For Books/PDF : mail to [rahulonlinetutor@gmail.com](mailto:rahulonlinetutor@gmail.com) see video 31

**Pre-Requisites:**

Make sure your Appium server is stared

Command in Command prompt : Appium

Make sure your Emulator is opened.

Command : Open the Emulator in cmd C:\Users\Nbarnana\AppData\Local\Android\Sdk\emulator

And enter the command emulator -avd emulatorname

My desired Capabilities:

1)Emulator : Open the Emulator

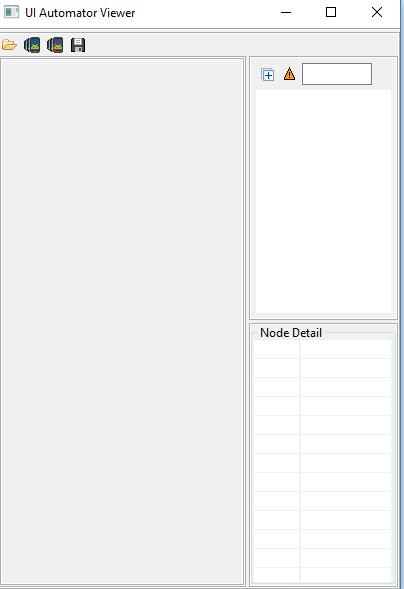
2)Invoke App

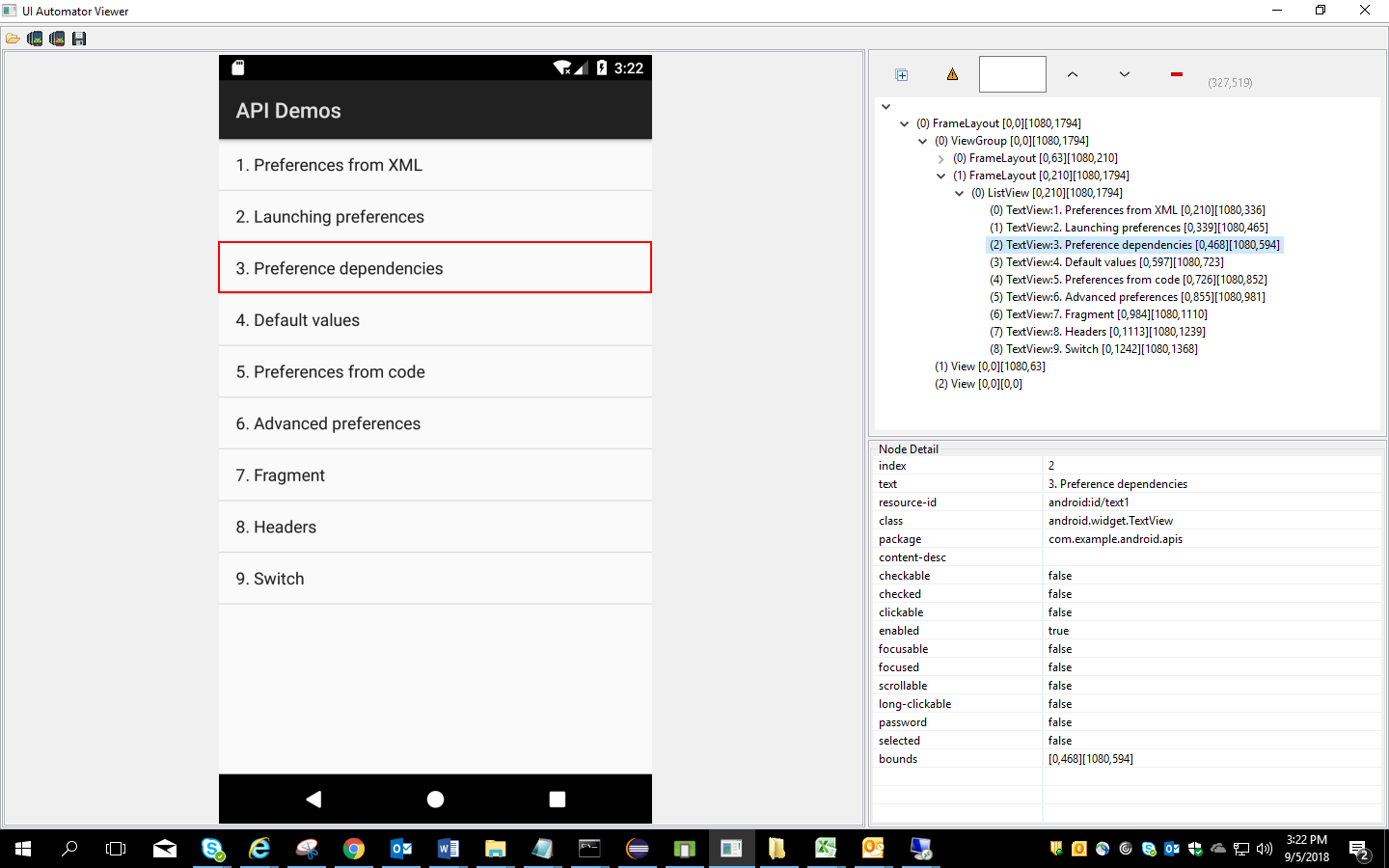
3)Connection step to server

4)Code

Android UI Automator Tool : used to inspect elements. It will be available in below location.

C:\Users\Nbarnana\AppData\Local\Android\Sdk\tools\bin





Open the Emulator first and then invoke the Android UI Automator

Locators used in Selenium:

1)xpath id

2)class name

3)androidUiAutomator

Basic program

package appiumTest;

import java.io.File;

import java.net.MalformedURLException;

import java.net.URL;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.remote.DesiredCapabilities;

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

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

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

public class BaseClass {

public static AndroidDriver<AndroidElement> Capabilities() throws MalformedURLException {

File f=new File("src");

File fs=new File(f,"ApiDemos-debug.apk");

DesiredCapabilities cap= new DesiredCapabilities();

cap.setCapability(MobileCapabilityType.DEVICE\_NAME, "NarayanaEmulator1");

cap.setCapability(MobileCapabilityType.APP, fs.getAbsolutePath());

//Connection to the server

//AndroidDriver driver=new AndroidDriver(conection to the server link, capability object);

//We are working on Android server so we are selection AndroidDriver

//http://127.0.0.1: is common for both android and ios server and 4723 is appium server which you are working

AndroidDriver<AndroidElement> driver=new AndroidDriver<>(new URL("http://127.0.0.1:4723/wd/hub"),cap);

return driver;

//System.out.println("Program executed sucessfully");

}

}

**package** appiumTest;

**import** java.net.MalformedURLException;

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

**import** org.openqa.selenium.By;

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

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

**public** **class** IdentifyingElements **extends** BaseClass{

**public** **static** **void** main(String[] args) **throws** MalformedURLException {

AndroidDriver<AndroidElement> driver=*Capabilities*();

driver.manage().timeouts().implicitlyWait(10, TimeUnit.***SECONDS***);

//xpath id, class name and androidUiAutomator

driver.findElement(By.*xpath*("//android.widget.TextView[@text='Preference']")).click();

driver.findElement(By.*xpath*("//android.widget.TextView[@text='3. Preference dependencies']")).click();

driver.findElement(By.*id*("android:id/checkbox")).click();

driver.findElement(By.*xpath*("//android.widget.TextView[@text='WiFi settings']")).click();

//driver.findElement(By.id("android:id/edit")).sendKeys("Hello");

driver.findElement(By.*className*("android.widget.EditText")).sendKeys("Hello");

driver.findElement(By.*xpath*("//android.widget.Button[@text='OK']")).click();

}

}

How to Work with Android UI Automator locator :

driver.findElementByAndroidUIAutomator("attribute(“value”)");

//Java doesn’t accept quotes inside the quotes. so we have to use \

driver.findElementByAndroidUIAutomator("text(\"Views\")").click();

/validate all clickable options in Views tab

System.***out***.println(driver.findElementsByAndroidUIAutomator("new UiSelector().clickable(true)").size());

**Gestures in Appium :** To work on Gestures in Appium we use Touch Actions class

Below are the core Gestures in Appium Android and IOS

Tap, Press and release, scroll ,swipe and drag down

**1)Tap and press and release in Appium:**

**package** appiumTest;

**import** java.net.MalformedURLException;

**import** java.time.Duration;

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

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

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

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

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.interactions.touch.TouchActions;

//import io.appium.java\_client.TouchAction;

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

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

**import** io.appium.java\_client.touch.TapOptions;

**import** io.appium.java\_client.touch.WaitOptions;

**import** io.appium.java\_client.touch.offset.ElementOption;

**public** **class** Gestures **extends** BaseClass{

**public** **static** **void** main(String[] args) **throws** MalformedURLException {

AndroidDriver<AndroidElement> driver=*Capabilities*();

driver.findElementByAndroidUIAutomator("text(\"Views\")").click();

//Tapping on element

TouchAction t=**new** TouchAction<>(driver);//.tap(TapOptions.tapOptions().withElement(ElementOption.element(driver.findElementByAndroidUIAutomator("text(\"Expandable Lists\")")))).perform();

t.tap(TapOptions.*tapOptions*().withElement(ElementOption.*element*(driver.findElementByAndroidUIAutomator("text(\"Expandable Lists\")")))).perform();

//Clicking on Custom Adapter

driver.findElement(By.*xpath*("//android.widget.TextView[@text='1. Custom Adapter']")).click();

driver.findElement(By.*xpath*("//android.widget.TextView[@text='People Names']"));

//How to press the element for 5 second

t.press(ElementOption.*element*(driver.findElement(By.*xpath*("//android.widget.TextView[@text='People Names']")))).waitAction(**new** WaitOptions().withDuration(Duration.*ofSeconds*(5))).release().perform();

driver.findElement(By.*xpath*("//android.widget.TextView[@text='Sample action']")).click();

}

}

**Swipe in Appium:**

package appiumTest;

import java.net.MalformedURLException;

import java.time.Duration;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import io.appium.java\_client.TouchAction;

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

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

import io.appium.java\_client.touch.WaitOptions;

import io.appium.java\_client.touch.offset.ElementOption;

public class SwipeGesture extends BaseClass{

public static void main(String[] args) throws MalformedURLException {

AndroidDriver<AndroidElement> driver=Capabilities();

driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

driver.findElementByAndroidUIAutomator("text(\"Views\")").click();

driver.findElement(By.xpath("//android.widget.TextView[@text='Date Widgets']")).click();

//driver.findElementByAndroidUIAutomator("text(\"Date Widgets\")").click();

//xpath will not work if the tag name has special character like $ so we need to use either below or findElementByAndroidUIAutomator

//driver.findElement(By.xpath("\\\*[@text='Date Widgets']")).click();

driver.findElementByAndroidUIAutomator("text(\"2. Inline\")").click();

driver.findElement(By.xpath("//\*[@content-desc='9']")).click();

//Swipe gesture : using moveTo option we can swipe

TouchAction t=new TouchAction<>(driver);

t.press(ElementOption.element(driver.findElement(By.xpath("//\*[@content-desc='15']")))).waitAction(new WaitOptions().withDuration(Duration.ofSeconds(5))).moveTo(ElementOption.element(driver.findElement(By.xpath("//\*[@content-desc='45']")))).release().perform();

System.out.println(driver.findElement(By.id("android:id/time\_header")).getText());

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Scroll Down in Appium:** *By using AndroidUIAutomator*

**package** appiumTest;

**import** java.net.MalformedURLException;

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

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

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

**public** **class** ScrollGesture **extends** BaseClass{

**public** **static** **void** main(String[] args) **throws** MalformedURLException {

AndroidDriver<AndroidElement> driver=*Capabilities*();

driver.manage().timeouts().implicitlyWait(5, TimeUnit.***SECONDS***);

driver.findElementByAndroidUIAutomator("text(\"Views\")").click();

//Appium is

//**driver.findElementByAndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"WebView\"));");**

**driver.findElementByAndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Radio Group\"));");**

System.***out***.println("Scrolled succesfully");

}

}

**Drag and Drop in Appium:**

**package** appiumTest;

**import** java.net.MalformedURLException;

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

**import** org.openqa.selenium.By;

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

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

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

**import** io.appium.java\_client.touch.offset.ElementOption;

**public** **class** DragAndDrop **extends** BaseClass{

**public** **static** **void** main(String[] args) **throws** MalformedURLException {

AndroidDriver<AndroidElement> driver=*Capabilities*();

driver.manage().timeouts().implicitlyWait(5, TimeUnit.***SECONDS***);

driver.findElementByAndroidUIAutomator("text(\"Views\")").click();

driver.findElementByAndroidUIAutomator("text(\"Drag and Drop\")").click();

TouchAction t=**new** TouchAction<>(driver);

//t.longPress(ElementOption.element(driver.findElement(By.className("")))).moveTo(ElementOption.element(driver.findElement(By.id("com.example.android.apis:id/drag\_dot\_3")))).release().perform();

//t.longPress(ElementOption.element(driver.findElementByAndroidUIAutomator("resource-id(\"com.example.android.apis:id/drag\_dot\_1\")"))).moveTo(ElementOption.element(driver.findElementByAndroidUIAutomator("resource-id(\"com.example.android.apis:id/drag\_dot\_1\")"))).release().perform();

t.longPress(ElementOption.*element*(driver.findElement(By.*xpath*("//android.widget.TextView[@resource-id='com.example.android.apis:id/drag\_dot\_1']")))).moveTo(ElementOption.*element*(driver.findElement(By.*xpath*("//android.widget.TextView[@resource-id='com.example.android.apis:id/drag\_dot\_1']")))).release().perform();

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***MiscellaneousConcepts\*\*\*\*\*\*\*\*\*\*

**package** appiumTest;

**import** java.net.MalformedURLException;

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

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

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

**import** io.appium.java\_client.android.~~AndroidKeyCode~~;

**import** io.appium.java\_client.android.nativekey.AndroidKey;

**import** io.appium.java\_client.android.nativekey.KeyEvent;

**import** io.appium.java\_client.android.nativekey.KeyEventFlag;

**public** **class** MiscellaneousConcepts **extends** BaseClass{

@SuppressWarnings("deprecation")

**public** **static** **void** main(String[] args) **throws** MalformedURLException {

AndroidDriver<AndroidElement> driver=*Capabilities*();

driver.manage().timeouts().implicitlyWait(5, TimeUnit.***SECONDS***);

//To know the activity of app i.e each page has an activity

//output : .ApiDemos

System.***out***.println(driver.currentActivity());

//To check the context whether it is native app or hybrid app or Web app

//output : NATIVE\_APP

System.***out***.println(driver.getContext());

//To check the orientation i.e. landscape or portrait

System.***out***.println(driver.getOrientation());

//To check mobile is locked or not

System.***out***.println(driver.isDeviceLocked());

//To hide keyboard

//System.out.println(driver.hideKeyboard());

driver.findElementByAndroidUIAutomator("text(\"Views\")").click();

//To Navigate to back page

//driver.pressKeyCode(AndroidKeyCode.BACK);

//driver.pressKey(new KeyEvent(AndroidKey.BACK));

// driver.pressKey(new KeyEvent(AndroidKey.BACK).

//

// withFlag(KeyEventFlag.SOFT\_KEYBOARD).

//

// withFlag(KeyEventFlag.EDITOR\_ACTION));

}

}

**Invoke App with Package Activity**

Test the app without providing the .apk file. This can be done by using package name and package activity.

Package name :

Package Activity : Activity means for every page there should be an activity page. The first page will be the main launch activity.

Download apk info app from playstore and check the package name and activity for the particular app.