**Locators in Appium**

**Different Types of Locators in Appium**

|  |  |  |
| --- | --- | --- |
| **Locator Strategy** | **Example** | **Code** |
| Accessibility ID | unique\_element\_name | driver.findElement(AppiumBy.accessibilityId("unique\_element\_name")); |
| ID | app\_package/resource-id | driver.findElement(AppiumBy.id("resource\_id")); |
| Class Name | android.widget.TextView | driver.findElement(AppiumBy.className("android.widget.TextView")); |
| XPath | //UiAutomator2\_class\_name[@attribute\_name = "attribute\_value"]/axes::expressions | driver.findElement(AppiumBy.xpath("//UiAutomator2\_class\_name[@attribute\_name = \"attribute\_value\"]/axes::expressions")); |
| UiAutomator2 (UiSelector) | new UiSelector().text(\"Animation\") | driver.findElement(AppiumBy.androidUIAutomator("new UiSelector().text(\"Animation\")")); |
| Image | base64\_encoded\_string | driver.findElement(AppiumBy.image("base64\_encoded\_string")); |

* Name & tagname Locators doesn’t work with Android, so if we try to use that then we will get Errors & Exception

**How to Inspect the Locators**

* In Appium inspector click on the Search button 🡪 Select desired Locator 🡪 Enter the Locator Data & click on ok

**Program for Different Locator Usage**

package Project\_2\_Locators\_Usage;

import java.net.MalformedURLException;

import java.net.URL;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.remote.DesiredCapabilities;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

import io.appium.java\_client.android.AndroidDriver;

import io.appium.java\_client.remote.MobileCapabilityType;

public class Program\_1\_Diff\_Locator\_Usage {

@Test

public void locatorUsageTest() throws Throwable

{

DesiredCapabilities dc=new DesiredCapabilities();

dc.setCapability(MobileCapabilityType.PLATFORM\_NAME, "Android");

dc.setCapability(MobileCapabilityType.DEVICE\_NAME, "Galaxy M30s");

dc.setCapability(MobileCapabilityType.AUTOMATION\_NAME, "UiAutomator2");

dc.setCapability(MobileCapabilityType.UDID, "RZ8M83ZJH2W");

dc.setCapability("appPackage", "io.appium.android.apis");

dc.setCapability("appActivity", ".ApiDemos");

URL u=new URL("http://localhost:4723");

AndroidDriver driver=new AndroidDriver(u,dc);

driver.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS);

WebElement ele = driver.findElement(AppiumBy.accessibilityId("Accessibility"));

System.out.println(ele.getText());

ele=driver.findElements(AppiumBy.id("android:id/text1")).get(1);

System.out.println(ele.getText());

ele = driver.findElements(AppiumBy.className("android.widget.TextView")).get(2);

System.out.println(ele.getText());

ele=driver.findElement(AppiumBy.xpath("//android.widget.TextView[@content-desc='Accessibility']"));

System.out.println(ele.getText());

driver.findElements(AppiumBy.xpath("//android.widget.TextView[@text='Accessibility']"));

System.out.println(ele.getText());

ele = driver.findElement(AppiumBy.tagName("Accessibility"));

System.out.println(ele.getText());

}

}

**Locating Element Using UiAutomator(UiSelecter Class)**

**package** Project\_2\_Locators\_Usage;

**import** java.net.MalformedURLException;

**import** java.net.URL;

**import** java.util.concurrent.TimeUnit;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.remote.DesiredCapabilities;

**import** org.testng.annotations.Test;

**import** io.appium.java\_client.AppiumBy;

**import** io.appium.java\_client.android.AndroidDriver;

**import** io.appium.java\_client.remote.MobileCapabilityType;

**public** **class** Program\_2\_Locating\_Element\_Using\_UiAutomator\_Test {

@Test

**public** **void** locatingUsingUiAutomator() **throws** Throwable

{

DesiredCapabilities dc=**new** DesiredCapabilities();

dc.setCapability(MobileCapabilityType.***PLATFORM\_NAME***, "Android");

dc.setCapability(MobileCapabilityType.***DEVICE\_NAME***, "Galaxy M30s");

dc.setCapability(MobileCapabilityType.***AUTOMATION\_NAME***, "UiAutomator2");

dc.setCapability(MobileCapabilityType.***UDID***, "RZ8M83ZJH2W");

dc.setCapability("appPackage", "io.appium.android.apis");

dc.setCapability("appActivity", ".ApiDemos");

URL u=**new** URL("http://localhost:4723");

AndroidDriver driver=**new** AndroidDriver(u,dc);

driver.manage().timeouts().~~implicitlyWait~~(20, TimeUnit.***SECONDS***);

WebElement myElement = driver

.findElement(AppiumBy.*androidUIAutomator*("new UiSelector().text(\"Accessibility\")"));

System.***out***.println(myElement.getText());

myElement = driver

.findElements(AppiumBy.*androidUIAutomator*("new UiSelector().className(\"android.widget.TextView\")")).get(2);

System.***out***.println(myElement.getText());

myElement = driver

.findElement(AppiumBy.*androidUIAutomator*("new UiSelector().description(\"Accessibility\")"));

System.***out***.println(myElement.getText());

myElement = driver

.findElements(AppiumBy.*androidUIAutomator*("new UiSelector().resourceId(\"android:id/text1\")")).get(1);

System.***out***.println(myElement.getText());

}

}

**How to start Appium Server Automatically through Program**

For Windows we need this path which will help us to invoke Appium Server

C://Users//SOUMYASANTA SAHOO//AppData//Roaming//npm//node\_modules//appium//build//lib//main.js

We have to add a piece of code in our Program i,e

To start server

File f=**new** File("C://Users//SOUMYASANTA SAHOO//AppData//Roaming//npm//node\_modules//appium//build//lib//main.js");

AppiumDriverLocalService sb=**new** AppiumServiceBuilder()

.withAppiumJS(f)

.withIPAddress("127.0.0.1").usingPort(4723).withTimeout(Duration.*ofSeconds*(300)).build();

sb.start();

To stop server

sb.stop();