## NODE\_HOME under system variable NODE\_HOME: C:\Program Files\nodejs

## 

## 

## System variable -> path->new ->C:\Program Files\nodejs\node\_modules\npm\bin

## 

## Requirements:

## 

## Download Android studio

## Download Android studio from the below site.

## <https://developer.android.com/studio/?gclid=EAIaIQobChMIzsK2--rH4wIVwbTtCh1mxAD2EAAYASAAEgLb3_D_BwE>

## and click on Download Android studio button

## 

## Accept the terms and conditions

## 

## Click on download Android studio button Windows.

## 

## 

## 

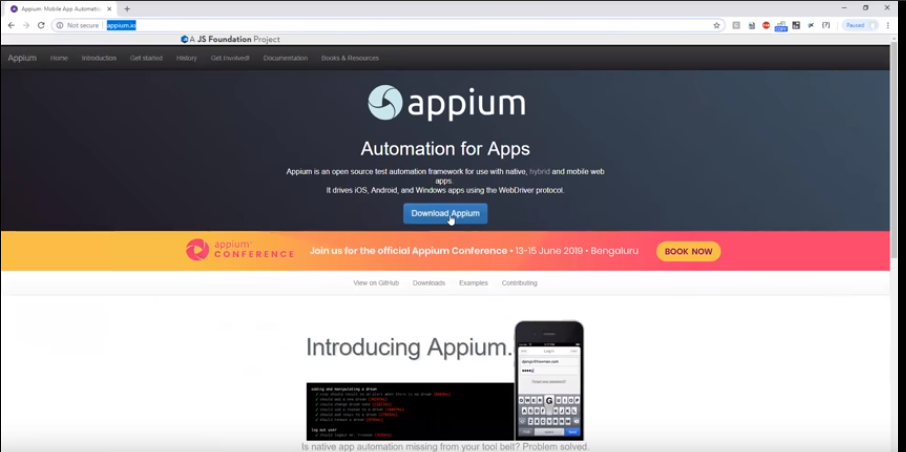
## 

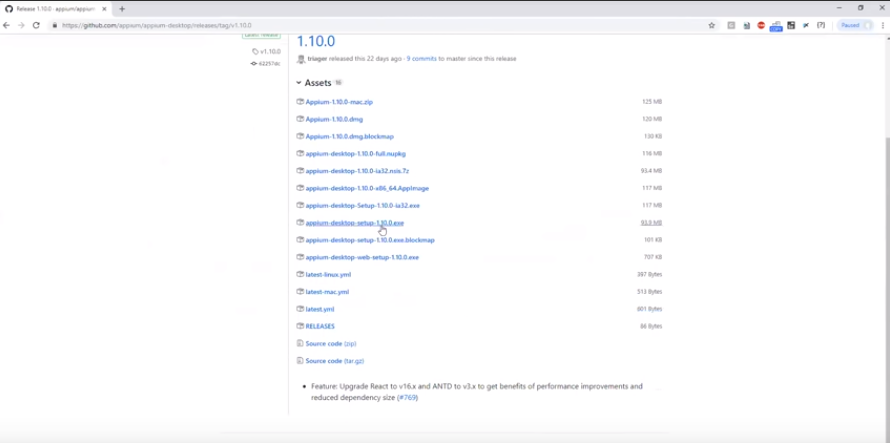
## 

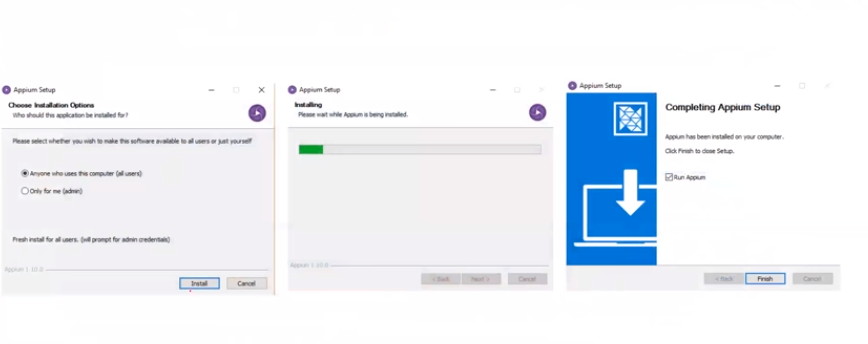
## 

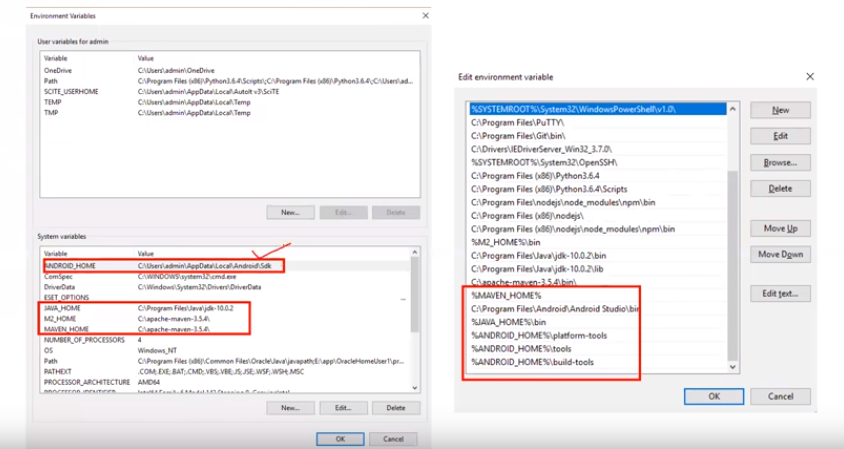
**Download appium:**

**Go to appium.io**



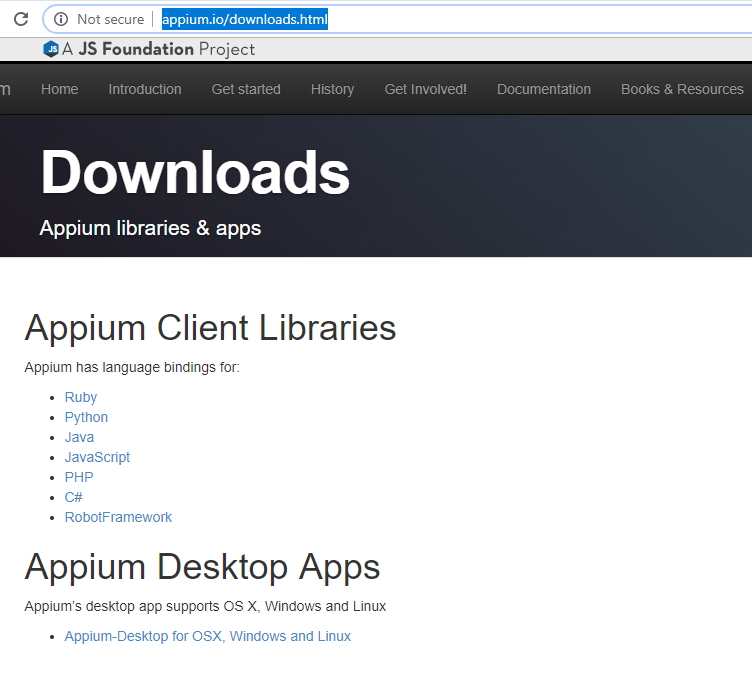








Go to the <http://appium.io/downloads.html>

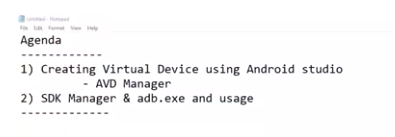


Click on java.

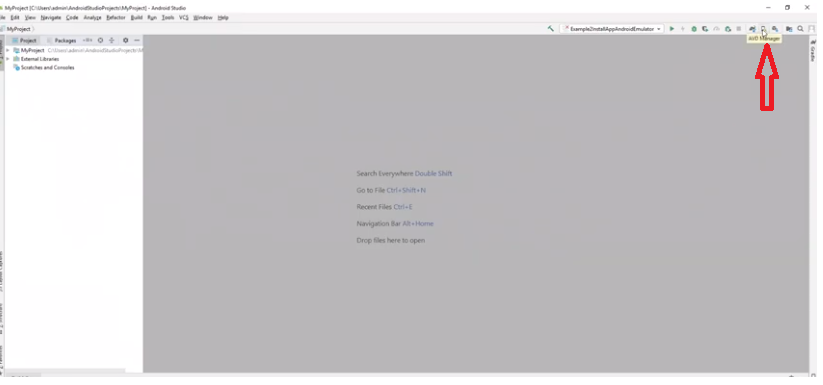


Click on download icon.

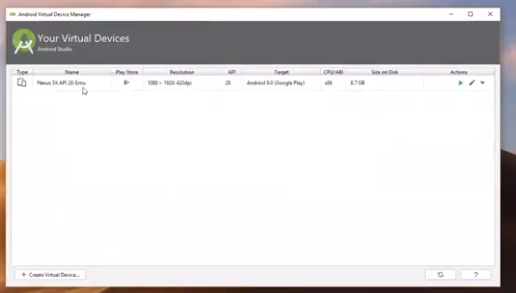
Creating avd and vt.



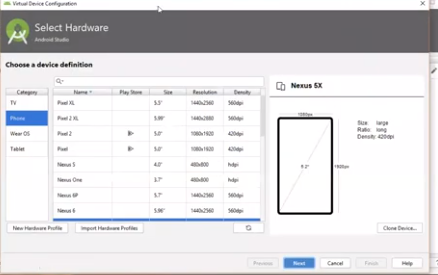
Start the android studio.



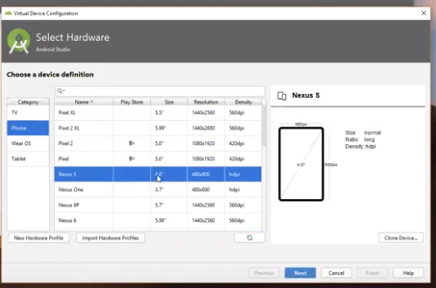
Click on avd manager icon at the top of the window.

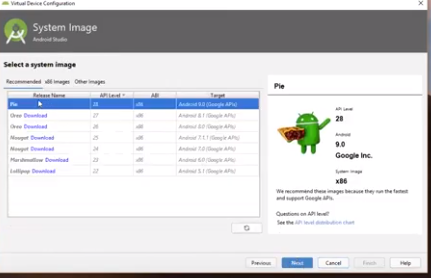


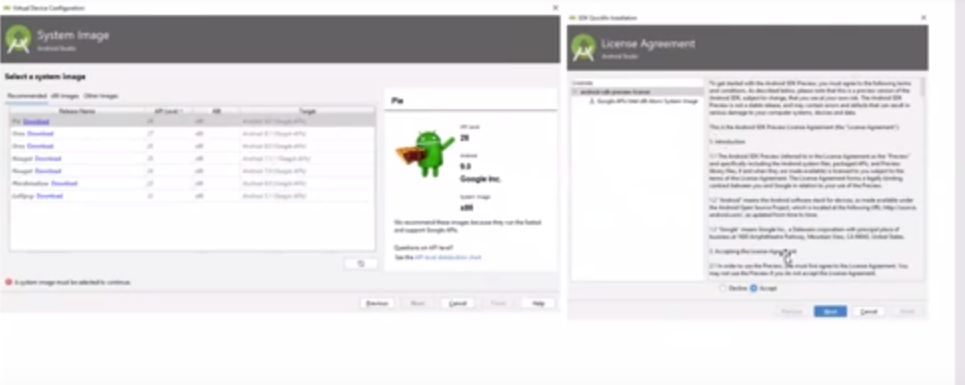
Click on “create virtual Device button”.

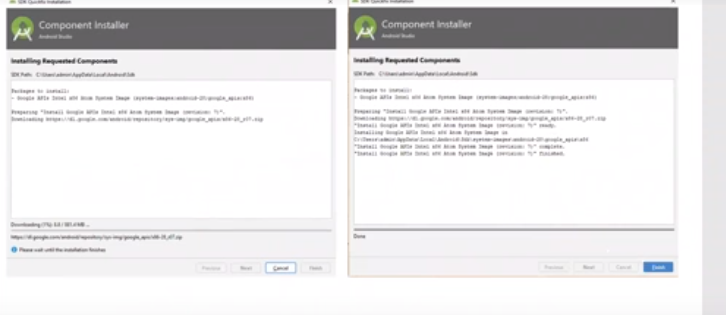


Select device and click on next button.

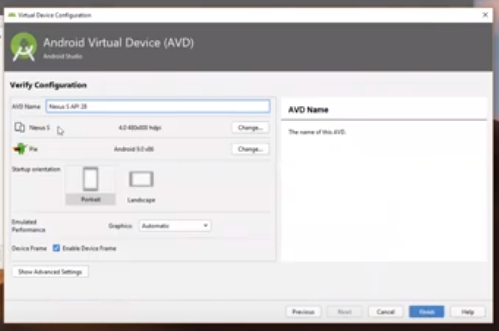








Click on next

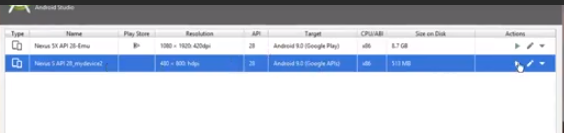


Click on finish button

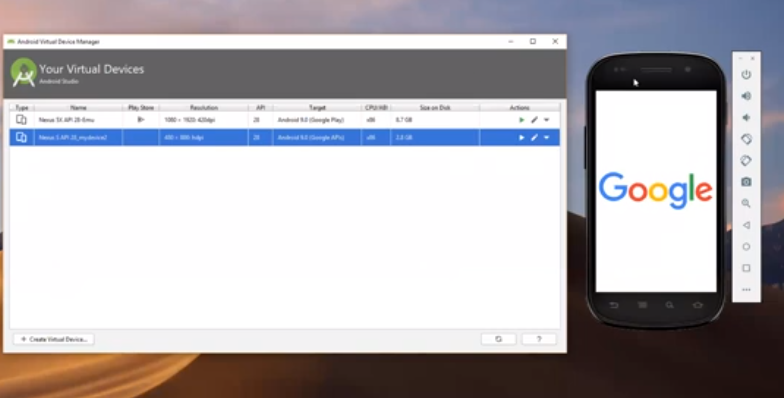
The new device is created successfully as below screen



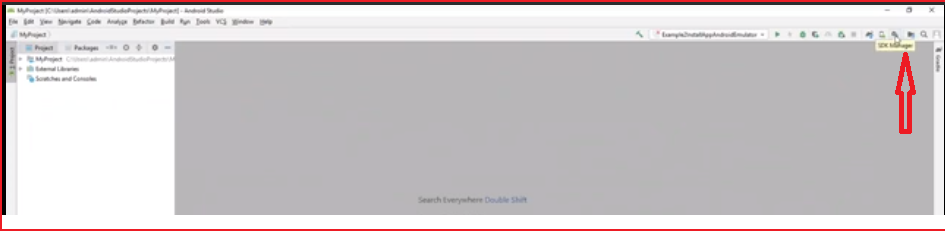
To turn on the device click on run button.



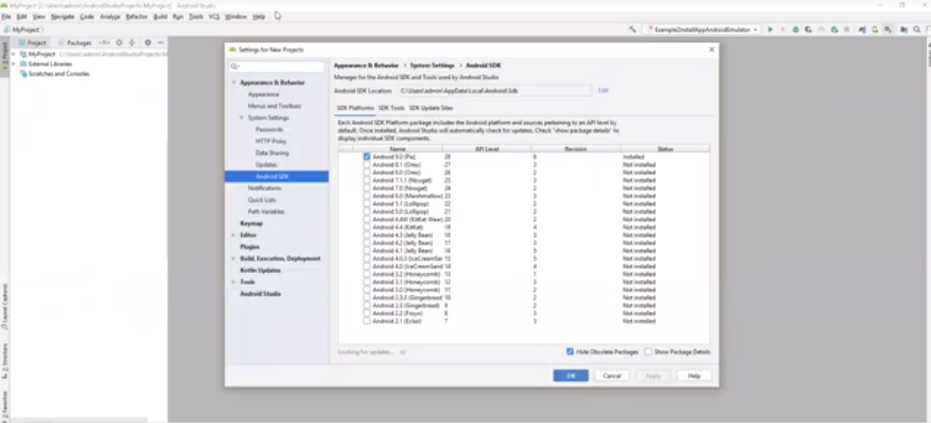
Now the device is opened.



Click on sdk manager at the top of the android window



The sdk popup is opened with list of versions



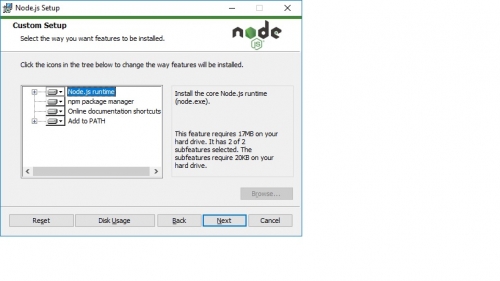
## Install Node & NPM

Download latest **node** MSI from [nodejs.org](https://nodejs.org/en/download/) and install node. This will also install node’s **npm** tool (node package manager) which we will use to install Appium Server.

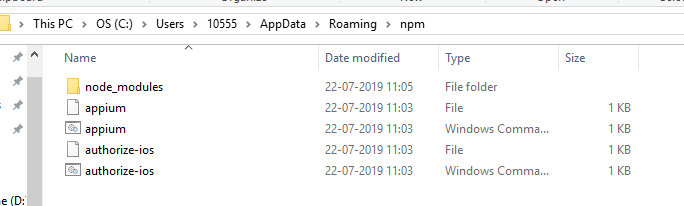
Appium Server is built on node.js so it is mandatory that we install it, before we install Appium.



Ensure you select to install the npm tools and the system paths:



Once completed, the npm and nodejs paths should be in your PATH environment variable (done by installer)  
**System System Path: C:\Program Files\node.js  
User System Path: C:\Users\10555\AppData\Roaming\npm**

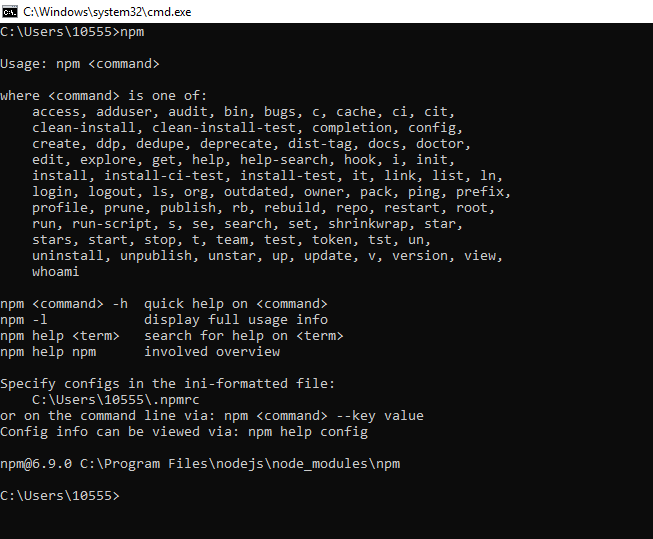


Install Appium using npm. Ref link below.

<https://www.edgewordstraining.co.uk/2017/07/05/install-appium-server-windows/>

the node js is install in the C:\Program Files\nodejs.

Go to command prompt and enter” npm” to check the nodejs is installed or not.



Install Appium using node’s npm package manager.

Go to command prompt and type the below command and click on enter. Observe the Appium is downloading process.

**npm install -g appium**

To uninstall or update Appium you can use the npm package manager, for example:

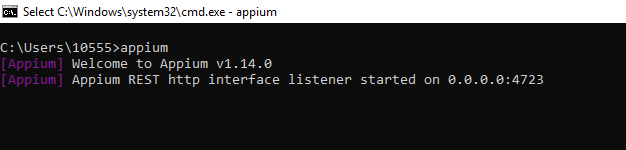
npm uninstall -g appium

npm install -g appium@1.6

Where the @ can be used to choose a particular version to install

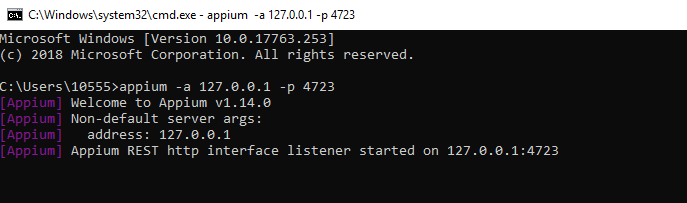
Start Appium:

to start the Appium go to command prompt. And enter Appium and click on enter button. the Appium is started



to start the server using appium -a 127.0.0.1 -p 4723

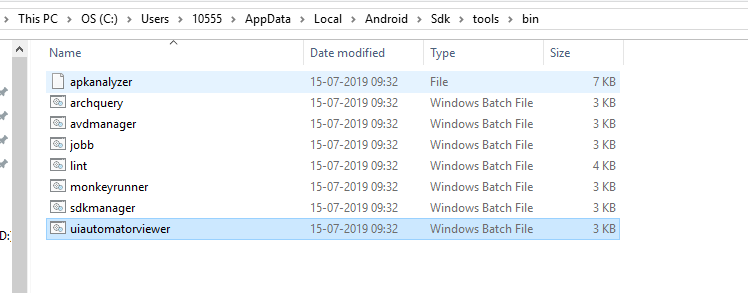
where **-a** is the IP address (localhost in this case) and **-p** is the port number



**UIautomater viewer**

Working with ui automater viewer for identifying the objects.

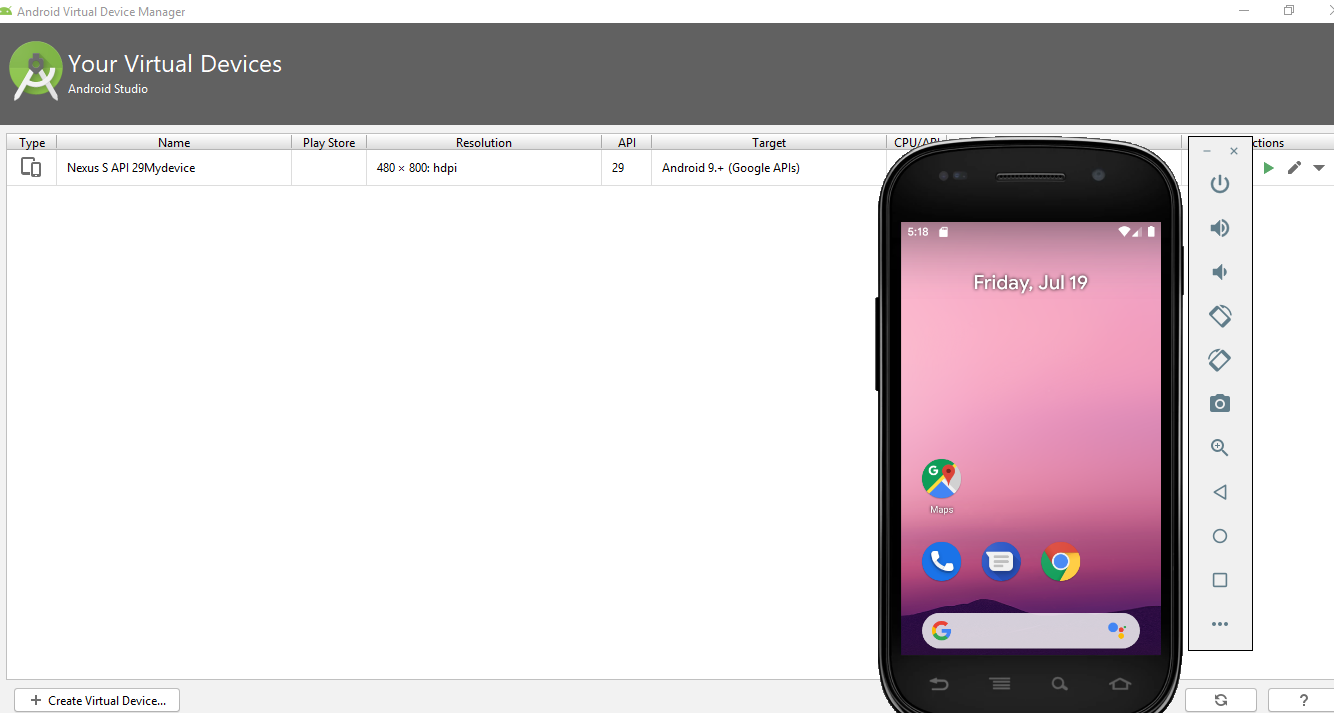
Go to the C:\Users\10555\AppData\Local\Android\Sdk\tools\bin path .

click on “uiautomaterviewer” batch file.

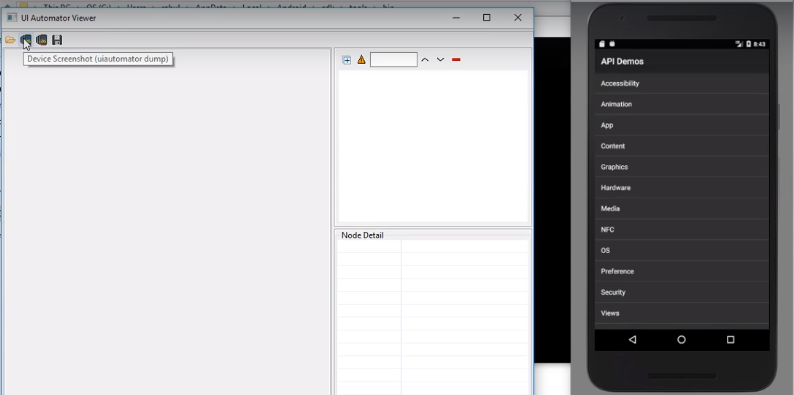
The command window is opened after some time the ui autmaterviewer is opened as below.



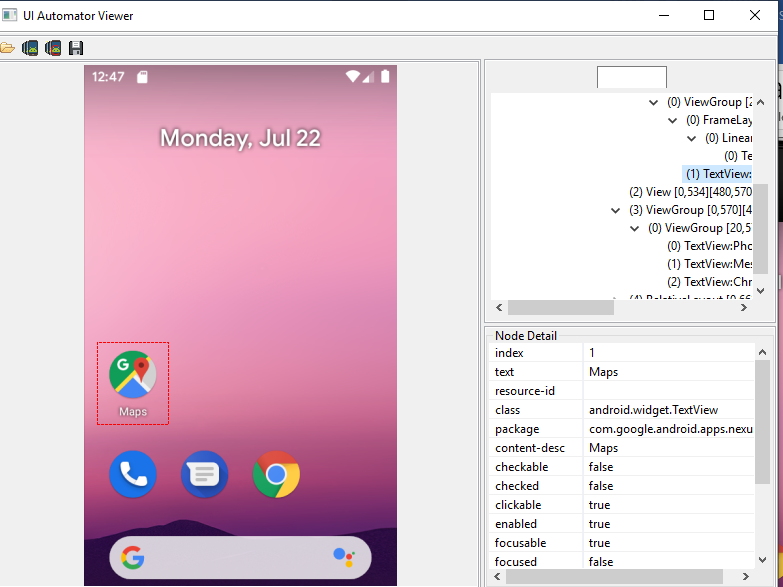
The device is opened.

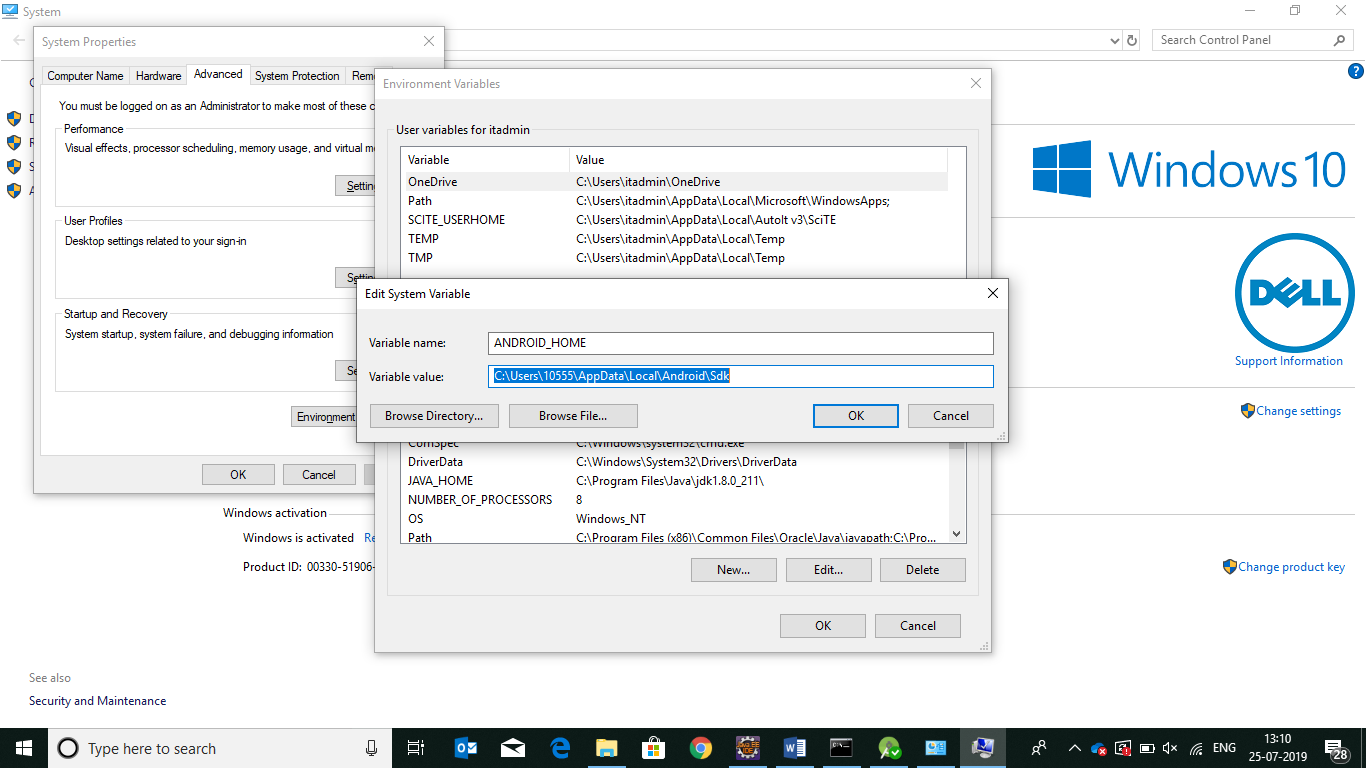


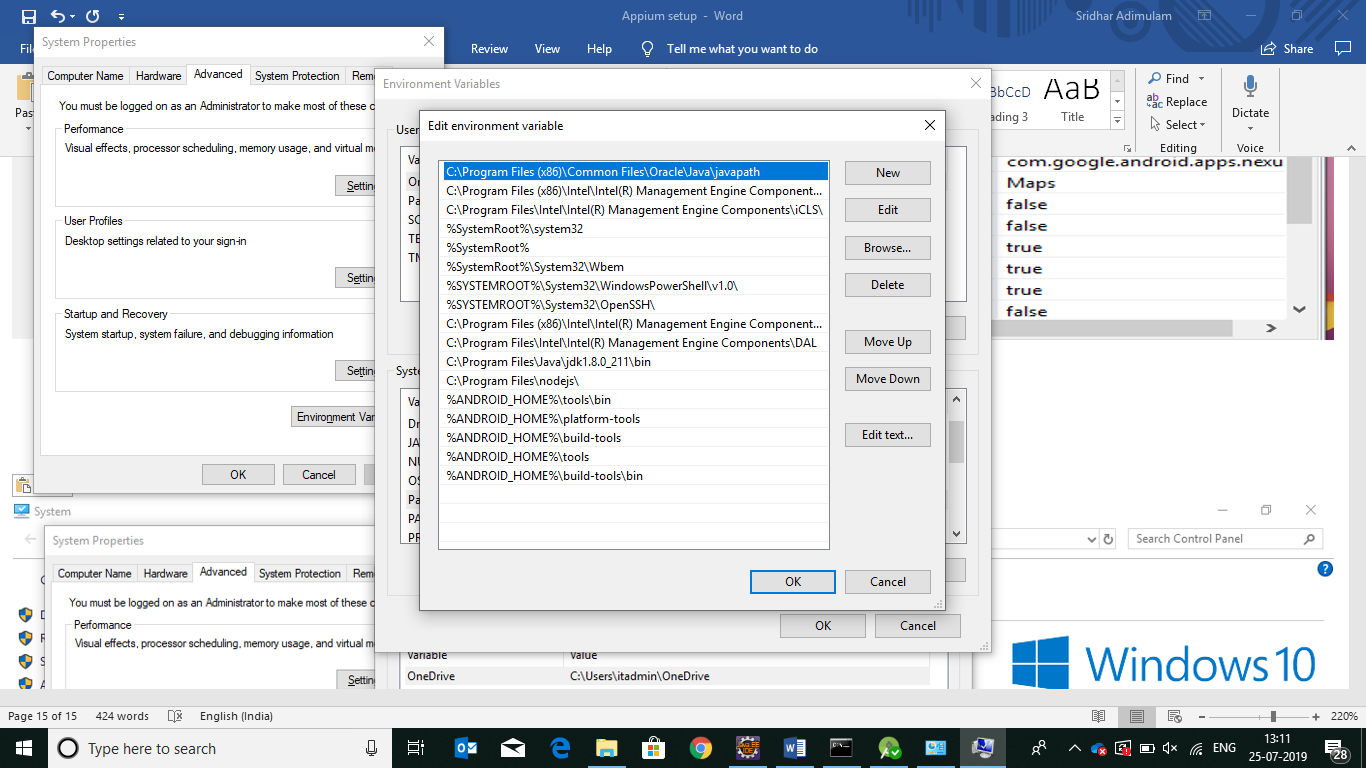
Click on the link for taking screen shot.

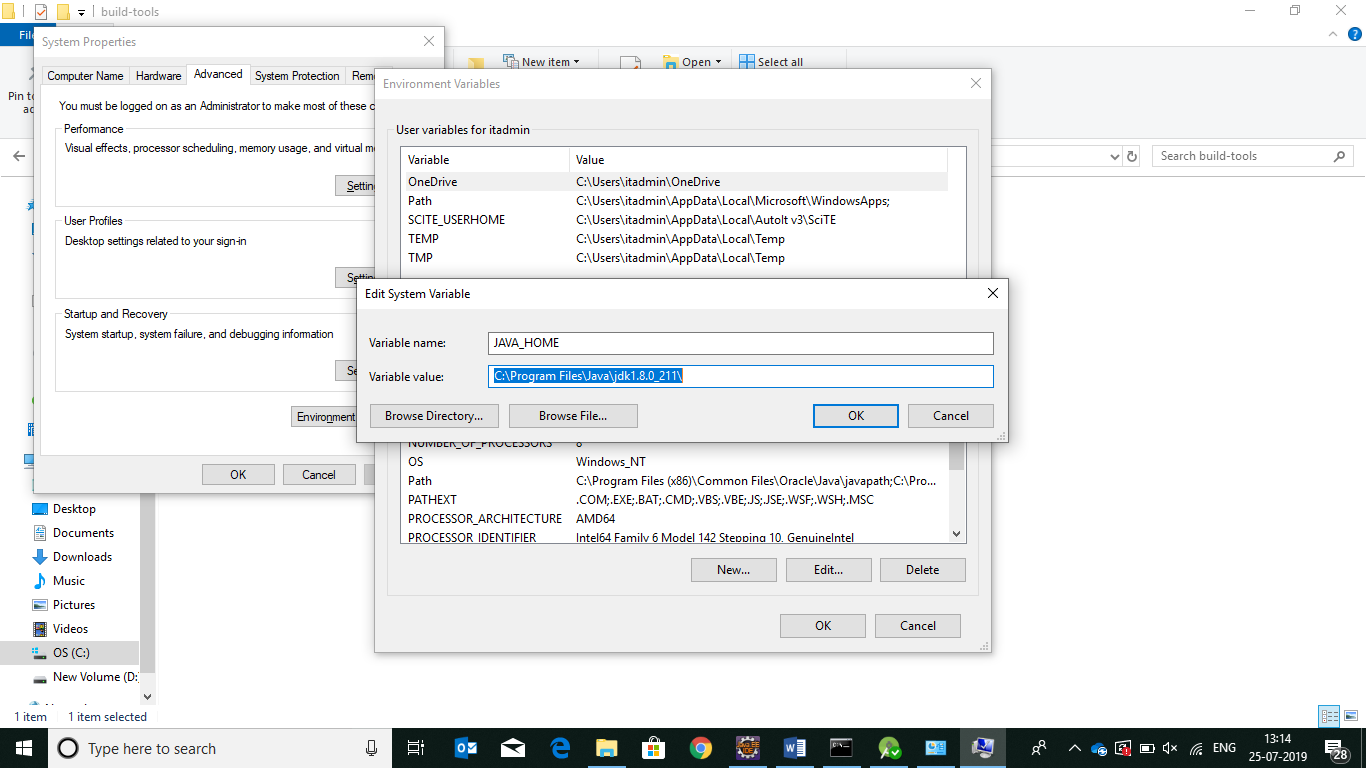


The screenshot and its properties is stored in to the UI automater .





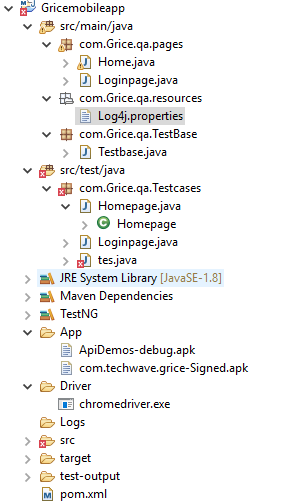




**Create project in Eclipse for Appium:**

1. Create maven project in the Eclipse.

the below is the structure of the maven project.



The pom.xml file is below.

<project xmlns=*"http://maven.apache.org/POM/4.0.0"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"*>

<modelVersion>4.0.0</modelVersion>

<groupId>Gricemobileapp</groupId>

<artifactId>Gricemobileapp</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<name>Gricemobileapp</name>

<url>http://maven.apache.org</url>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.compiler.target>1.8</maven.compiler.target>

<maven.compiler.source>1.8</maven.compiler.source>

</properties>

<dependencies>

<!-- https://mvnrepository.com/artifact/io.appium/java-client -->

<dependency>

<groupId>io.appium</groupId>

<artifactId>java-client</artifactId>

<version>7.0.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java ->

<dependency>

<groupId>org.seleniumhq.selenium</groupId>

<artifactId>selenium-java</artifactId>

<version>3.141.59</version>

</dependency>

<!--https://mvnrepository.com/artifact/com.google.code.gson/gson -->

<dependency>

<groupId>com.google.code.gson</groupId>

<artifactId>gson</artifactId>

<version>2.8.4</version>

</dependency>

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

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>6.14.3</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/com.google.code.gson/gson -->

<dependency>

<groupId>com.google.code.gson</groupId>

<artifactId>gson</artifactId>

<version>2.8.5</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.commons/commons-lang3 ->

<dependency>

<groupId>org.apache.commons</groupId>

<artifactId>commons-lang3</artifactId>

<version>3.1</version>

</dependency>

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>6.11</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

<version>3.16</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-examples</artifactId>

<version>3.16</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-excelant</artifactId>

<version>3.16</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml</artifactId>

<version>3.16</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml-schemas</artifactId>

<version>3.16</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-scratchpad</artifactId>

<version>3.16</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>ooxml-schemas</artifactId>

<version>1.1</version>

</dependency>

<dependency>

<groupId>commons-codec</groupId>

<artifactId>commons-codec</artifactId>

<version>1.10</version>

</dependency>

<dependency>

<groupId>org.apache.commons</groupId>

<artifactId>commons-collections4</artifactId>

<version>4.1</version>

</dependency>

<dependency>

<groupId>commons-logging</groupId>

<artifactId>commons-logging</artifactId>

<version>1.2</version>

</dependency>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.12</version>

</dependency>

<dependency>

<groupId>log4j</groupId>

<artifactId>log4j</artifactId>

<version>1.2.17</version>

</dependency>

<dependency>

<groupId>com.aventstack</groupId>

<artifactId>extentreports</artifactId>

<version>3.0.6</version>

</dependency>

</dependencies>

</project>

Log4j.properties:

#Define the root logger with appender

log.dir=Logs/

log4j.rootLogger=INFO, FILE,stdout,HTML

# Define file Appender

log4j.appender.FILE=org.apache.log4j.FileAppender

log4j.appender.FILE.File=Logs/Framework.log

log4j.appender.FILE.Append=true

log4j.appender.FILE.layout=org.apache.log4j.PatternLayout

log4j.appender.FILE.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss}%-5p %c**{1}**:%L - %m%n

log4j.appender.file.MaxFileSize=5MB

log4j.appender.file.MaxBackupIndex=5

log4j.appender.HTML=org.apache.log4j.FileAppender

log4j.appender.HTML.File=Logs/application.html

log4j.appender.HTML.layout=org.apache.log4j.HTMLLayout

log4j.appender.HTML.layout.Title=Application log

log4j.appender.HTML.layout.LocationInfo=true

log4j.appender.HTML.Append=true

#Direct log messages to stdout

log4j.appender.stdout=org.apache.log4j.ConsoleAppender

log4j.appender.stdout.Target=System.out

log4j.appender.stdout.Append=true

log4j.appender.stdout.layout=org.apache.log4j.PatternLayout

log4j.appender.stdout.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c**{1}**:%L - %m%n

Testbase class:

package com.Grice.qa.TestBase;

import java.io.File;

import java.net.MalformedURLException;

import java.net.URL;

import java.util.Properties;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.remote.DesiredCapabilities;

import org.testng.ITestResult;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.AfterSuite;

import org.testng.annotations.BeforeSuite;

import com.aventstack.extentreports.ExtentReports;

import com.aventstack.extentreports.ExtentTest;

import com.aventstack.extentreports.Status;

import com.aventstack.extentreports.markuputils.ExtentColor;

import com.aventstack.extentreports.markuputils.MarkupHelper;

import com.aventstack.extentreports.reporter.ExtentHtmlReporter;

import com.aventstack.extentreports.reporter.configuration.ChartLocation;

import com.aventstack.extentreports.reporter.configuration.Theme;

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

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

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

public class Testbase {

public static ExtentHtmlReporter htmlReporter;

public static ExtentReports report;

public static ExtentTest test;

public static Properties prop;

//public Pageobjects p;

public WebDriver dr;

public static AndroidDriver<AndroidElement>driver;

//public static RemoteWebDriver driver;

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

//File fs=new File("D:\\seleniumworkspace\\Gricemobileapp\\ApiDemos-debug.apk");

File fs=new File("D:\\seleniumworkspace\\Gricemobileapp\\App\\com.techwave.grice-Signed.apk");

DesiredCapabilities cap=new DesiredCapabilities();

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

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

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

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

System.out.println("launch sucessfully");

//driver = new RemoteWebDriver(new URL("http://127.0.0.1:4723/wd/hub"), cap);

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS); driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS); driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();

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

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

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

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

}

@BeforeSuite

public void setUp() throws Exception

{

htmlReporter = new ExtentHtmlReporter(System.getProperty("user.dir") +"/test-output/GriceReport.html");

report = new ExtentReports();

report.attachReporter(htmlReporter);

htmlReporter.config().setChartVisibilityOnOpen(true);

htmlReporter.config().setDocumentTitle("Techwave.net Reports");

htmlReporter.config().setReportName("Littleword Report");

htmlReporter.config().setTestViewChartLocation(ChartLocation.TOP);

htmlReporter.config().setTheme(Theme.DARK);

}

@AfterMethod

public void getResult(ITestResult result)

{

if(result.getStatus() == ITestResult.FAILURE)

{

test.log(Status.FAIL, MarkupHelper.createLabel(result.getName()+" Test case FAILED due to below issues:", ExtentColor.RED));

test.fail(result.getThrowable());

}

else if(result.getStatus() == ITestResult.SUCCESS)

{

//String screenShotPath=GetScreenShot.capture(driver,"ScreenshotName");

test.log(Status.PASS, MarkupHelper.createLabel(result.getName()+" Test Case PASSED", ExtentColor.GREEN));

}

else

{

test.log(Status.SKIP, MarkupHelper.createLabel(result.getName()+" Test Case SKIPPED", ExtentColor.ORANGE));

test.skip(result.getThrowable());

}

}

@AfterSuite

public void teardown() throws InterruptedException{

//dr.quit();

report.flush();

System.setProperty("webdriver.chrome.driver", System.getProperty("user.dir")+"\\Driver\\chromedriver.exe");

dr=new ChromeDriver();

dr.manage().window().maximize();

dr.get(System.getProperty("user.dir")+"\\test-output\\GriceReport.html");

//driver.get("D:\seleniumworkspace\Gricemobileapp\test-output\GriceReport.html

Thread.sleep(2000);

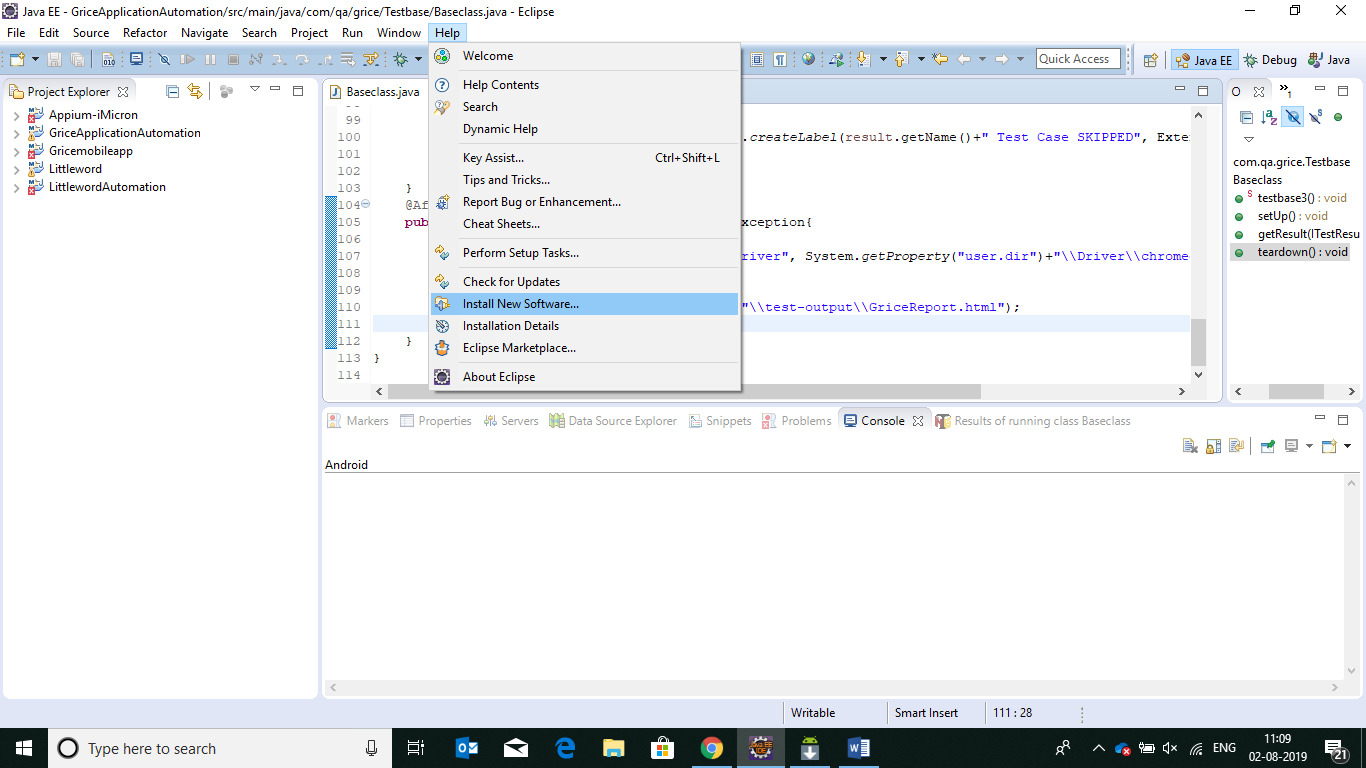
}

}

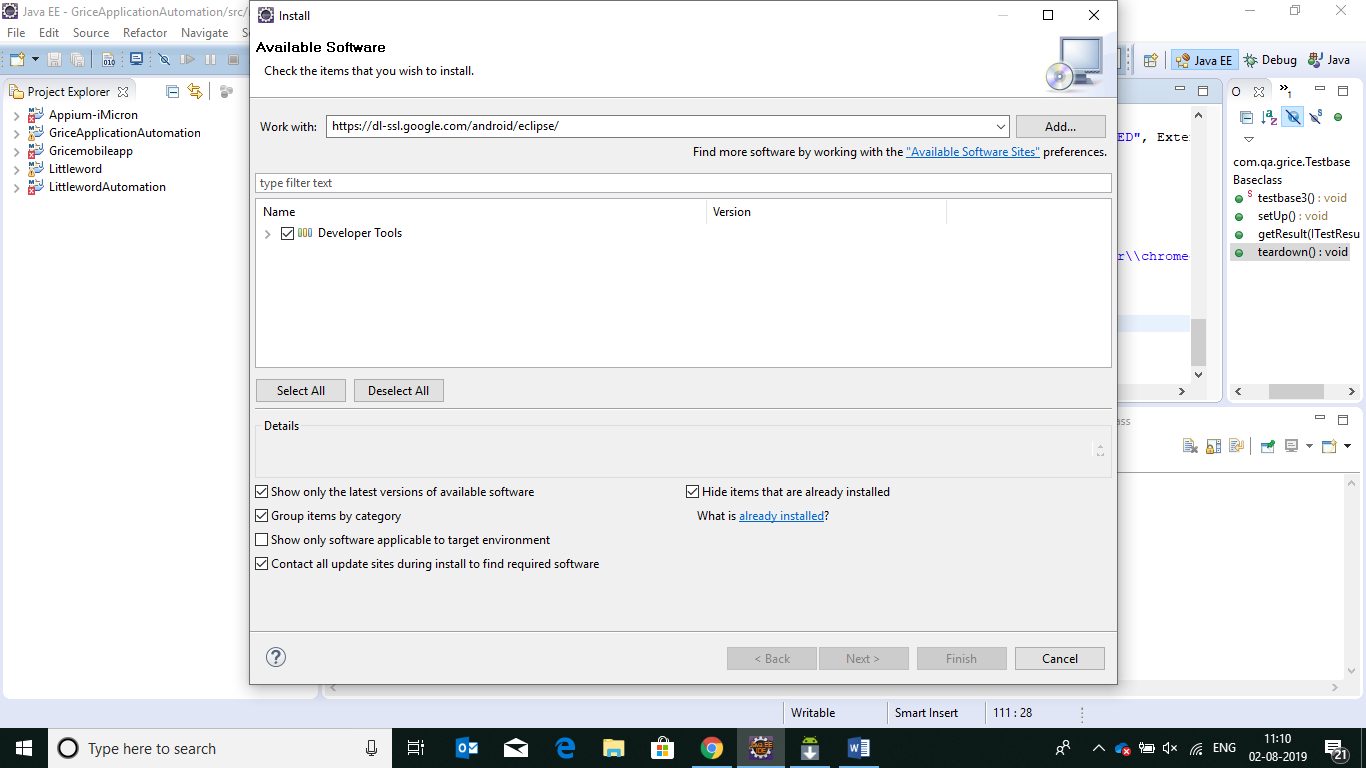
Adding sdk to eclipse:

1.Launch eclipse

2.help->Install new software

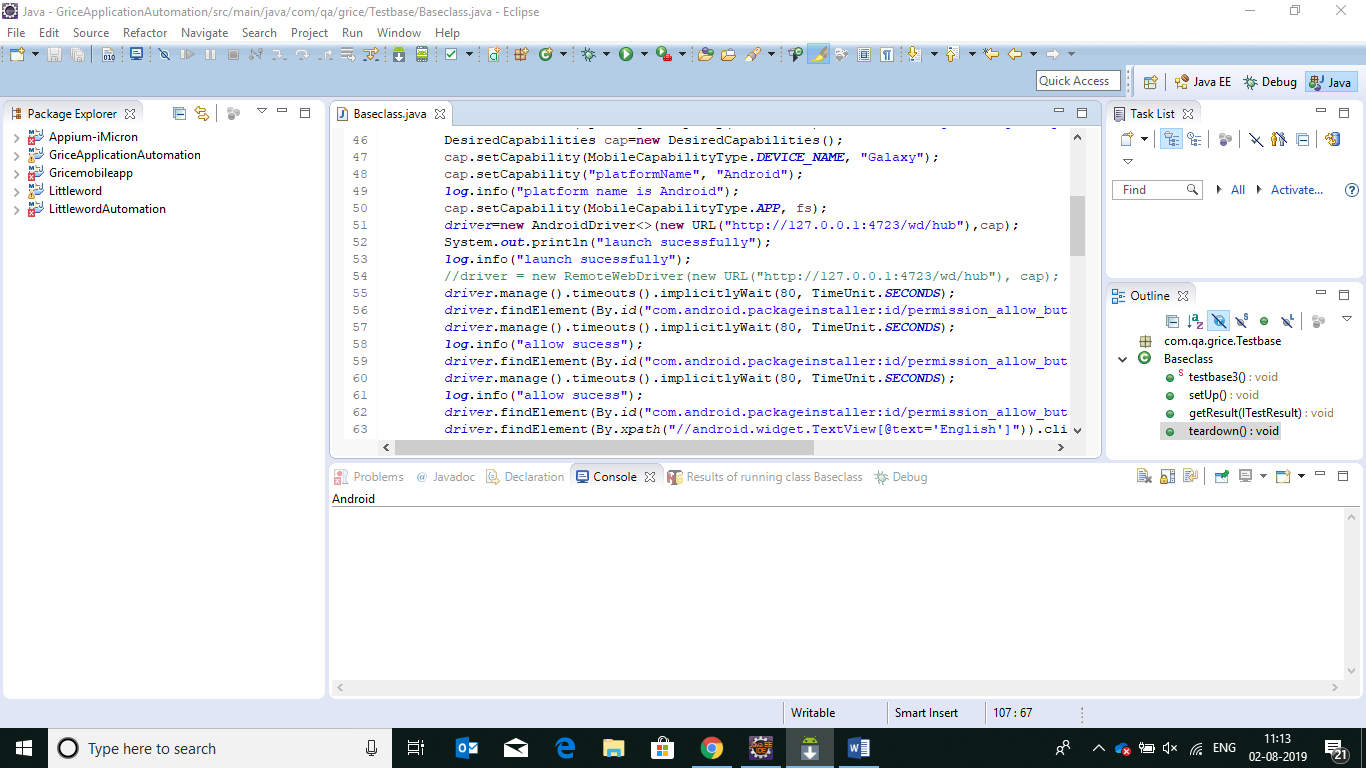


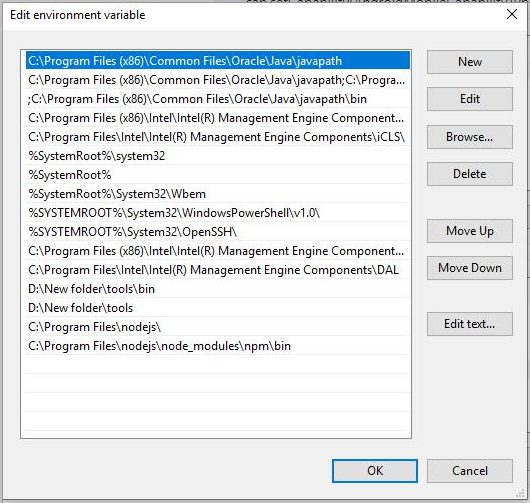
3.enter “https://dl-ssl.google.com/android/eclipse/”

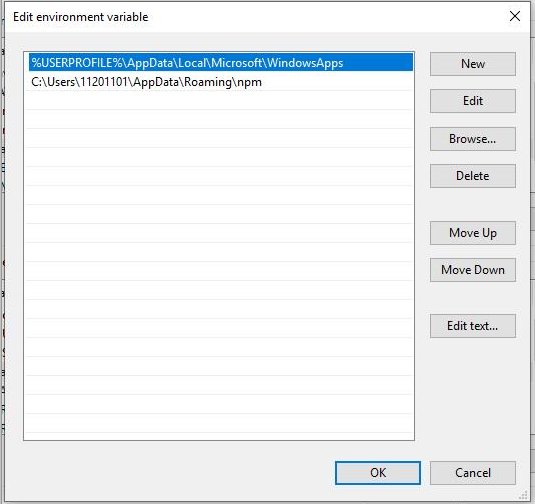


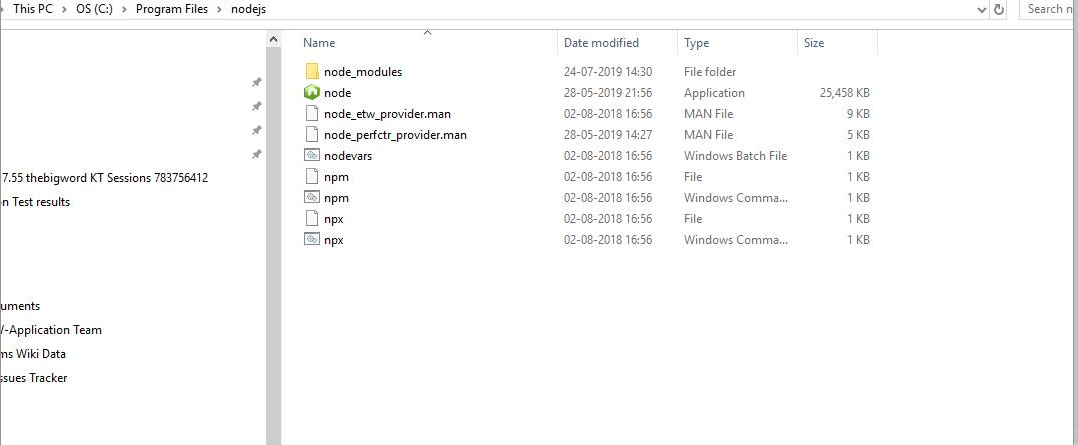
4.click on finish

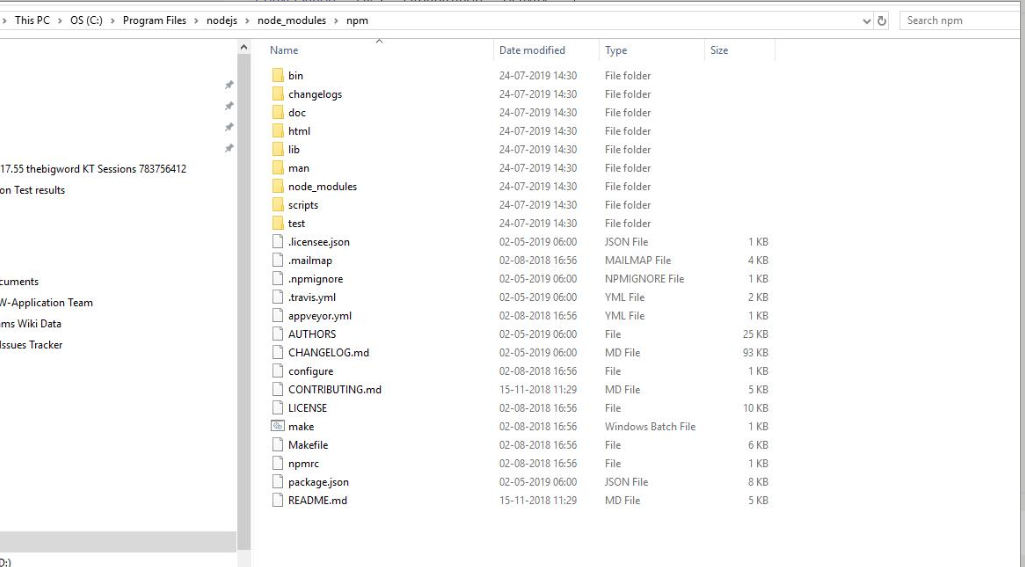
5.observe the sdk is displaying in the top navigation bar.



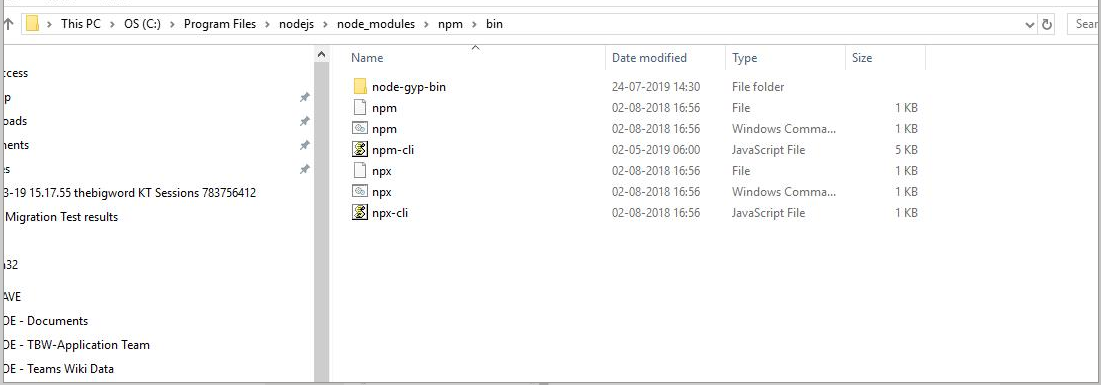


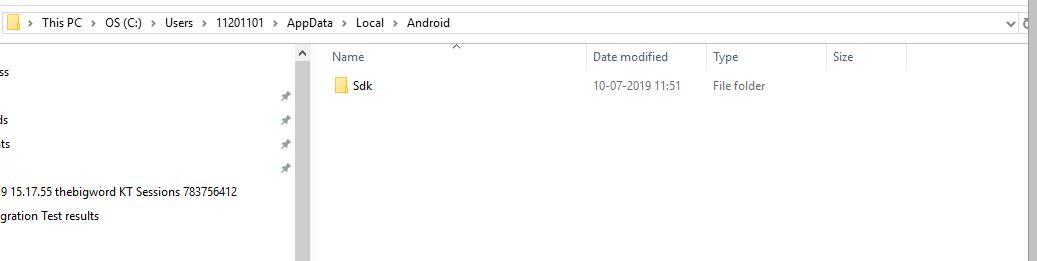


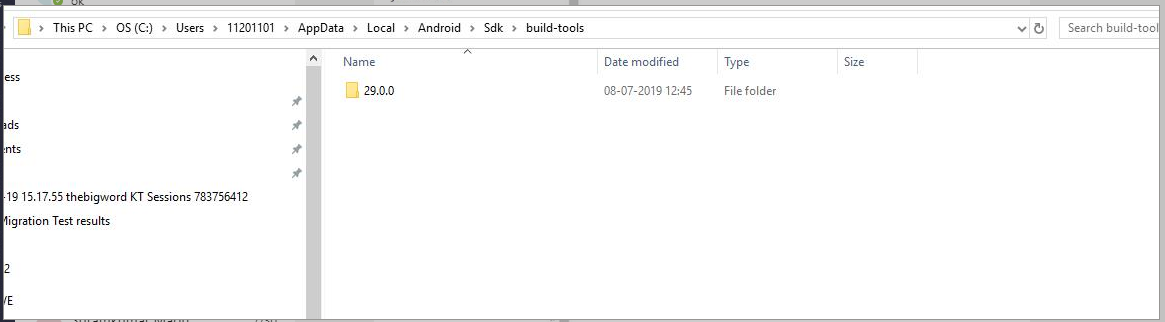


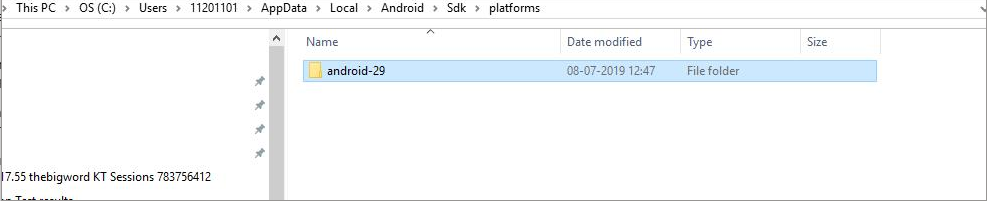


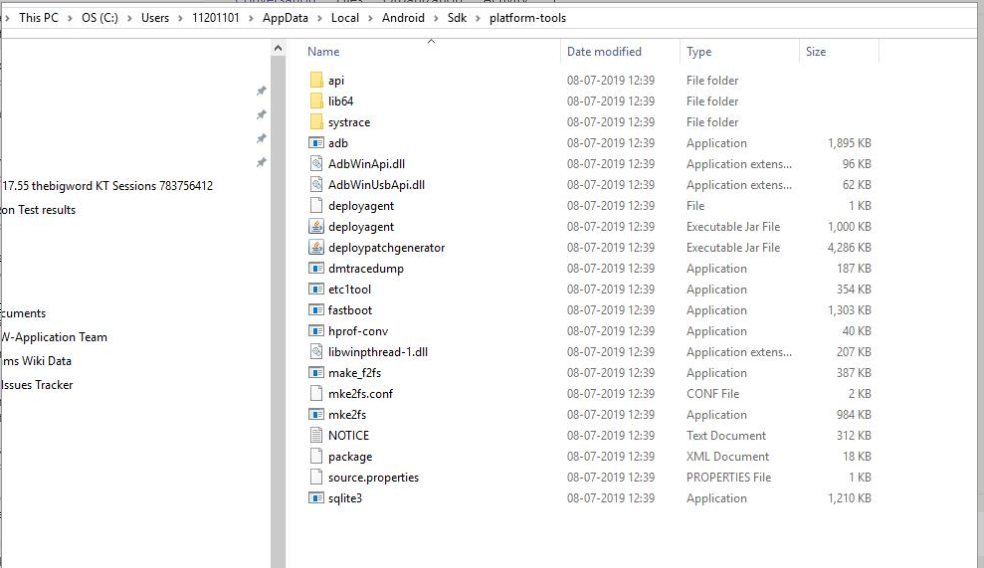


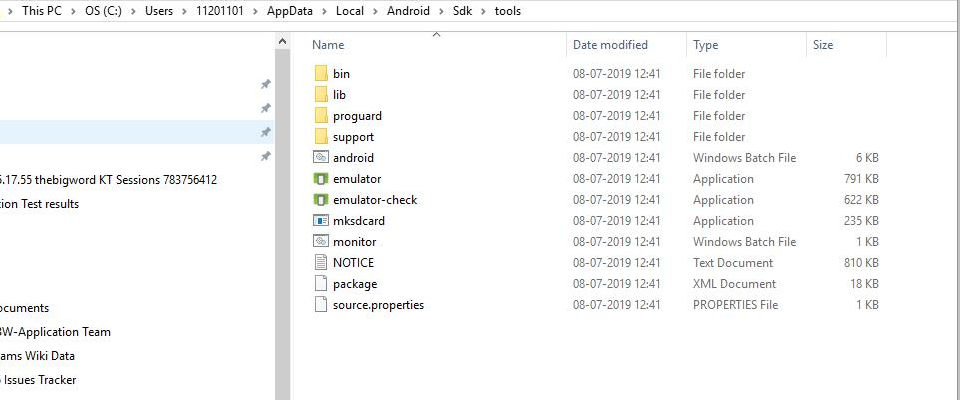


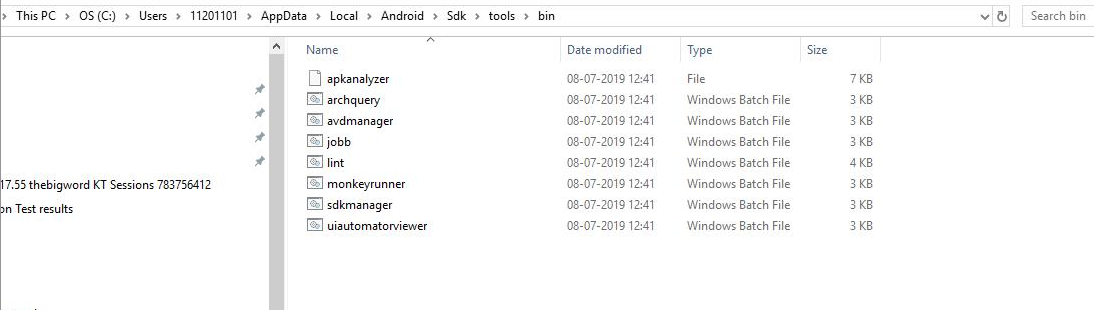


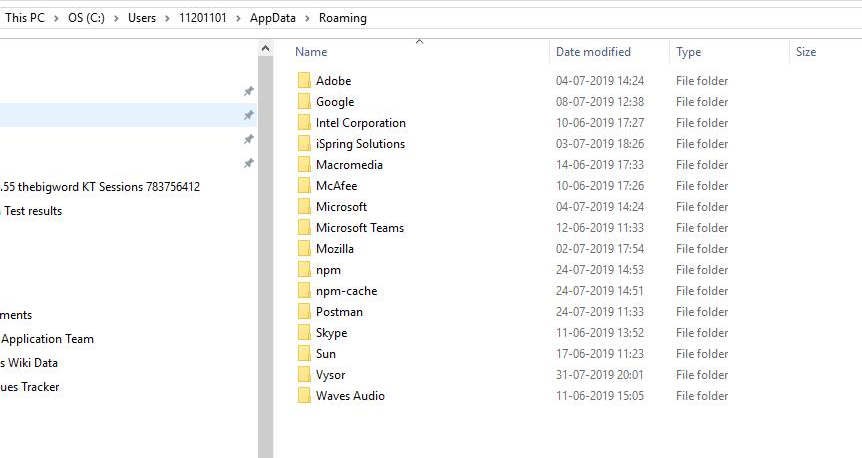


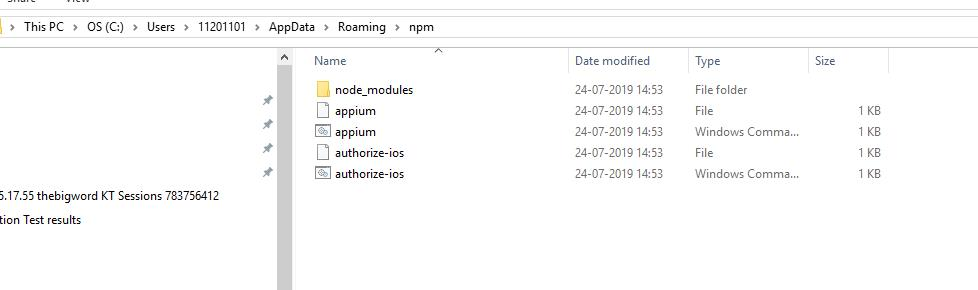


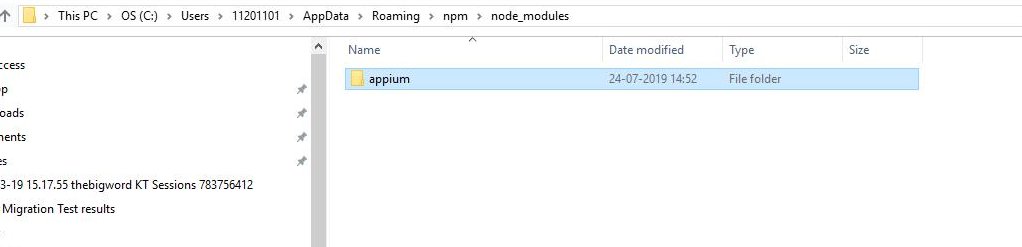












**Log4j.properties**

#Define the root logger with appender

log.dir=Logs/

log4j.rootLogger=INFO, FILE,stdout,HTML

# Define file Appender

log4j.appender.FILE=org.apache.log4j.FileAppender

log4j.appender.FILE.File=Logs/Framework.log

log4j.appender.FILE.Append=true

log4j.appender.FILE.layout=org.apache.log4j.PatternLayout

log4j.appender.FILE.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss}%-5p %c{1}:%L - %m%n

log4j.appender.file.MaxFileSize=5MB

log4j.appender.file.MaxBackupIndex=5

log4j.appender.HTML=org.apache.log4j.FileAppender

log4j.appender.HTML.File=Logs/application.html

log4j.appender.HTML.layout=org.apache.log4j.HTMLLayout

log4j.appender.HTML.layout.Title=Application log

log4j.appender.HTML.layout.LocationInfo=true

log4j.appender.HTML.Append=true

#Direct log messages to stdout

log4j.appender.stdout=org.apache.log4j.ConsoleAppender

log4j.appender.stdout.Target=System.out

log4j.appender.stdout.Append=true

log4j.appender.stdout.layout=org.apache.log4j.PatternLayout

log4j.appender.stdout.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n

**Base class:**

package com.Grice.qa.TestBase;

import java.io.File;

import java.net.MalformedURLException;

import java.net.URL;

import java.util.Properties;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.remote.DesiredCapabilities;

import org.testng.ITestResult;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.AfterSuite;

import org.testng.annotations.BeforeSuite;

import com.aventstack.extentreports.ExtentReports;

import com.aventstack.extentreports.ExtentTest;

import com.aventstack.extentreports.Status;

import com.aventstack.extentreports.markuputils.ExtentColor;

import com.aventstack.extentreports.markuputils.MarkupHelper;

import com.aventstack.extentreports.reporter.ExtentHtmlReporter;

import com.aventstack.extentreports.reporter.configuration.ChartLocation;

import com.aventstack.extentreports.reporter.configuration.Theme;

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

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

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

public class Testbase {

public static ExtentHtmlReporter htmlReporter;

public static ExtentReports report;

public static ExtentTest test;

public static Properties prop;

public WebDriver dr;

public static AndroidDriver<AndroidElement>driver;

//public static RemoteWebDriver driver;

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

File fs=new File("D:\\seleniumworkspace\\Gricemobileapp\\com.techwave.grice-Signed.apk");

DesiredCapabilities cap=new DesiredCapabilities();

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

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

cap.setCapability(MobileCapabilityType.APP, fs);

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

System.out.println("launch sucessfully");

//driver = new RemoteWebDriver(new URL("http://127.0.0.1:4723/wd/hub"), cap);

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();

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

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

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

driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);

}

@BeforeSuite

public void setUp() throws Exception

{

htmlReporter = new ExtentHtmlReporter(System.getProperty("user.dir") +"/test-output/GriceReport.html");

report = new ExtentReports();

report.attachReporter(htmlReporter);

htmlReporter.config().setChartVisibilityOnOpen(true);

htmlReporter.config().setDocumentTitle("Techwave.net Reports");

htmlReporter.config().setReportName("Littleword Report");

htmlReporter.config().setTestViewChartLocation(ChartLocation.TOP);

htmlReporter.config().setTheme(Theme.DARK);

}

@AfterMethod

public void getResult(ITestResult result)

{

if(result.getStatus() == ITestResult.FAILURE)

{

test.log(Status.FAIL, MarkupHelper.createLabel(result.getName()+" Test case FAILED due to below issues:", ExtentColor.RED));

test.fail(result.getThrowable());

}

else if(result.getStatus() == ITestResult.SUCCESS)

{

//String screenShotPath=GetScreenShot.capture(driver,"ScreenshotName");

test.log(Status.PASS, MarkupHelper.createLabel(result.getName()+" Test Case PASSED", ExtentColor.GREEN));

}

else

{

test.log(Status.SKIP, MarkupHelper.createLabel(result.getName()+" Test Case SKIPPED", ExtentColor.ORANGE));

test.skip(result.getThrowable());

}

}

@AfterSuite

public void teardown() throws InterruptedException{

dr.quit();

report.flush();

System.setProperty("webdriver.chrome.driver", System.getProperty("user.dir")+"\\Driver\\chromedriver.exe");

dr=new ChromeDriver();

dr.manage().window().maximize();

dr.get(System.getProperty("user.dir")+"\\test-output\\GriceReport.html");

Thread.sleep(2000);

}

}

**Imagetotext**

**package com.Grice.qa.pages;**

**import java.io.File;**

**import java.net.MalformedURLException;**

**import java.net.URL;**

**import java.text.SimpleDateFormat;**

**import java.util.Calendar;**

**import java.util.Properties;**

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

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

**import org.openqa.selenium.WebDriver;**

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

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

**import com.aventstack.extentreports.ExtentReports;**

**import com.aventstack.extentreports.ExtentTest;**

**import com.aventstack.extentreports.reporter.ExtentHtmlReporter;**

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

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

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

**public class Imagetotexttrnslation {**

**public static ExtentHtmlReporter htmlReporter;**

**public static ExtentReports report;**

**public static ExtentTest test;**

**public static Properties prop;**

**public static WebDriver dr;**

**public static File fs;**

**public static DesiredCapabilities cap;**

**public static AndroidDriver<AndroidElement>driver;**

**//public static RemoteWebDriver driver;**

**public static SimpleDateFormat far;**

**public static Calendar cal;**

**static{**

**cal=Calendar.getInstance();**

**far=new SimpleDateFormat("dd\_MM\_yyyy\_hh\_mm\_ss");**

**System.out.println(cal.getTime().toString());**

**htmlReporter=new ExtentHtmlReporter(System.getProperty("user.dir") +"/test-output/GriceReport\_"+far.format(cal.getTime())+".html");**

**}**

**@Test**

**public void imtt() throws InterruptedException, MalformedURLException{**

**fs=new File("D:\\Gricemobileapp\\Gricemobileapp\\com.techwave.grice-Signed.apk");**

**cap=new DesiredCapabilities();**

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

**cap.setCapability("platformName", "Android");**

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

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

**System.out.println("launch sucessfully");**

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

**driver.manage().timeouts().implicitlyWait(100, TimeUnit.SECONDS);**

**driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**driver.findElement(By.id("com.android.packageinstaller:id/permission\_allow\_button")).click();**

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

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

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

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**//login functionality**

**driver.findElement(By.xpath("//android.widget.EditText[@text='Work Email ID']")).sendKeys("ranjithkumar.nagiri@techwave.net");**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**driver.findElement(By.xpath("//android.widget.EditText[@text='Access Code']")).sendKeys("11201088A");**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

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

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**//camera**

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

**driver.manage().timeouts().implicitlyWait(200, TimeUnit.SECONDS);**

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

**driver.manage().timeouts().implicitlyWait(200, TimeUnit.SECONDS);**

**driver.findElementByAndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"English\"));").click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

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

**driver.findElementByAndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"German\"));").click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**driver.findElement(By.xpath("//android.widget.TextView[@bounds='[936,84][1080,228]']")).click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**Thread.sleep(2000);**

**driver.findElement(By.xpath("//android.widget.TextView[@bounds='[936,84][1080,228]']")).click();**

**Thread.sleep(4000);**

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

**Thread.sleep(6000);**

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

**Thread.sleep(10000);**

**System.out.println(driver.findElement(By.className("android.widget.TextView")).getText());**

**//voice**

**driver.findElement(By.xpath("//android.widget.ImageButton[@index='4']")).click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**driver.findElementByAndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Microsoft Server Speech Text to Speech Voice (de-AT, Michael)\"));").click();**

**Thread.sleep(6000);**

**//share**

**driver.findElement(By.xpath("//android.widget.ImageButton[@index='2']")).click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**// pin**

**driver.findElement(By.xpath("//android.widget.ImageButton[@index='0']")).click();**

**driver.manage().timeouts().implicitlyWait(80, TimeUnit.SECONDS);**

**}**

**}**