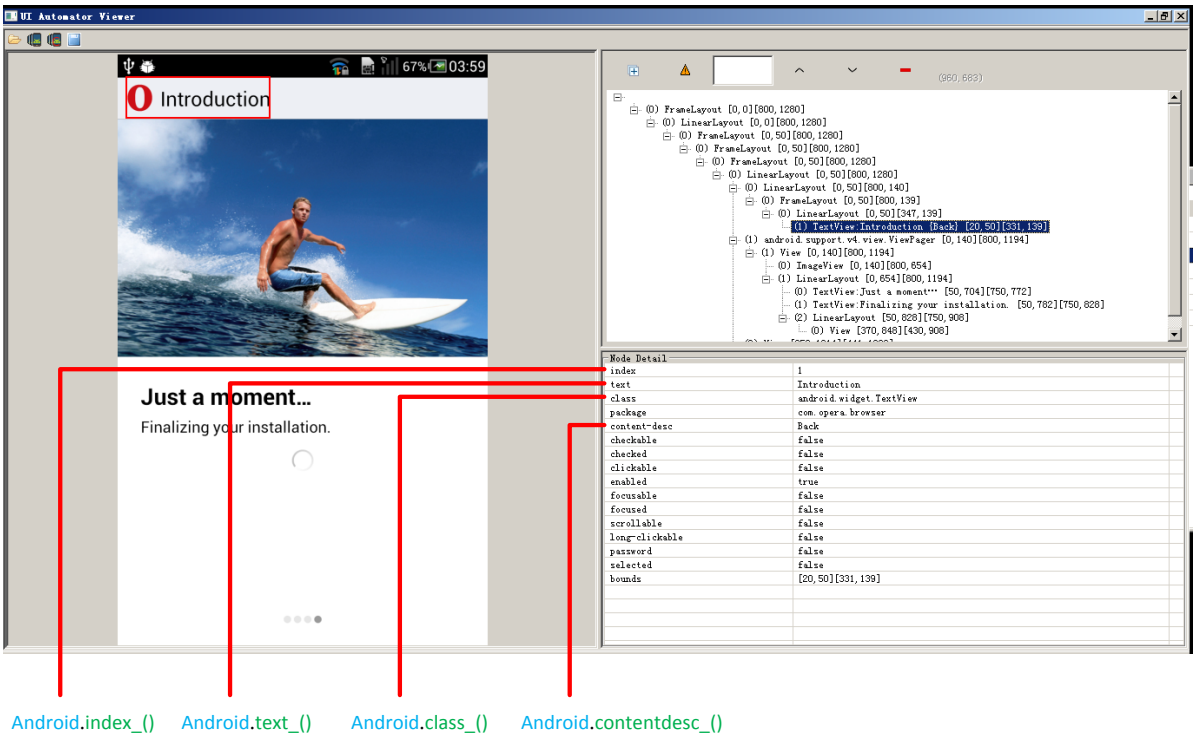
The default value is True.

Android.index_(index_)

Return an Android instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

index_:

The index-value of your desired Android element's attribute. Must be a int >= 0.



Android's class, text, content-desc & index

Definition in Ios

Ios.type_(type_)

Return an Ios instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

type_:

The type-value of your desired Ios element's attribute. Must be a str.

Ios.name_(name_, strict = True)

Return an Ios instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

name_:

The name-value of your desired Ios element's attribute. Must be a str.

strict:

If strict were True, the desired element should have strictly the same name-value as you pass. Or the very element's name-value should be contained in that you pass. Must be a str.

The default value is True.

Ios.label_(label_, strict = True)

Return an Ios instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

label_:

The label-value of your desired Ios element's attribute. Must be a str.

strict:

If strict were True, the desired element should have strictly the same label-value as you pass. Or the very element's label-value should be contained in that you pass. Must be a str.

The default value is True.

Ios.value_(value_, strict = True)

Return an Ios instance that could be passed to Phone.find()/Element.find()/Element.waitexist()/Element.waitextinct() as an attribute pattern.

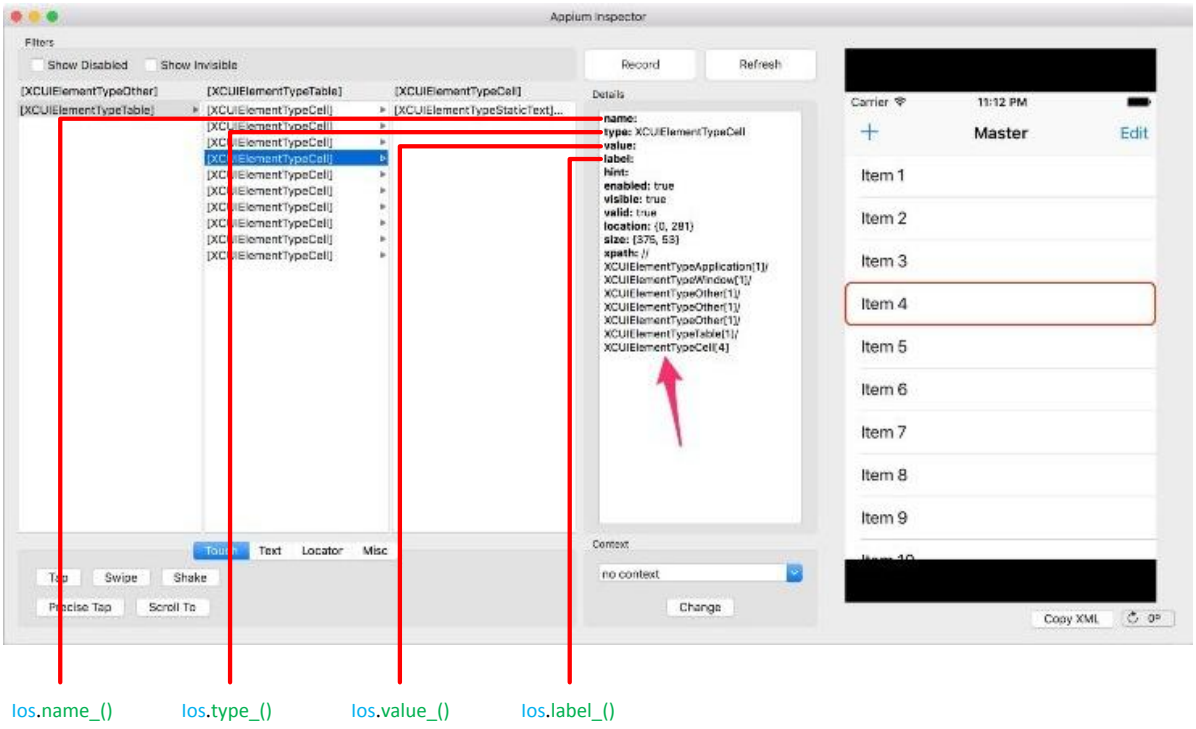
value_:

The value-value of your desired Ios element's attribute. Must be a str.

strict:

If strict were True, the desired element should have strictly the same value-value as you pass. Or the very element's value-value should be contained in that you pass. Must be a str.

The default value is True.



Ios.name_() Ios.type_() Ios.value_() Ios.label_()

Ios's type, name, label & value