APPIUM:

Webdriver+Mobile Feature:

Appium is an open source, cross-platform test automation tool for native, hybrid and mobile web apps, tested on simulators (iOS), emulators (Android), and real devices (iOS, Android).

Supported Platforms

* iOS
* Android

Why Appium?

1. You don't have to recompile your app or modify it in any
2. way, due to use of standard automation APIs on all platforms.
3. You can use any testing framework.
4. You can write the code in any language Supported by WebDriver
5. If you use Apple's UIAutomation library without Appium you can only write tests using JavaScript and you can only run tests through the Instruments application. Similarly, with Google's UiAutomator you can only write tests in Java. Appium opens up the possibility of true cross-platform native mobile automation.

### Advantages

* Appium Framework is a free and open-source framework.
* It provides cross-platform solutions for native and hybrid apps.
* Programming languages like C#, Java, PHP, Python, Ruby are supported.
* App automation is made possible using Appium mobile testing.
* Support simulators, emulators, and real devices concurrently.
* JSON wire protocol supported.
* Independent of mobile device means it could be used in any mobile device.
* Excellent support for Android versions above 4.1.

Disadvantages

* Doesn’t support image comparison.
* Appium does not support testing of Android Version lower than 4.2.

Prerequesites:

1.Java

2.node

3.Android studio

4.AppiumInspector

I)After download set the environment variables of java with new system variable

ii)After download node set the path create new system variable as home-node

Iii)Apart from that open node.js in program files and open node-modules go to npm-> bin then copy that path and paste in the environment path.

Iv)Copy the Sdk path of endroid studio create one new variable

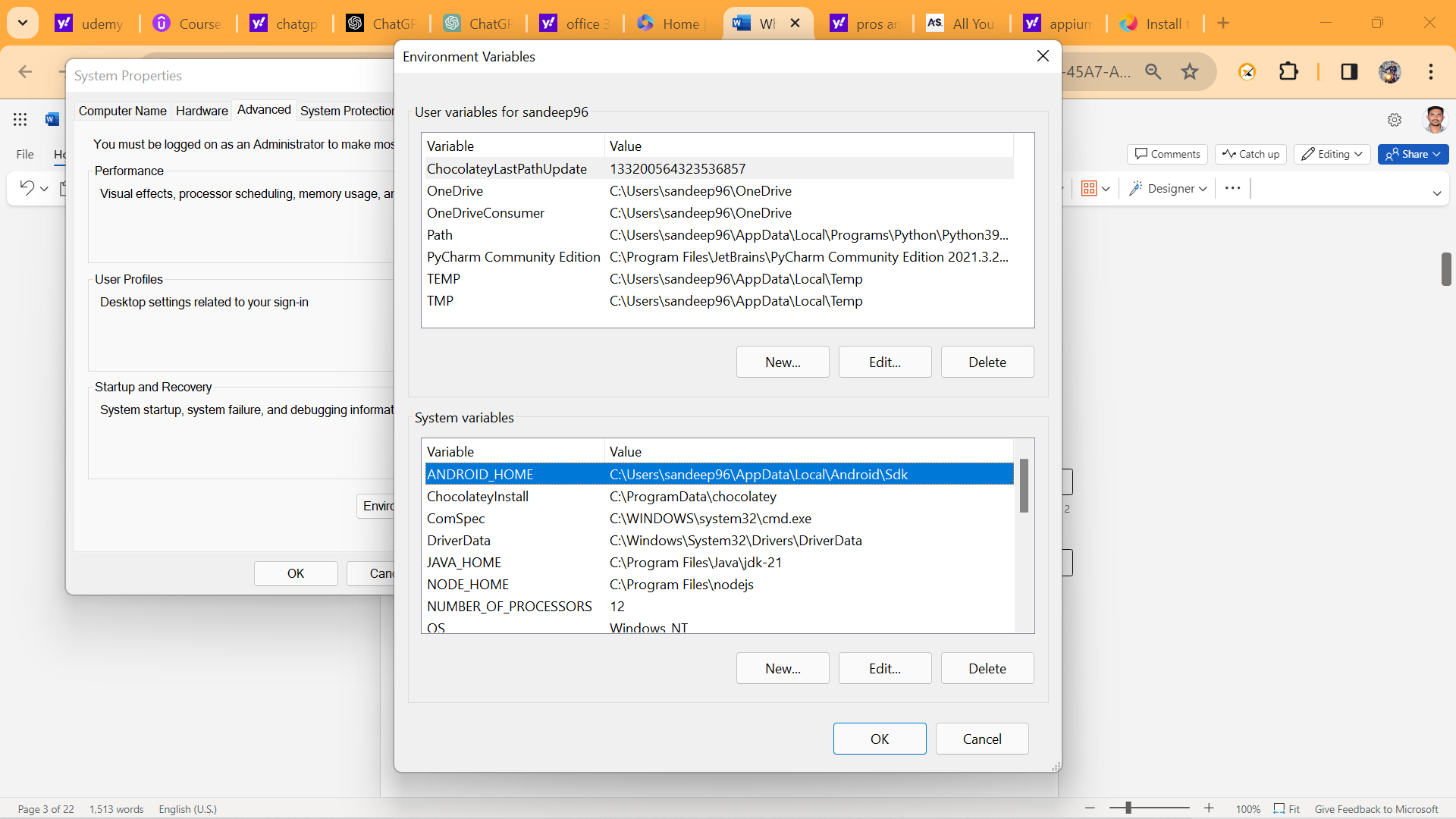
Goto user->uid->Appdata->local->Android->SDK->copy the path

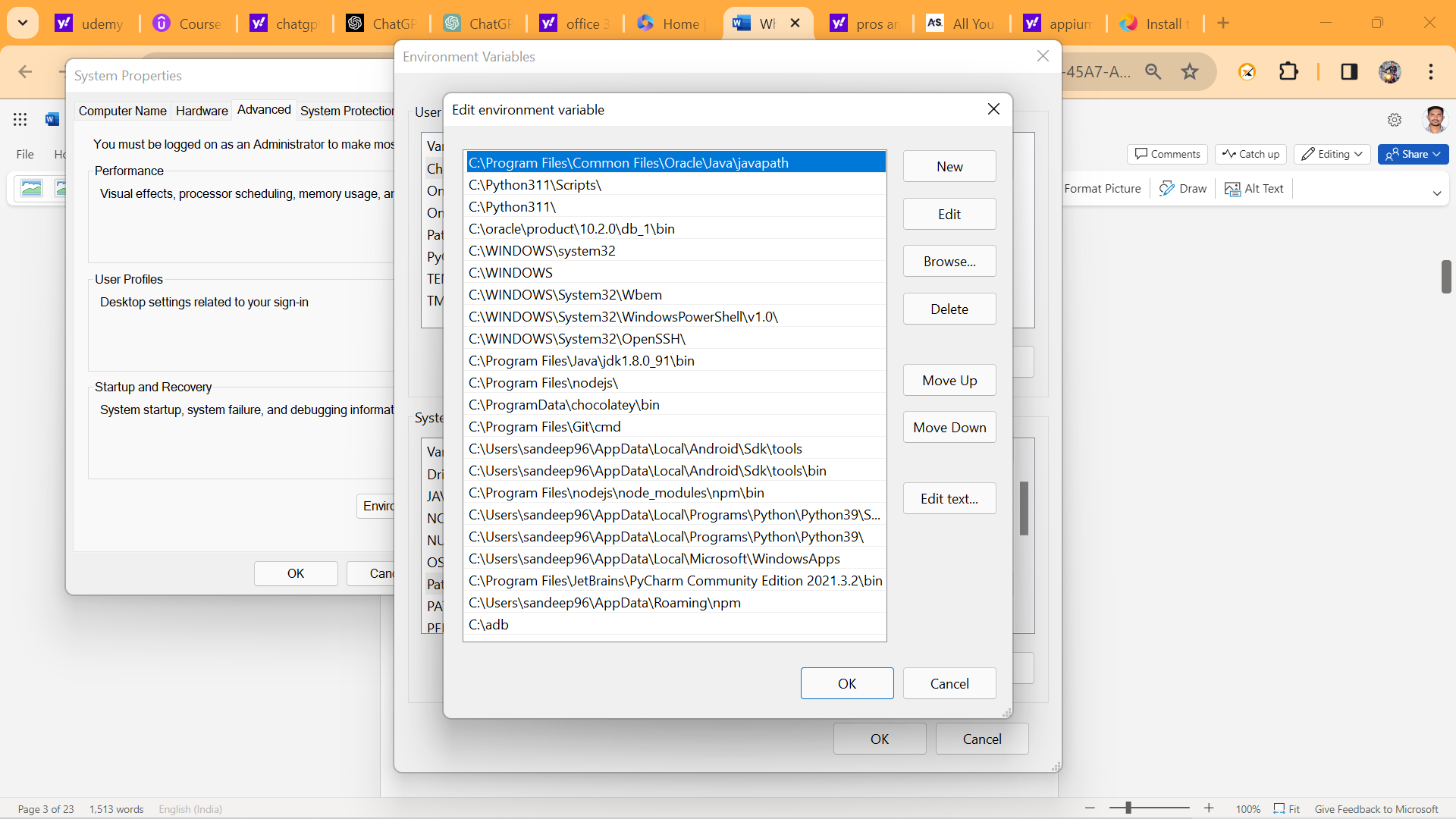
v)c:\user\ss22055\Appdata\local\Android\Sdk\tools\bin....

vi) c:\user\ss22055\Appdata\local\Android\Sdk\tools

vii) c:\user\ss22055\Appdata\local\Android\Sdk\platformtools

Installation process

1. 

We need

1. Android Studio
2. Java
3. Node js

DO NOT SKIP - Important Note on dependencies version compatibility

1.  Node version minimum should be 16. you can check your version with below command

node –v If it is less than 16, please upgrade

2.  This course is taught on latest Appium Server 2.0 version. If you use older Version 1.22, then you will run into issues.

You can check your Appium version with below command appium -v If you have less than 2 version, use below steps to install correctly

npm uninstall -g appium Delete  appium folder in node modules npm install -g appium@next

Starting the appium server we use appium command

1. Appium Java client 8.0.0 has few bugs. So please select the version Greater than 8.0.0 from mavenrepository.com. (Always select latest available)

So, as of now, below is Latest perfect working (Appium & TestNG) Combination which you can have in your POM.xml Project for smoother execution

<dependency>

<groupId>io.appium</groupId>

<artifactId>java-client</artifactId>

<version>9.0.0</version>

</dependency>

<! -- https://mvnrepository.com/artifact/org.testng/testng -->

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>6.14.3</version>

</dependency>

UIAUTOMATOR2

UiAutomator2 is a UI testing framework introduced by google to facilitate automation on a android device or emulators

In command prompt use the commands like

appium driver list

C:\Users\sv22036>appium driver list

Listing available drivers

appium driver install uiautomator2

It will install the uiautomator2 framework for android devices. We will check this one by using command like

For Appium server connection in eclipse we use

Uiautomator2Options op=new Uiautomator2Options ();

op.setDeviceName(“sandeepDevice”);

op.setApp("C:\\Users\\sandeep96\\Desktop\\GeneralStore\\General-Store.apk");

For android devices we use uiautomator2 framework.

AndroidDriver driver=new AndroidDriver(new URL(“http://127.0.0.1.4723”), op);

Here we will get compile time error and import that exception.

Installation of Appium Inspector

* Go to chrome
* Search Appium download
* Select GitHub releases appium/appium desktop
* Install [Appium-Server-GUI-windows-1.22.3-4.exe](https://github.com/appium/appium-desktop/releases/download/v1.22.3-4/Appium-Server-GUI-windows-1.22.3-4.exe)
* then setup the appium inspector

After successfully setup the appium inspector you must give the desired capabilities like

* App path
* DeviceName
* AutomationName UiAutomator2
* platformName android

Then click on the start session button.

From here onwards actual automation script will start and you must inspect all those elements

Mainly here we will use locators like

* AccessibilityId
* Xpath
* id

And also, here we use appiumBy instead of By.

Then creating Base class with util methods and extend it to the child Appium tests.

Base Class:

Public class baseClass

{

@BeforeClass

Public void baseTest()

{

Uiautomator2Options op=new Uiautomator2Options ();

op.setDeviceName(“sandeepDevice”);

Op.setApp(“path”);

AndroidDriver driver=new AndroidDriver(new URL(“http://127.0.0.1.4723”), op);

}

}

Appium Test:

------------------

Public class AppiumTest extends baseClass

{

@Test

Public void appium()

{

Driver.findelement(appiumBy.accessibilityId(“preferences”). click ();

}}

LONG PRESS SCRIPT

-------------------------------

Base Class:

---------------

Public class baseClass

{

@BeforeClass

Public void baseTest()

{

Uiautomator2Options op=new Uiautomator2Options ();

op.setDeviceName(“sandeepDevice”);

Op.setApp(“path”);

AndroidDriver driver=new AndroidDriver(new URL(“http://127.0.0.1.4723”), op);

}

Public void longPress(WebElement ele)

{

((JavascriptExecutor)driver).executeScript(“mobile: LongClickgesture”,ImmutableMap.of(“ElementId”, (RemoteWebElemet)ele).getId(),” duration”,2000));

}

}

Long Press:

---------------

Public class LongPress extends BaseClass

{

@Test

Public void longpress()

{

Driver.findElement(appiumBy.accessibleId(“views”)).click();

Driver.findElement(appiumBy.xpath(“android.widget.TestView[@test=’Expandable Lists’]”)).click();

Driver.findElement(appiumBy.accessibleId(“1.custom Adapter”)). click ();

WebElement ele=((JavascriptExecutor)driver).executeScript(“mobile: LongClickgesture”,ImmutableMap.of(“ElementId”, Driver.findElement(appiumBy.xpath(“android.widget.TestView[@test=People Names’]”));

LongPress(ele);

/\* (RemoteWebElemet)ele).getId(),” duration”,2000)); \*/

}

}

SCROLLING GESTURE:

Base Class:

Public class baseClass

{

@BeforeClass

Public void baseTest()

{

Uiautomator2Options op=new Uiautomator2Options ();

op.setDeviceName(“sandeepDevice”);

Op.setApp(“path”);

AndroidDriver driver=new AndroidDriver(new URL(“http://127.0.0.1.4723”), op);

}

Public void longPress(WebElement ele)

{

((JavascriptExecutor)driver).executeScript(“mobile: LongClickgesture”,ImmutableMap.of(“ElementId”, (RemoteWebElemet)ele).getId(),” duration”,2000));

}

Public void scrollabe()

{

Driver.findElement(AppiumBy.androidUIAutomator(“new UIScrollable(new UISelector()).scrollIntoView(test (\ “webView\”))”);

}

ScrollingTo Element:

----------------------------

Public class Scrolling extends BaseClass

{

@Test

Public void scroll ()

{

Driver.findElement(appiumBy.accessibleId(“views”)). click();

Scrollable ();

/\*Driver.findElement(AppiumBy.androidUIAutomator(“new UIScrollable(new UISelector()).scrollIntoView(test (\ “webView\”))”); \*/

}

}

SWIPE FUNCTIONALITY:

Base Class

Public class baseClass

{

@BeforeClass

Public void baseTest()

{

Uiautomator2Options op=new Uiautomator2Options ();

op.setDeviceName(“sandeepDevice”);

Op.setApp(“path”);

AndroidDriver driver=new AndroidDriver(new URL(“http://127.0.0.1.4723”), op);

}

Public void longPress(WebElement ele)

{

((JavascriptExecutor)driver).executeScript(“mobile: LongClickgesture”,ImmutableMap.of(“ElementId”, (RemoteWebElemet)ele).getId(),” duration”,2000));

}

Public void scrollabe()

{

Driver.findElement(AppiumBy.androidUIAutomator(“new UIScrollable(new UISelector()).scrollIntoView(test (\ “webView\”))”);

}

Public void swipe (WebElement Ele, String direction)

{

((JavascriptExecutor) driver).executeScript("mobile: swipeGesture", ImmutableMap.of(

“ElementId”, (RemoteWebElemet)ele).getId(),  
   
 "direction", "direction",  
 "percent", 0.75  
));

SWIPE FUNCTIONALITY:

-----------------------------------

Public class Swipe extends BaseClass

{

@Test

Public void swipe ()

{

Driver.findElement(appiumBy.accessibleId(“views”)). click();

Driver.findElement(appiumBy.accessibleId(“Gallery”)). click ();

Driver.findElement(appiumBy.xpath(“android.widget.TestView[@test=’1. photos’]”)). click ();

String firstImg=Driver.findElement(appiumBy.xpath(“(android.widget.TestView)[1])”);

Assert.assertEquals(firstImg,” true”);

//swipe

/\*((JavascriptExecutor) driver).executeScript("mobile: swipeGesture", ImmutableMap.of(

“ElementId”, (RemoteWebElemet)ele).getId(),  
   
 "direction", "left",  
 "percent", 0.75  
));\*/

Swipe (firstImg,” left”);

Assert.assertEquals(firstImg,” false”);

DRAG AND DROP:

Base Class

Public class baseClass

{

@BeforeClass

Public void baseTest()

{

Uiautomator2Options op=new Uiautomator2Options ();

op.setDeviceName(“sandeepDevice”);

Op.setApp(“path”);

AndroidDriver driver=new AndroidDriver(new URL(“http://127.0.0.1.4723”), op);

}

Public void longPress(WebElement ele)

{

((JavascriptExecutor)driver).executeScript(“mobile: LongClickgesture”,ImmutableMap.of(“ElementId”, (RemoteWebElemet)ele).getId(),” duration”,2000));

}

Public void scrollabe()

{

Driver.findElement(AppiumBy.androidUIAutomator(“new UIScrollable(new UISelector()).scrollIntoView(test (\ “webView\”))”);

}

Public void swipe (WebElement Ele, String direction)

{

((JavascriptExecutor) driver).executeScript("mobile: swipeGesture", ImmutableMap.of(

“ElementId”, (RemoteWebElemet)ele).getId(),  
   
 "direction", "direction",  
 "percent", 0.75  
));

Drag Functionality:

----------------------

Public class DragAndDrop extends BaseClass

{

@Test

Public void drag ()

{

Driver.findElement(appiumBy.accessibleId(“views”)). click ();

Driver.findElement(appiumBy.accessibleId(“Drag and Drop”)). click ();

String source=Driver.findElement(appiumBy.id(“io.appium.android.apis:id/drag\_dot\_”));

// Java

((JavascriptExecutor) driver).executeScript("mobile: dragGesture", ImmutableMap.of(

"elementId", ((RemoteWebElement) element).getId(),

"endX", 619,

"endY", 560

));Here we need to click on any option in the above options i,e taps co-ordinates it will gives the end X and end Y values.

### mobile: flingGesture

This gesture performs fling gesture on the given element/area. Available since Appium v1.19

#### Supported arguments

* *elementId*: The id of the element to be flinged. If the element id is missing then fling bounding area must be provided. If both the element id and the fling bounding area are provided then this area is effectively ignored.
* *left*: The left coordinate of the fling bounding area
* *top*: The top coordinate of the fling bounding area
* *width*: The width of the fling bounding area
* *height*: The height of the fling bounding area
* *direction*: Direction of the fling. Mandatory value. Acceptable values are: up, down, left and right (case insensitive)
* *speed*: The speed at which to perform this gesture in pixels per second. The value must be greater than the minimum fling velocity for the given view (50 by default). The default value is 7500 \* displayDensity

#### Returned value

The returned value is a boolean one and equals to true if the object can still scroll in the given direction

#### Usage examples

// Java  
boolean canScrollMore = (Boolean) ((JavascriptExecutor) driver).executeScript("mobile: flingGesture", ImmutableMap.of(  
 "elementId", ((RemoteWebElement) element).getId(),  
 "direction", "down",  
 "speed", 500  
));

### mobile: pinchOpenGesture

This gesture performs pinch-open gesture on the given element/area. Available since Appium v1.19

#### Supported arguments

* *elementId*: The id of the element to be pinched. If the element id is missing then pinch bounding area must be provided. If both the element id and the pinch bounding area are provided then the area is effectively ignored.
* *left*: The left coordinate of the pinch bounding area
* *top*: The top coordinate of the pinch bounding area
* *width*: The width of the pinch bounding area
* *height*: The height of the pinch bounding area
* *percent*: The size of the pinch as a percentage of the pinch area size. Valid values must be float numbers in range 0..1, where 1.0 is 100%. Mandatory value.
* *speed*: The speed at which to perform this gesture in pixels per second. The value must not be negative. The default value is 2500 \* displayDensity

#### Usage examples

// Java  
((JavascriptExecutor) driver).executeScript("mobile: pinchOpenGesture", ImmutableMap.of(  
 "elementId", ((RemoteWebElement) element).getId(),  
 "percent", 0.75  
));

### mobile: pinchCloseGesture

This gesture performs pinch-close gesture on the given element/area. Available since Appium v1.19

#### Supported arguments

* *elementId*: The id of the element to be pinched. If the element id is missing then pinch bounding area must be provided. If both the element id and the pinch bounding area are provided then the area is effectively ignored.
* *left*: The left coordinate of the pinch bounding area
* *top*: The top coordinate of the pinch bounding area
* *width*: The width of the pinch bounding area
* *height*: The height of the pinch bounding area
* *percent*: The size of the pinch as a percentage of the pinch area size. Valid values must be float numbers in range 0..1, where 1.0 is 100%. Mandatory value.
* *speed*: The speed at which to perform this gesture in pixels per second. The value must not be negative. The default value is 2500 \* displayDensity

#### Usage examples

// Java  
((JavascriptExecutor) driver).executeScript("mobile: pinchCloseGesture", ImmutableMap.of(  
 "elementId", ((RemoteWebElement) element).getId(),  
 "percent", 0.75  
));

Real Time Project:

------------------------

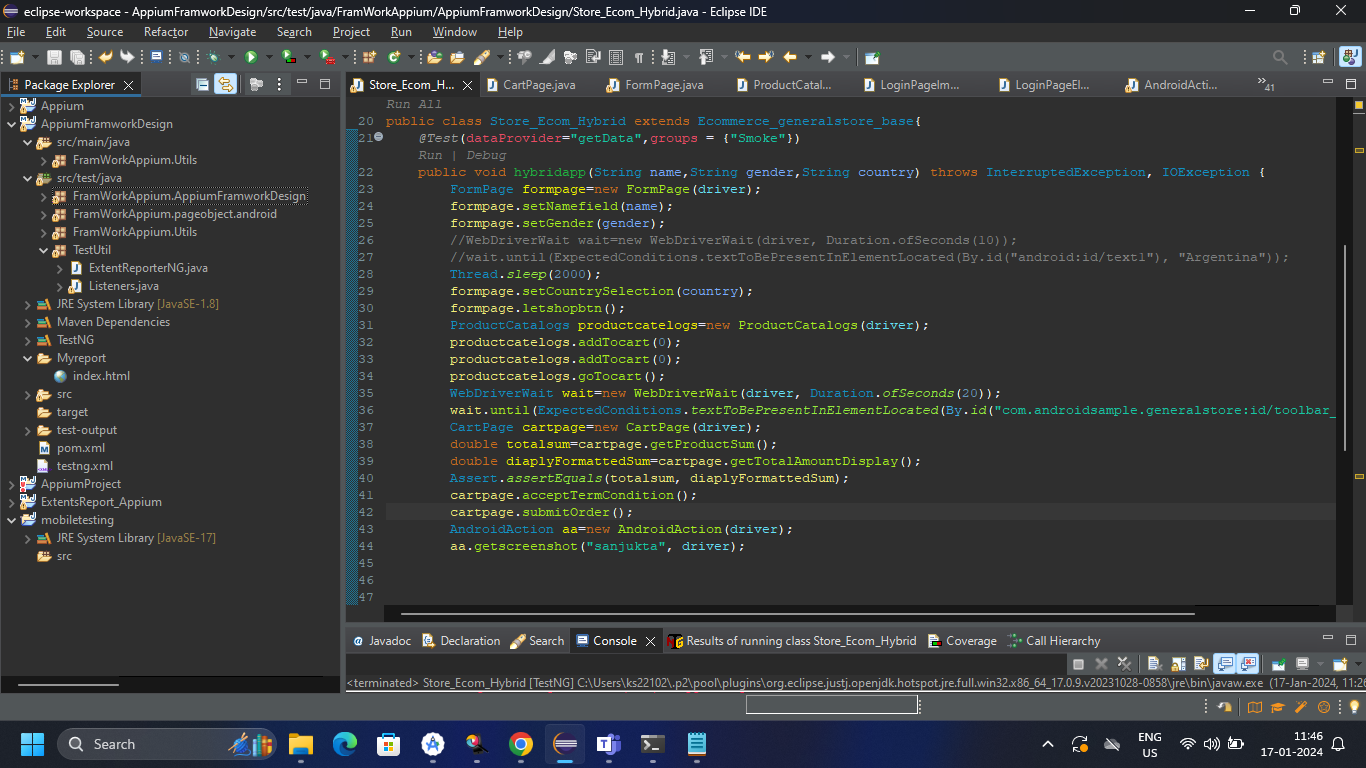
C:\Users\sandeep96\AppData\Local\Android\Sdk\platform-tools>adb install C:\Users\sandeep96\Desktop\GeneralStore\General-Store.apk

Performing Streamed Install

Success

By using above command we will install the General-Store app into our virtual device.

FrameWork----------



Ecommerece\_base class-

package FramWorkAppium.AppiumFramworkDesign;

import java.net.MalformedURLException;

import java.net.URL;

import java.time.Duration;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.remote.RemoteWebElement;

import org.testng.annotations.BeforeClass;

import com.google.common.collect.ImmutableMap;

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

import io.appium.java\_client.android.options.UiAutomator2Options;

public class Ecommerce\_generalstore\_base {

AndroidDriver driver;

*@BeforeClass*(alwaysRun = true)

public void app() throws MalformedURLException

{

// AppiumDriverLocalService service=new AppiumServiceBuilder().withAppiumJS(new File("C:\\Users\\ks22102\\AppData\\Roaming\\npm\\node\_modules\\appium\\build\\lib\\main.js"))

// .withIPAddress("127.0.0.1").usingPort(4723).build();

// service.start();

UiAutomator2Options options=new UiAutomator2Options();

options.setDeviceName("sanjukta");

//options.setChromedriverExecutable("C:\\Users\\ks22102\\eclipseworkspace\\Appium\\src\\test\\java\\resources\\chromedriver.exe");

options.setApp("C:\\Users\\ks22102\\eclipseworkspace\\Appium\\src\\test\\java\\resources\\General-Store.apk");

driver=new AndroidDriver (new URL("http://127.0.0.1:4723"), options);

driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(20));

}

public Double getformattedAmout(String amount) {

Double price=Double.*parseDouble*(amount.substring(1));

return null;

}

}

Framwork.Pageobject.Android-

1.cartPage-

package FramWorkAppium.pageobject.android;

import java.util.List;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.PageFactory;

import FramWorkAppium.Utils.AndroidAction;

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

import io.appium.java\_client.pagefactory.AndroidFindBy;

import io.appium.java\_client.pagefactory.AppiumFieldDecorator;

public class CartPage extends AndroidAction{

AndroidDriver driver;

@AndroidFindBy(id="com.androidsample.generalstore:id/productPrice")

private List<WebElement> productprice;

@AndroidFindBy (id="com.androidsample.generalstore:id/totalAmountLbl")

private WebElement totalamount;

@AndroidFindBy(id="com.androidsample.generalstore:id/termsButton")

private WebElement terms;

@AndroidFindBy(id="android:id/button1")

private WebElement acceptbutton;

@AndroidFindBy(xpath="(//android.widget.CheckBox[@text='Send me e-mails on discounts related to selected products in future'])")

private WebElement checkbox;

@AndroidFindBy(id="com.androidsample.generalstore:id/btnProceed")

private WebElement proceedbtn;

public CartPage(AndroidDriver driver) {

super(driver);

this.driver=driver;

PageFactory.initElements(new AppiumFieldDecorator(driver), this);

}

public List<WebElement> getproductPriceList() {

return productprice;

}

public double getProductSum() {

int count=productprice.size();

double sum=0;

//retrive the price

for(int i=0;i<count;i++) {

String amountstring=productprice.get(i).getText();

//remove the dollar and convert into double(

Double priceam= Double.parseDouble((amountstring).substring(1));

sum=sum+priceam;

}

return sum;

}

public double getTotalAmountDisplay() {

return getformattedAmout(totalamount.getText());

}

public void acceptTermCondition() throws InterruptedException {

longpress(terms);

acceptbutton.click();

}

public void submitOrder() {

checkbox.click();

proceedbtn.click();

}

public void teardown() {

driver.quit();

}

}

2.FormPage-

package FramWorkAppium.pageobject.android;

import org.openqa.selenium.By;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.PageFactory;

import FramWorkAppium.Utils.AndroidAction;

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

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

import io.appium.java\_client.pagefactory.AndroidFindBy;

import io.appium.java\_client.pagefactory.AppiumFieldDecorator;

public class FormPage extends AndroidAction{

AndroidDriver driver;

public FormPage(AndroidDriver driver) {

super(driver);//it will call the parent class driver

this.driver=driver;//to avoid confusion

PageFactory.*initElements*(new AppiumFieldDecorator(driver), this);

}

*@AndroidFindBy*(id="com.androidsample.generalstore:id/nameField")

private WebElement namefield;

*@AndroidFindBy*(xpath="//android.widget.RadioButton[@text='Female']")

private WebElement femaleoption;

*@AndroidFindBy*(xpath="//android.widget.RadioButton[@text='Male']")

private WebElement maleoption;

*@AndroidFindBy*(id="android:id/text1")

private WebElement countryselection;

*@AndroidFindBy*(id="com.androidsample.generalstore:id/btnLetsShop")

private WebElement letsshopbutton;

public void setNamefield(String name) {

namefield.sendKeys(name);

driver.hideKeyboard();

}

public void setGender(String gender) {

if(gender.contains("female")) {

femaleoption.click();

}else {

maleoption.click();

}

}

public void setCountrySelection(String countryName) {

countryselection.click();

scrollToText(countryName);

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

}

public void letshopbtn() {

letsshopbutton.click();

}

public void setActivity() {

Activity activity = new Activity("resources", "com.androidsample.generalstore.MainActivity");

//driver.startsActivity(activity);

}

}

3.CategoryPage-

package FramWorkAppium.pageobject.android;

import java.util.List;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.PageFactory;

import FramWorkAppium.Utils.AndroidAction;

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

import io.appium.java\_client.pagefactory.AndroidFindBy;

import io.appium.java\_client.pagefactory.AppiumFieldDecorator;

public class ProductCatalogs extends AndroidAction{

AndroidDriver driver;

*@AndroidFindBy*(xpath="(//android.widget.TextView)[@text='ADD TO CART']")

private List< WebElement> addtocart;

*@AndroidFindBy*(id="com.androidsample.generalstore:id/appbar\_btn\_cart")

private WebElement cart;

public ProductCatalogs(AndroidDriver driver) {

super(driver);

this.driver=driver;

PageFactory.*initElements*(new AppiumFieldDecorator(driver), this);

}

public void addTocart(int index) {

addtocart.get(index).click();

}

public void goTocart() throws InterruptedException {

cart.click();

Thread.*sleep*(2000);

}

public void screens() {

}

}

FramWorkAppium.Util-

1.AndroidAction\_ReusableClass

package FramWorkAppium.Utils;

import java.io.File;

import java.io.IOException;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.OutputType;

import org.openqa.selenium.TakesScreenshot;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.remote.RemoteWebElement;

import org.openqa.selenium.support.PageFactory;

import com.google.common.collect.ImmutableMap;

import com.google.common.io.Files;

import io.appium.java\_client.AppiumBy;

import io.appium.java\_client.AppiumDriver;

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

public class AndroidAction {

AndroidDriver driver;

public AndroidAction() {

}

public AndroidAction(AndroidDriver driver) {

this.driver=driver;

}

public void longpress(WebElement ele) throws InterruptedException {

((JavascriptExecutor)driver).executeScript("mobile: longClickGesture",ImmutableMap.of(

"elementId",((RemoteWebElement)ele).getId(),"duration",2000));

Thread.sleep(2000);

}

public void scrollToText(String ele) {

driver.findElement(AppiumBy.androidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\""+ele+"\"));"));

}

public void scrollToEndAction() {

boolean canScrollMore;

do {

canScrollMore = (Boolean) ((JavascriptExecutor) driver).executeScript("mobile: scrollGesture", ImmutableMap.of("left", 100, "top", 100, "width", 200, "height", 200,"direction", "down", "percent", 3.0));

}while(canScrollMore);

}

public void swipeAction(WebElement ele,String direction) {

((JavascriptExecutor) driver).executeScript("mobile: swipeGesture", ImmutableMap.of("elementId",

((RemoteWebElement)ele).getId(),

"direction", "left",

"percent", 0.75

));

}

public void dragAndDrop(WebElement ele,int endX,int endY) {

((JavascriptExecutor) driver).executeScript("mobile: dragGesture", ImmutableMap.of(

"elementId", ((RemoteWebElement) ele).getId(),

"endX", 621,

"endY", 560

));

}

public Double getformattedAmout(String amount) {

Double price=Double.parseDouble(amount.substring(1));

return price;

}

public String getscreenshot(String testCaseName,AppiumDriver driver) throws IOException {

File sorce=driver.getScreenshotAs(OutputType.FILE);

String destination=System.getProperty("user.dir")+"\\Myreport"+testCaseName+".png";

Files.copy(sorce, new File(destination));

return destination;

//1.capture and place in folder 2.extent report and pick file and attach in the report

}

public void screenshot(WebDriver driver,String index)

{

TakesScreenshot ts = (TakesScreenshot)driver;

File src = ts.getScreenshotAs(OutputType.FILE);

File dest=new File("./screenshots/"+index+".png");

try {

Files.copy(src, dest);

} catch (Exception e) {

System.out.println(e.getMessage());

e.printStackTrace();

}

System.out.println("Screenshot took successfully");

}

}

2.ExtentReport-

package FramWorkAppium.Utils;

import com.aventstack.extentreports.ExtentReports;

importcom.aventstack.extentreports.reporter.ExtentSparkReporter;

public class ExtentReporterNG {

static ExtentReports extent;

public static ExtentReports getReporterObject() {

String path=System.getProperty("user.dir")+"\\Myreport\\index.html";

//String path="C:\\Users\\ks22102\\eclipse-workspace\\ExtentsReport\_Appium\\reports123\\index.html";

//it is a helper class of report it will attached to the main report

ExtentSparkReporter reporter=new ExtentSparkReporter(path);

reporter.config().setReportName("WebAutomation");

reporter.config().setDocumentTitle("Test Results");

//it is main class of extent report

extent=new ExtentReports();

extent.attachReporter(reporter);

extent.setSystemInfo("Tester", "sanjukta senapati");

return extent;

}}

3.Listener-

package FramWorkAppium.Utils;

import static org.testng.Assert.ARRAY\_MISMATCH\_TEMPLATE;

import java.io.IOException;

import org.testng.ITestContext;

import org.testng.ITestListener;

import org.testng.ITestResult;

import com.aventstack.extentreports.ExtentReports;

import com.aventstack.extentreports.ExtentTest;

import com.aventstack.extentreports.Status;

import FramWorkAppium.Utils.AndroidAction;

import io.appium.java\_client.AppiumDriver;

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

public class Listeners extends AndroidAction implements ITestListener{

AppiumDriver driver;

public Listeners(AndroidDriver driver) {

super(driver);

// TODO Auto-generated constructor stub

}

public Listeners() {

}

ExtentTest test;

ExtentReports extent =ExtentReporterNG.getReporterObject();

@Override

public void onTestStart(ITestResult result) {

test=extent.createTest(result.getMethod().getMethodName());

}

@Override

public void onTestSuccess(ITestResult result) {

test.log(Status.PASS, "Test Passed");

}

@Override

public void onTestFailure(ITestResult result) {

test.fail(result.getThrowable());

try {

try {

driver=(AppiumDriver)result.getTestClass().getRealClass().getField("driver").get(result.getInstance());

} catch (IllegalAccessException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (NoSuchFieldException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (SecurityException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

} catch (IllegalArgumentException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

try {

test.addScreenCaptureFromPath(getscreenshot(result.getMethod().getMethodName(),driver), result.getMethod().getMethodName());

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

//test.addScreenCaptureFromPath(getscreenshot(result.getMethod().getMethodName(),driver), result.getMethod().getMethodName());

}

@Override

public void onTestSkipped(ITestResult result) {

// TODO Auto-generated method stub

ITestListener.super.onTestSkipped(result);

}

@Override

public void onTestFailedButWithinSuccessPercentage(ITestResult result) {

// TODO Auto-generated method stub

ITestListener.super.onTestFailedButWithinSuccessPercentage(result);

}

@Override

public void onTestFailedWithTimeout(ITestResult result) {

// TODO Auto-generated method stub

ITestListener.super.onTestFailedWithTimeout(result);

}

@Override

public void onStart(ITestContext context) {

// TODO Auto-generated method stub

ITestListener.super.onStart(context);

}

@Override

public void onFinish(ITestContext context) {

extent.flush();

}

}

FramworkAppium\_mainScript-

package FramWorkAppium.AppiumFramworkDesign;

import java.io.IOException;

import java.time.Duration;

import org.testng.annotations.Test;

import org.testng.AssertJUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.support.ui.ExpectedConditions;

import org.openqa.selenium.support.ui.WebDriverWait;

import org.testng.Assert;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.DataProvider;

import FramWorkAppium.Utils.AndroidAction;

import FramWorkAppium.pageobject.android.CartPage;

import FramWorkAppium.pageobject.android.FormPage;

import FramWorkAppium.pageobject.android.ProductCatalogs;

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

public class Store\_Ecom\_Hybrid extends Ecommerce\_generalstore\_base{

*@Test*(dataProvider="getData",groups = {"Smoke"})

public void hybridapp(String name,String gender,String country) throws InterruptedException, IOException {

FormPage formpage=new FormPage(driver);

formpage.setNamefield(name);

formpage.setGender(gender);

//WebDriverWait wait=new WebDriverWait(driver, Duration.ofSeconds(10));

//wait.until(ExpectedConditions.textToBePresentInElementLocated(By.id("android:id/text1"), "Argentina"));

Thread.*sleep*(2000);

formpage.setCountrySelection(country);

formpage.letshopbtn();

ProductCatalogs productcatelogs=new ProductCatalogs(driver);

productcatelogs.addTocart(0);

productcatelogs.addTocart(0);

productcatelogs.goTocart();

WebDriverWait wait=new WebDriverWait(driver, Duration.*ofSeconds*(20));

wait.until(ExpectedConditions.*textToBePresentInElementLocated*(By.*id*("com.androidsample.generalstore:id/toolbar\_title"), "Cart"));

CartPage cartpage=new CartPage(driver);

double totalsum=cartpage.getProductSum();

double diaplyFormattedSum=cartpage.getTotalAmountDisplay();

Assert.*assertEquals*(totalsum, diaplyFormattedSum);

cartpage.acceptTermCondition();

cartpage.submitOrder();

AndroidAction aa=new AndroidAction(driver);

aa.getscreenshot("sanjukta", driver);

}

*@BeforeMethod*

public void presetup() {

Activity activity = new Activity("com.androidsample.generalstore", "com.androidsample.generalstore.MainActivity");

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

}

*@DataProvider*

public Object[] [] getData() {

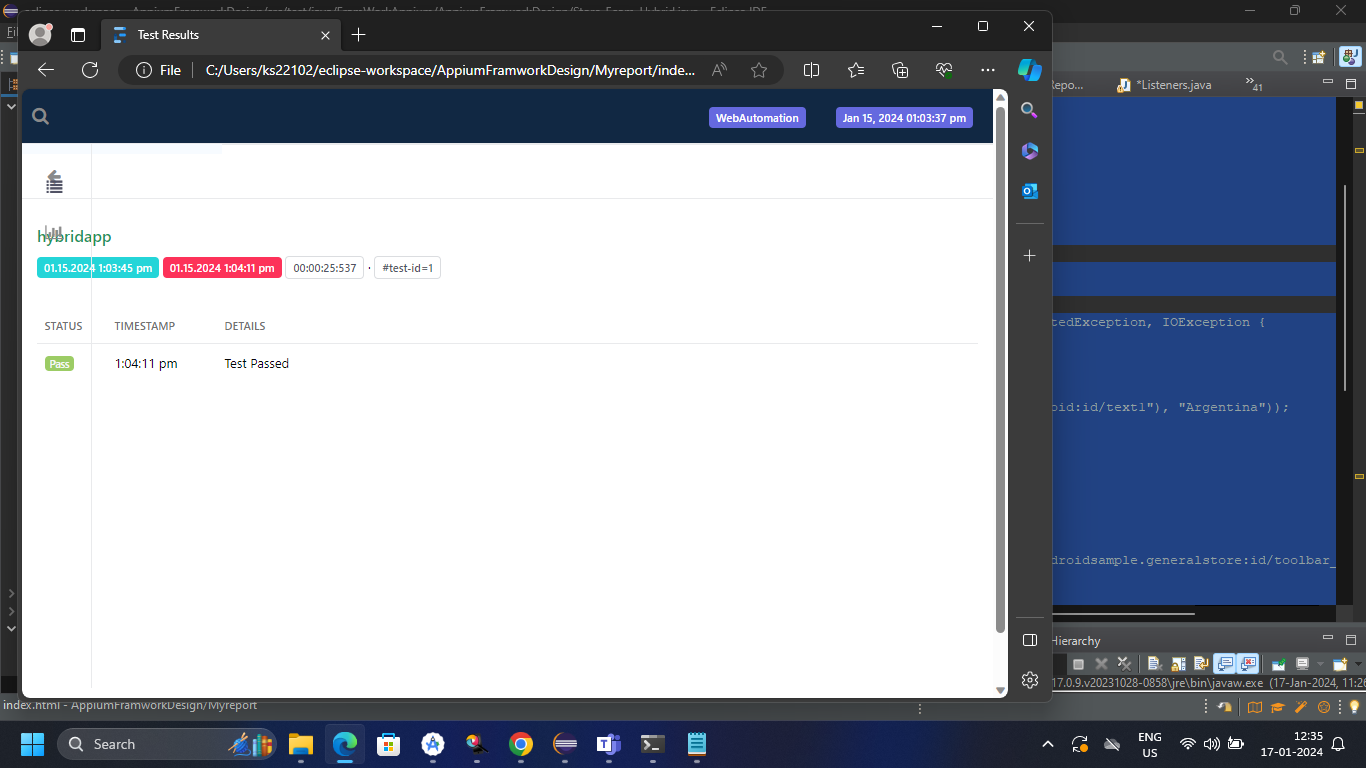
return new Object [][] {{"sanjukta senapati","female","Argentina"},

};

}

}

Report



Appium Practice

BaseTest-

package sanjuktaproject.Appium;

import java.io.File;

import java.net.MalformedURLException;

import java.net.URL;

import java.time.Duration;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.remote.RemoteWebElement;

import org.testng.annotations.AfterClass;

import org.testng.annotations.BeforeClass;

import com.google.common.collect.ImmutableMap;

import com.google.common.util.concurrent.Service;

import io.appium.java\_client.AppiumBy;

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

import io.appium.java\_client.android.options.UiAutomator2Options;

import io.appium.java\_client.service.local.AppiumDriverLocalService;

import io.appium.java\_client.service.local.AppiumServiceBuilder;

public class BaseTest {

public AndroidDriver driver;

public AppiumDriverLocalService service;

@BeforeClass

public void configureAppium() throws MalformedURLException {

// service=new AppiumServiceBuilder().withAppiumJS(new File("C:\\Users\\ks22102\\AppData\\Roaming\\npm\\node\_modules\\appium\\build\\lib\\main.js"))

// .withIPAddress("127.0.0.1").usingPort(4723).build();

//service.start();

UiAutomator2Options options=new UiAutomator2Options();

options.setDeviceName("sanjukta");

options.setApp("C:\\Users\\ks22102\\eclipseworkspace\\Appium\\src\\test\\java\\resources\\ApiDemos-debug.apk");

driver=new AndroidDriver (new URL("http://127.0.0.1:4723"), options);

driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));

}

public void longpress(WebElement ele) throws InterruptedException {

((JavascriptExecutor)driver).executeScript("mobile: longClickGesture",ImmutableMap.of(

"elementId",((RemoteWebElement)ele).getId(),"duration",2000));

Thread.sleep(2000);

}

}}

public void scrollUptoexactElement(WebElement ele) {

driver.findElement(AppiumBy.androidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"ele\"));"));

}

public void scrollToEndAction() {

boolean canScrollMore;

do {

canScrollMore = (Boolean) ((JavascriptExecutor) driver).executeScript("mobile: scrollGesture",

ImmutableMap.of(

"left", 100, "top", 100, "width", 200, "height", 200,

"direction", "down",

"percent", 3.0

));

}while(canScrollMore);

}

public void swipeAction(WebElement ele,String direction) {

((JavascriptExecutor) driver).executeScript("mobile: swipeGesture", ImmutableMap.of("elementId",

((RemoteWebElement)ele).getId(),

"direction", "left",

"percent", 0.75

));

}

public void dragAndDrop(WebElement ele,int endX,int endY) {

((JavascriptExecutor) driver).executeScript("mobile: dragGesture", ImmutableMap.of(

"elementId", ((RemoteWebElement) ele).getId(),

"endX", 621,

"endY", 560

));

}

public Double getformattedAmout(String amount) {

Double price=Double.parseDouble(amount.substring(1));

return null;

}

@AfterClass

public void tearDown() {

//driver.quit();

//service.stop();

}

}

2.Testcase-

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.testng.Assert;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

public class Appium\_testcase extends BaseTest{

@Test

public void wifisetting() {

driver.findElement(AppiumBy.accessibilityId("Preference")).click();

driver.findElement(AppiumBy.accessibilityId("3. Preference dependencies")).click();

//driver.findElement(AppiumBy.accessibilityId("")).click();

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

driver.findElement(By.xpath("(//android.widget.RelativeLayout)[2]")).click();

//assertion

String alerttitle=driver.findElement(By.id("android:id/alertTitle")).getText();

Assert.assertEquals(alerttitle, "WiFi settings");

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

driver.findElements(AppiumBy.className("android.widget.Button")).get(1).click();

}

}

3.Activity-

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.openqa.selenium.DeviceRotation;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

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

public class AppPackage\_Activity extends BaseTest {

@Test

public void activityDemo() {

// To skip navigation step we can directly navigate to required page(wifi page)

// by using activity class

Activity activity = new Activity("com.google.android.apps.nexuslauncher", "com.google.android.apps.nexuslauncher.NexusLauncherActivity");

// driver.findElement(AppiumBy.accessibilityId("Preference")).click();

//clickon prference dependencies

//driver.findElement(AppiumBy.accessibilityId("3. Preference dependencies")).click();

//click on wifi checkbox

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

// for device rotation for chnage landscape mode

//DeviceRotation landscape = new DeviceRotation(0, 0, 90);

//driver.rotate(landscape);

}

}

4.basicTestcase-

package sanjuktaproject.Appium;

import java.io.File;

import java.net.MalformedURLException;

import java.net.URL;

import org.openqa.selenium.By;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

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

import io.appium.java\_client.android.options.UiAutomator2Options;

import io.appium.java\_client.service.local.AppiumDriverLocalService;

import io.appium.java\_client.service.local.AppiumServiceBuilder;

public class AppiumBasics {

AndroidDriver driver;

@Test

public void app() throws MalformedURLException

{

// AppiumDriverLocalService service=new AppiumServiceBuilder().withAppiumJS(new File("C:\\Users\\ks22102\\AppData\\Roaming\\npm\\node\_modules\\appium\\build\\lib\\main.js"))

// .withIPAddress("127.0.0.1").usingPort(4723).build();

// service.start();

UiAutomator2Options options=new UiAutomator2Options();

options.setDeviceName("sanjukta");

options.setApp("C:\\Users\\ks22102\\eclipseworkspace\\Appium\\src\\test\\java\\resources\\ApiDemos-debug.apk");

driver=new AndroidDriver (new URL("http://127.0.0.1:4723"), options);

driver.findElement(AppiumBy.accessibilityId("Preference")).click();

driver.findElement(AppiumBy.accessibilityId("3. Preference dependencies")).click();

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

// service.stop();

//driver.quit();

}

}

4.DragAndDrop-

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.remote.RemoteWebElement;

import org.testng.Assert;

import org.testng.annotations.Test;

import com.google.common.collect.ImmutableMap;

import io.appium.java\_client.AppiumBy;

public class DragAndDropTest extends BaseTest{

@Test

public void dragdropDemo() throws InterruptedException {

//click on views

driver.findElement(AppiumBy.accessibilityId("Views")).click();

//click on drag and drop

driver.findElement(AppiumBy.accessibilityId("Drag and Drop")).click();

//store the store address

WebElement source=driver.findElement(By.id("io.appium.android.apis:id/drag\_dot\_1"));

//drag and drop in the perticular cordinates

dragAndDrop(source, 621, 560);

/\*

\* ((JavascriptExecutor) driver).executeScript("mobile: dragGesture",

\* ImmutableMap.of( "elementId", ((RemoteWebElement) source).getId(), "endX",

\* 621, "endY", 560

\*

\* ));

\*/

String text=driver.findElement(By.id("io.appium.android.apis:id/drag\_result\_text")).getText();

System.out.println(text);

Assert.assertEquals(text, "Dropped!");

Thread.sleep(2000);

}

}

5.LongPress-

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.devtools.idealized.Javascript;

import org.openqa.selenium.remote.RemoteWebElement;

import org.testng.Assert;

import org.testng.annotations.Test;

import com.google.common.collect.ImmutableMap;

import io.appium.java\_client.AppiumBy;

public class Long\_press extends BaseTest {

@Test

public void longpressGesture() throws InterruptedException {

driver.findElement(AppiumBy.accessibilityId("Views")).click();

driver.findElement(AppiumBy.accessibilityId("Expandable Lists")).click();

driver.findElement(By.xpath("//android.widget.TextView[@content-desc=\"1. Custom Adapter\"]")).click();

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

longpress(ele);

//((JavascriptExecutor)driver).executeScript("mobile: longClickGesture",ImmutableMap.of(

// "elementId",((RemoteWebElement)ele).getId(),"duration",2000));

// Thread.sleep(2000);

String menuetext=driver.findElement(By.id("android:id/title")).getText();

Assert.assertEquals(menuetext, "Sample menu");

Assert.assertTrue(driver.findElement(By.id("android:id/title")).isDisplayed());

}

}

5.ScrollTest-

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.testng.annotations.Test;

import com.google.common.collect.ImmutableMap;

import io.appium.java\_client.AppiumBy;

public class ScrollTest extends BaseTest{

@Test

public void scrollDownDemo() throws InterruptedException {

driver.findElement(AppiumBy.accessibilityId("Views")).click();

//where to scroll is known prior or upto exact element

//driver.findElement(AppiumBy.androidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"WebView\"));"));

scrollToEndAction();

//No prior idea

/\*

\* boolean canScrollMore; do { canScrollMore = (Boolean) ((JavascriptExecutor)

\* driver).executeScript("mobile: scrollGesture", ImmutableMap.of( "left", 100,

\* "top", 100, "width", 200, "height", 200, "direction", "down", "percent", 3.0

\* )); }while(canScrollMore);

\*/

Thread.sleep(2000);

}

}

6.GeneralStrore APK-

i)testScenario-1

Addto cart-

package sanjuktaproject.Appium;

import java.time.Duration;

import org.openqa.selenium.By;

import org.openqa.selenium.support.ui.ExpectedConditions;

import org.openqa.selenium.support.ui.WebDriverWait;

import org.testng.Assert;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

import io.appium.java\_client.functions.ExpectedCondition;

public class Store\_AddTocart extends Ecommerce\_generalstore\_base {

@Test

public void addToCart() throws InterruptedException {

//enter the name

driver.findElement(By.id("com.androidsample.generalstore:id/nameField")).sendKeys("sanjukta senapati");

//we need to hide keyboard

driver.hideKeyboard();

//select radio button as female

driver.findElement(By.xpath("//android.widget.RadioButton[@text='Female']")).click();

//click on country dropdown

driver.findElement(By.id("android:id/text1")).click();

//select argentina from dropdown

driver.findElement(AppiumBy.androidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Argentina\"));"));

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

//Click on lets shop

driver.findElement(By.id("com.androidsample.generalstore:id/btnLetsShop")).click();

//scroll untill views of jordan 6

driver.findElement(AppiumBy.androidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Jordan 6 Rings\"));"));

//count the product who have same id

int productcount=driver.findElements(By.id("com.androidsample.generalstore:id/productName")).size();

//run in loop and print

//driver.findElement(By.xpath("(//android.widget.TextView)[@text='Jordan 6 Rings']")).click();

for(int i=0;i<productcount;i++) {

String productname=driver.findElements(By.id("com.androidsample.generalstore:id/productName")).get(i).getText(); //the //

// product which is matches with jordan 6 Rings it will add to cart

if(productname.equalsIgnoreCase("Jordan 6 Rings")) {

driver.findElements(By.id("com.androidsample.generalstore:id/productAddCart")).get(i).click(); } }

//go to cart

driver.findElement(By.id("com.androidsample.generalstore:id/appbar\_btn\_cart")

).click(); Thread.sleep(2000);

//wait until the cart page is loaded

//WebDriverWait wait=new WebDriverWait(driver, Duration.ofSeconds(15));

// wait.until(ExpectedConditions.attributeContains(driver.findElement(By.id("com.androidsample.generalstore:id/toolbar\_title")), "text", "cart"));

//validate the exact item is added in the page or not

String lastpageproduct=driver.findElement(By.id("com.androidsample.generalstore:id/productName")).getText();

Assert.assertEquals(lastpageproduct, "Jordan 6 Rings");

}

}

TestScenario-2

AddproductSumPrice-

package sanjuktaproject.Appium;

import java.util.List;

import org.openqa.selenium.By;

import org.openqa.selenium.WebElement;

import org.testng.Assert;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

public class Store\_addproduct\_Sumprice extends Ecommerce\_generalstore\_base{

@Test

public void storeDemo() throws InterruptedException {

driver.findElement(By.id("com.androidsample.generalstore:id/nameField")).sendKeys("sanjukta senapati");

//we need to hide keyboard

driver.hideKeyboard();

//select radio button as female

driver.findElement(By.xpath("//android.widget.RadioButton[@text='Female']")).click();

//click on country dropdown

driver.findElement(By.id("android:id/text1")).click();

//select argentina from dropdown

driver.findElement(AppiumBy.androidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Argentina\"));"));

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

//Click on lets shop

driver.findElement(By.id("com.androidsample.generalstore:id/btnLetsShop")).click();

//add the first item from shop

driver.findElements(By.xpath("(//android.widget.TextView)[@text='ADD TO CART']")).get(0).click();

//add 2nd item from shop after adding 1st item the 2nd item become zero index in that shop

driver.findElements(By.xpath("(//android.widget.TextView)[@text='ADD TO CART']")).get(0).click();

//click on cart

driver .findElement(By.id("com.androidsample.generalstore:id/appbar\_btn\_cart")).click();

//stote the two price in a list

List<WebElement>price=driver.findElements(By.id("com.androidsample.generalstore:id/productPrice"));

//size of price

int count=price.size();

double sum=0;

//retrive the price

for(int i=0;i<count;i++) {

String amountstring=price.get(i).getText();

//remove the dollar and convert into double(

Double priceam= Double.parseDouble((amountstring).substring(1));

sum=sum+priceam;

}

//to validate the total price displayed as sum of the two product are not

String displaysum=driver.findElement(By.id("com.androidsample.generalstore:id/totalAmountLbl")).getText();

//String a=((displaysum).substring(0));

Double formatstring=getformattedAmout(displaysum);//Double.parseDouble(((displaysum).substring(1)));

//Assert.assertEquals(displaysum, b);

//long press on term and condition

WebElement termcond=driver.findElement(By.id("com.androidsample.generalstore:id/termsButton"));

longpress(termcond);//it is coming from ecommerce base class

String term=driver.findElement(By.id("com.androidsample.generalstore:id/alertTitle")).getText();

Assert.assertEquals(term, "Terms Of Conditions");

//click on close button of the term and condition pop up

driver.findElement(By.id("android:id/button1")).click();

//to click check box

driver.findElement(By.xpath("(//android.widget.CheckBox[@text='Send me e-mails on discounts related to selected products in future'])")).click();

//to click on view to purchase

driver.findElement(By.id("com.androidsample.generalstore:id/btnProceed")).click();

}

}

3.TestScenario-

package sanjuktaproject.Appium;

import java.util.List;

import java.util.Set;

import org.testng.annotations.Test;

import org.testng.AssertJUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.Keys;

import org.openqa.selenium.WebElement;

import org.testng.Assert;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

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

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

public class Store\_Ecom\_Hybrid extends Ecommerce\_generalstore\_base{

*@Test*

public void hybridapp() throws InterruptedException {

driver.findElement(By.*id*("com.androidsample.generalstore:id/nameField")).sendKeys("sanjukta senapati");

//we need to hide keyboard

driver.hideKeyboard();

//select radio button as female

[driver.findElement(By.*xpath*("//android.widget.RadioButton[@text='Female']")).click](mailto:driver.findElement(By.xpath("//android.widget.RadioButton[@text='Female']")).click)();

//click on country dropdown

driver.findElement(By.*id*("android:id/text1")).click();

//select argentina from dropdown

driver.findElement(AppiumBy.*androidUIAutomator*("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Argentina\"));"));

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

//Click on lets shop

driver.findElement(By.*id*("com.androidsample"

+ ".generalstore:id/btnLetsShop")).click();

//add the first item from shop

driver.findElements(By.*xpath*("(//android.widget.TextView)[@text='ADD TO CART']")).get(0).click();

//add 2nd item from shop after adding 1st item the 2nd item become zero index in that shop

driver.findElements(By.*xpath*("(//android.widget.TextView)[@text='ADD TO CART']")).get(0).click();

//click on cart

driver .findElement(By.*id*("com.androidsample.generalstore:id/appbar\_btn\_cart")).click();

//stote the two price in a list

List<WebElement>price=driver.findElements(By.*id*("com.androidsample.generalstore:id/productPrice"));

//size of price

int count=price.size();

double sum=0;

//retrive the price

for(int i=0;i<count;i++) {

String amountstring=price.get(i).getText();

//remove the dollar and convert into double(

Double priceam= Double.*parseDouble*((amountstring).substring(1));

sum=sum+priceam;

}

//to validate the total price displayed as sum of the two product are not

String displaysum=driver.findElement(By.*id*("com.androidsample.generalstore:id/totalAmountLbl")).getText();

//String a=((displaysum).substring(0));

Double formatstring=getformattedAmout(displaysum);//Double.parseDouble(((displaysum).substring(1)));

//Assert.assertEquals(displaysum, b);

//long press on term and condition

WebElement termcond=driver.findElement(By.*id*("com.androidsample.generalstore:id/termsButton"));

longpress(termcond);//it is coming from ecommerce base class

String term=driver.findElement(By.*id*("com.androidsample.generalstore:id/alertTitle")).getText();

Assert.*assertEquals*(term, "Terms Of Conditions");

//click on close button of the term and condition pop up

driver.findElement(By.*id*("android:id/button1")).click();

//to click check box

driver.findElement(By.*xpath*("(//android.widget.CheckBox[@text='Send me e-mails on discounts related to selected products in future'])")).click();

//to click on view to purchase

driver.findElement(By.*id*("com.androidsample.generalstore:id/btnProceed")).click();

//to get the context(it will tell weather native or web or both here we are storing the context in set)

Set<String>contexts=driver.getContextHandles();

for(String contextName:contexts) {

System.***out***.println(contextName);

}

//code is for web app but is is not showing as web so, this code is not running

/\*

\* driver.context("WEBVIEW\_com.androidsample.generalstore");

\* driver.findElement(By.name("q")).sendKeys("rahul shetty acadamy");

\* driver.findElement(By.name("q")).sendKeys(Keys.ENTER); driver.pressKey(new

\* KeyEvent(AndroidKey.BACK)); driver.context("NATIVE\_APP");

\*/

}

}

5.TestScenario-6(swipe function)

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.remote.RemoteWebElement;

import org.testng.Assert;

import org.testng.annotations.Test;

import com.google.common.collect.ImmutableMap;

import io.appium.java\_client.AppiumBy;

public class SwipeFunctionalityTest extends BaseTest{

@Test

public void scrollDownDemo() throws InterruptedException {

//click on views

driver.findElement(AppiumBy.accessibilityId("Views")).click();

//click on galary

driver.findElement(AppiumBy.accessibilityId("Gallery")).click();

//click on photo

driver.findElement(By.xpath("//android.widget.TextView[@content-desc=\"1. Photos\"]")).click();

//first image

WebElement firstImage=driver.findElement(By.xpath("(//android.widget.ImageView)[1]"));

Assert.assertEquals(driver.findElement(By.xpath("(//android.widget.ImageView)[1]")).getAttribute("focusable"), "true");

//swipe photo

swipeAction(firstImage, "left");

/\*

\* ((JavascriptExecutor) driver).executeScript("mobile: swipeGesture",

\* ImmutableMap.of("elementId", ((RemoteWebElement)firstImage).getId(),

\* "direction", "left", "percent", 0.75 ));

\*/

Assert.assertEquals(driver.findElement(By.xpath("(//android.widget.ImageView)[1]")).getAttribute("focusable"), "false");

}

}

TestScenario-7

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.testng.Assert;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

public class Store\_FillformTest extends Ecommerce\_generalstore\_base{

@Test

public void fillform() {

//enter the name

driver.findElement(By.id("com.androidsample.generalstore:id/nameField")).sendKeys("sanjukta senapati");

//we need to hide keyboard

driver.hideKeyboard();

//select radio button as female

driver.findElement(By.xpath("//android.widget.RadioButton[@text='Female']")).click();

//click on country dropdown

driver.findElement(By.id("android:id/text1")).click();

//select argentina from dropdown

driver.findElement(AppiumBy.androidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Argentina\"));"));

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

//Click on lets shop

driver.findElement(By.id("com.androidsample.generalstore:id/btnLetsShop")).click();

//validate toast message

String toastmsg=driver.findElement(By.xpath("(//android.widget.toast)[1]")).getAttribute("name");

Assert.assertEquals(toastmsg, "Please enter your name");

}

}

8.DifferentActivity

package sanjuktaproject.Appium;

import org.openqa.selenium.By;

import org.openqa.selenium.DeviceRotation;

import org.testng.Assert;

import org.testng.annotations.Test;

import io.appium.java\_client.AppiumBy;

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

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

public class MiscellaneousActivity extends BaseTest{

@Test

public void MiscellaneousDemo() {

//click on preference

driver.findElement(AppiumBy.accessibilityId("Preference")).click();

//click on prference dependencies

driver.findElement(AppiumBy.accessibilityId("3. Preference dependencies")).click();

//click on wifi checkbox

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

//for device rotation for chnage landscape mode

DeviceRotation landscape=new DeviceRotation(0,0,90);

driver.rotate(landscape);

driver.findElement(By.xpath("(//android.widget.RelativeLayout)[2]")).click();

//assertion

String alerttitle=driver.findElement(By.id("android:id/alertTitle")).getText();

Assert.assertEquals(alerttitle, "WiFi settings");

//copy to clipboard paste it clipboard

driver.setClipboardText("sanjuktawifi");

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

//to press enter button

driver.pressKey(new KeyEvent(AndroidKey.ENTER));

driver.findElements(AppiumBy.className("android.widget.Button")).get(1).click();

//to press home button

driver.pressKey(new KeyEvent(AndroidKey.HOME));

//To press back button

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

}

}

MobileBrowserTest-

package sanjuktaproject.Appium;

import static org.testng.Assert.assertEquals;

import java.time.Duration;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.Keys;

import org.testng.Assert;

import org.testng.annotations.Test;

public class BobileBrowserTest extends BrowserBase\_Test {

@Test

public void browserdemo() throws InterruptedException {

/\*

\* driver.get("http://google.com"); Thread.sleep(2000);

\* System.out.println(driver.getTitle()); Thread.sleep(2000);

\* driver.findElement(By.name("q")).sendKeys("rahul shetty academy");

\* driver.findElement(By.name("q")).sendKeys(Keys.ENTER);

\*/

driver.get("https://rahulshettyacademy.com/angularAppdemo/");

driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));

driver.findElement(By.xpath("(//span[@class='navbar-toggler-icon'])")).click();

driver.findElement(By.cssSelector("a[routerlink\*='/products']")).click();

((JavascriptExecutor)driver).executeScript("window.scrollBy(0,1000)", ",");//scroll

String text=driver.findElement(By.cssSelector("a[href\*='products/3']")).getText();

System.out.println(text);

Assert.assertEquals(text, "Devops");

}

}

BrowserBase-

package sanjuktaproject.Appium;

import java.net.MalformedURLException;

import java.net.URL;

import java.time.Duration;

import org.testng.annotations.AfterClass;

import org.testng.annotations.BeforeClass;

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

import io.appium.java\_client.android.options.UiAutomator2Options;

public class BrowserBase\_Test {

AndroidDriver driver;

@BeforeClass

public void browserapp() throws MalformedURLException

{

// AppiumDriverLocalService service=new AppiumServiceBuilder().withAppiumJS(new File("C:\\Users\\ks22102\\AppData\\Roaming\\npm\\node\_modules\\appium\\build\\lib\\main.js"))

// .withIPAddress("127.0.0.1").usingPort(4723).build();

// service.start();

UiAutomator2Options options=new UiAutomator2Options();

options.setDeviceName("sanjukta");

options.setChromedriverExecutable("C:\\Users\\ks22102\\eclipseworkspace\\Appium\\src\\test\\java\\resources\\chromedriver.exe");

options.setCapability("browserName", "Chrome");

//options.setApp("C:\\Users\\ks22102\\eclipseworkspace\\Appium\\src\\test\\java\\resources\\General-Store.apk");

driver=new AndroidDriver (new URL("http://127.0.0.1:4723"), options);

driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(20));

}

@AfterClass

public void exitapp() {

//driver.quit();

}

}

IOS Automatio

1.Download the appium server

2.Install Xcode

3.Launch Iphone simulator

4.Setup appium maven jar with appium client jar

5.Desirecapabilities need to invoked the iphone

6.Understand how to inspect the object of iphone

7.Build the appium automation script

Content-

1.Dropdown, textbox, stepper, button , menu’s, alert handle

2.scroll down

3.swipe

4.longpress

5.Tab

6.Perform 3d touch event and slider

It is possible to automate in mac tab .