Guild line |Mobile Automation using Appium

**1.How to Set Up the Environment**

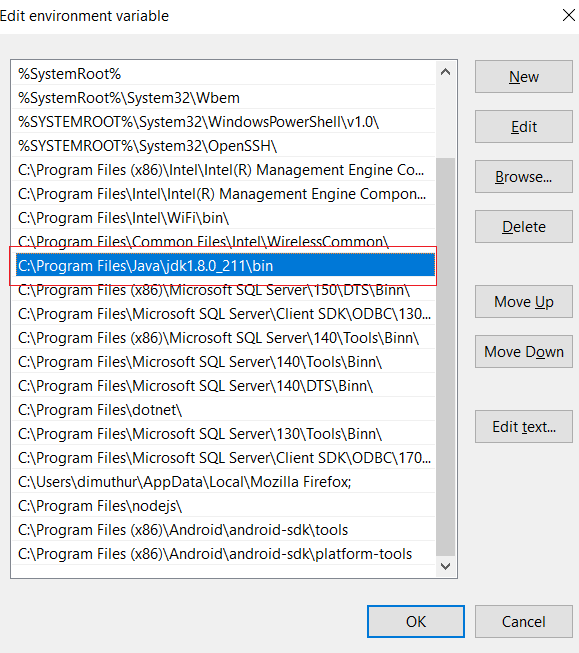
1.Donwnload & Install Java

<https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

2.Set the Java path

Navigates to below location & Set the path “C:\Program Files\Java\jdk1.8.0\_201\bin”

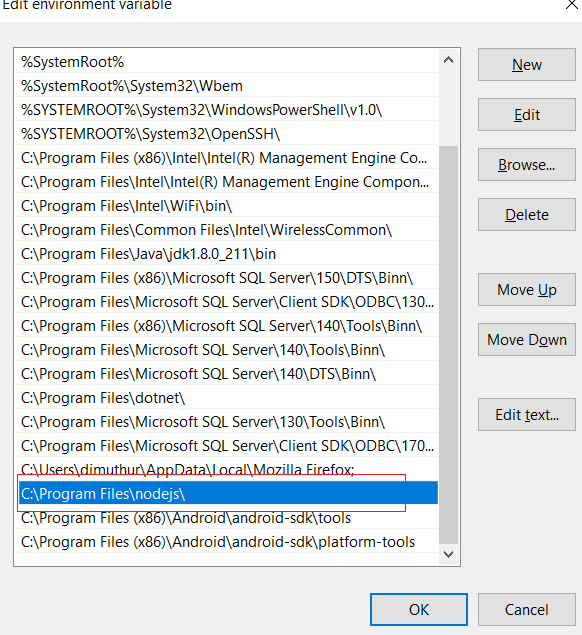
Advance System Properties - > Environment Variables->Path-Edit->New

****

3. Download & Install NodeJs

<https://nodejs.org/en/download/>

Note: By default Nodejs path is update in Environment variable, Please check the path & update , if not updated



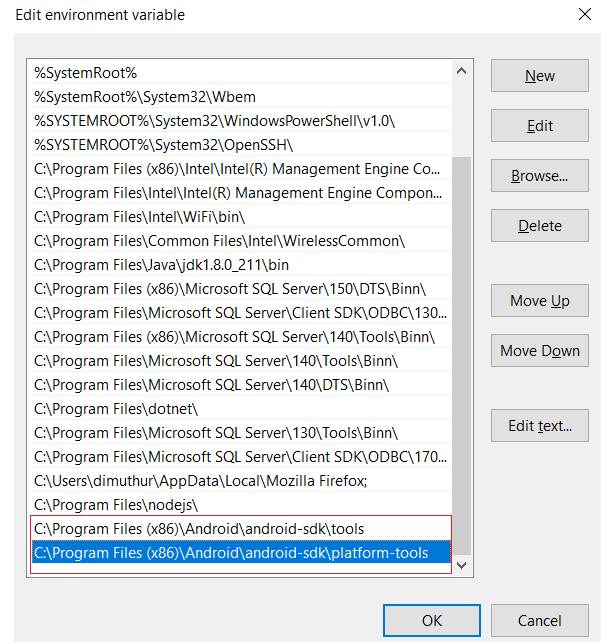
4. Download android studio (android-studio-ide-182.5314842-windows.exe) & Install

<https://developer.android.com/studio/?gclid=CjwKCAjwlPTmBRBoEiwAHqpvhUA9YZwa2GG3MFEtmEnMJzqmHUxwOsyfy9vBq-G95SA-HighblenNBoC2YcQAvD_BwE>

5.Set the below tow paths for Android in Edit environment variable section

C:\Program Files (x86)\Android\android-sdk\tools

C:\Program Files (x86)\Android\android-sdk\platform-tools



6.Download Appium & Install

<https://github.com/appium/appium-desktop/releases/tag/v1.11.0>

7.Download Eclipse

8.Install Test NG plugin

9.Download Selenium

10.Download Download Java-client java-client-7.0.0.jar

11. Download commons-lang3-3.1.jar, if you get an error when running test

**2.How to Create a Project**

* Open Eclipse
* Create a Java Project – File ->New->Java Project -> Give Project Name - Finish
* Add Selenium Jar fill – Right Click ->Build Path->Configure Build Path->Add External JARs->Select jars->Open->OK
* Add “java-client-7.0.0” – Follow the same way as above step
* Add “commons-lang3-3.1” - Follow the same way as above step
* Add Test NG library to project - Right Click ->Build Path->Configure Build Path->Add Library->Select TestNG->Next->Finish

**3.Sampel Code**

**package** demoTests;

**import** java.net.MalformedURLException;

**import** java.net.URL;

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

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

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

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

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

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

**public** **class** Demo

{

AndroidDriver driver;

@Test

**public** **void** setUp() **throws** MalformedURLException

{

DesiredCapabilities capabilities = **new** DesiredCapabilities();

capabilities.setCapability("deviceName", "Android Emulator");

capabilities.setCapability("platformName", "Android");

capabilities.setCapability("appPackage", "com.fqms");

capabilities.setCapability("appActivity","com.fqms.MainActivity");

driver = **new** AndroidDriver (**new** URL("http://127.0.0.1:4723/wd/hub"), capabilities);

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

driver.findElement(By.*xpath*("//android.widget.EditText[@text='Phone number']")).sendKeys("786776767");

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

driver.findElementByXPath("//android.widget.EditText[@text='Pin number']").sendKeys("345");

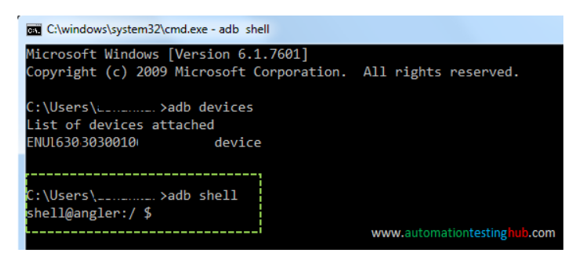
driver.quit();

}

}

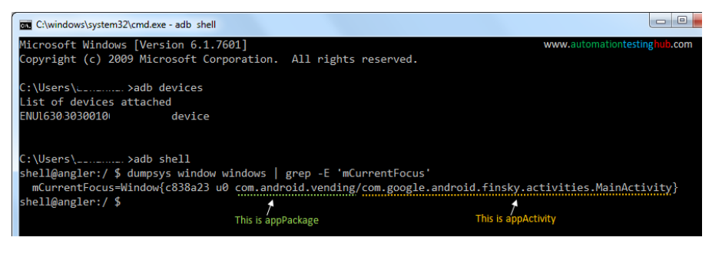
3.1 How to find App Package & App Activity

* Run ‘adb shell’ command. After running this command , the command prompt should look something like this



* Now in your mobile phone, open the app for which you want to find the App package & App activity

**Note:** Please make sure that you open the app before going to the next step, because command in the next step would provide the details only for the app which is currently in focus.

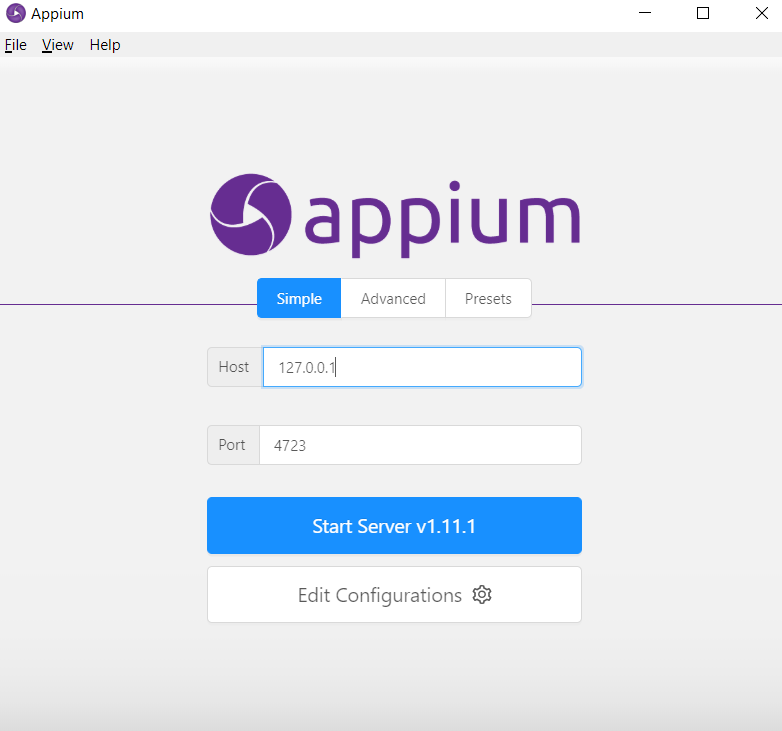
* Now run this command: dumpsys window windows | grep -E ‘mCurrentFocus’ in command prompt
* The above command would display the details of the app which is currently in focus. From that, you can figure out the appPackage and appActivity name as per the below image –

**3.How to Run a Test**

* Connect to device or open the emulator
* Open the Appium server



* Enter the Host as 127.0.0.1 & Port as 4723 & Click on Start Server v…



* Run the test using TestNG