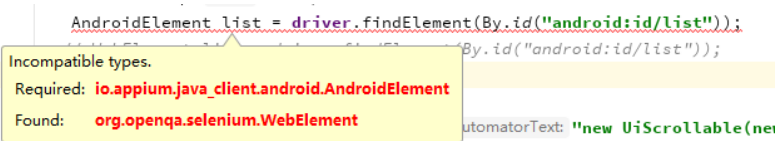


● webview 切换

- 自动向下滚动，代码在 \appium 自动化测试\appiumtest - findToast 中的 rolldown()

```
public void testWebView() throws InterruptedException {  
    //ApiDemos  
    driver.findElementByAccessibilityId("Views").click();  
    WebElement list = driver.findElement(By.id("android:id/list"));  
    //not AndroidElement  
    MobileElement radioGroup = list.findElement(MobileBy  
        .AndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(" +  
        "new UiSelector().text(\"WebView\");"));  
}  
}
```

- 如下问题的解决:



- 在 setUp()中无 setCapability("automationName", "uiautomator2");
- 类中添加 private static AppiumDriver<AndroidElement> driver2;
- 在 setUp()中改成 driver2 = new AppiumDriver<>(remoteUrl, desiredCapabilities);

- 完善后的自动向下滚动函数如下,代码在 \appium 自动化测试\appiumtest - findToast 中的 testWebView()

```
public void testWebView() throws InterruptedException {  
    //对应 ApiDemos 的 app 对应 setUpForWebview()  
    driver2.findElementByAccessibilityId("Views").click();  
    Thread.sleep(1000);  
    AndroidElement list = driver2.findElement(By.id("android:id/list"));  
    MobileElement webview = list  
        .findElement(MobileBy  
            .AndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(" +  
            "new UiSelector().text(\"WebView\");"));  
    webview.click();  
    System.out.println(driver2.getContextHandles());  
    for(AndroidElement e: driver2.findElementsByXPath("//*")){  
        System.out.println(e.getTagName());  
        System.out.println(e.getText());  
    }  
}
```

- adb shell pm list packages |grep webview  
adb shell dumpsys package com.adnroid.webview
- less /tmp/appium.log
- 注意对应版本 <https://github.com/appium/appium/blob/master/docs/en/writing-running-appium/web/chromedriver.md>，否则会提示[Chromedriver][STDERR][1.123][DEBUG]: DevTools request failed 等错误。
- 在打开 webview 时 adb shell cat /proc/net/unix | grep webview 可查看到进程  
或 adb shell cat /proc/net/unix | grep --text \_devtools\_remote  
通过 adb forward tcp:12184 localabstract:webview\_devtools\_remote\_4366(上一步进程中) 映射到端口

● capability 高级讲解

- 参照: <https://github.com/appium/appium/blob/master/docs/en/writing-running-appium/caps.md>
- browserName、printPageSourceOnFindFailure、chromedriverExecutable、dontStopAppOnReset
- capabilities.setCapability("noReset",true)  
capabilities.setCapability("fullReset",false)  
capabilities.setCapability("dontStopAppOnReset",true)
- (进阶第 11 阶)

```
capabilities.setCapability("automationName", "uiautomator2");  
capabilities.setCapability("capabilityName", "recreateChromedriverSessions", value "true");  
capabilities.setCapability("noReset", true);  
capabilities.setCapability("fullReset", false);  
capabilities.setCapability("dontStopAppOnReset", true);  
driver=new AndroidDriver<AndroidElement>(new URL( "http://127.0.0.1:4723/wd/hub"), ca
```

● Grid 模式

- 参照: <https://github.com/appium/appium/blob/master/docs/en/advanced-concepts/grid.md>  
其中 configuration 中的 ip 地址需要外部可访问
- 安装 wget http:// selenium-release.storage.googleapis.com/3.9/selenium-server-standalone-3.9.1.jar
- 启动 java -jar selenium-server-standalone-3.9.1.jar -role hub
- 浏览器可访问 127.0.0.1:4444/grid/console

● pageObject

- TestPageObject 文件和 MainActivity.fxml 文件中添加

<pre>public void testPageObject() throws InterruptedException{     MainActivity main = new MainActivity();     PageFactory.initElements(driver,main);     Welcome welcome = new Welcome();     PageFactory.initElements(driver.welcome);     welcome.跳过欢迎页();     main.自选.click(); }</pre>	<pre>&lt;element name="Views"&gt;     &lt;find&gt;findElementById&lt;/find&gt;     &lt;locator&gt;Views&lt;/locator&gt;     &lt;action&gt;click&lt;/action&gt; &lt;/element&gt;</pre>
--	---

- MainAcitivy 类和 Welcome 类

<pre>public class MainActivity {     @FindBy(xpath = "//*[@text='蛋卷基金']")     public WebElement 蛋卷基金;     @FindBy(xpath = "//*[@text='首页']")     public WebElement 首页;     @FindBy(xpath = "//*[@text='动态']")     public WebElement 动态;     @FindBy(xpath = "//*[@text='自选']")     public WebElement 自选;     public void enter 自选(){         自选.click();     } }</pre>	<pre>public class Welcome {     @FindBy(xpath = "//*[@text='登陆']")     public WebElement 登陆;     @FindBy(xpath = "//*[@text='跳过']")     public WebElement 跳过;     @FindBy(xpath = "//*[@text='下一步']")     public WebElement 下一步;      public void 跳过欢迎页() throws InterruptedException {         Thread.sleep(1000);         跳过.click();     } }</pre>
--	---

● 兼容性测试

- 尺寸、版本、定制 ROM、硬件
- <https://wetest.qq.com> 腾讯  
adb connect adb.wetest.qq.com:40470 → connected to adb.wetest.qq.com:40470  
adb devices → List of devices attached adb.wetest.qq.com:40470 device  
启动 appium 并运行测试代码后即可在远程自动测试。



- https://openstf.io STF Android 真机管理平台 ( Mac OS )  
brew install rethinkdb graphicsmagick zeromq protobuf yasm pkg-config 安装依赖  
npm install -g stf 安装  
rethinkdb & 启动  
stf local --public -ip 0.0.0.0  
ps -ef |grep rethinkdb ( 查看已经启动的 ) pkill rethinkdb 和 rm -r rethinkdb\_data/ 删掉  
浏览器可访问 http://localhost:7100 并登陆  
通过 ifconfig |grep 192.168 查询到 ip 地址可以外部连接真机

- 远程调试平台演练  
capabilities.setCapability("deviceName", System.getenv("deviceName"));  
shell 中 mvn test -Dtest="TestPageObject(所在的类名)"  
deviceName="xxx" mvn test -Dtest="TestPageObject" 连接真实设备  
从而完成多个设备的测试，自动化调度。

- https://mqc.aliyun.com 阿里等
- 常见问题：账号互踢、图形短信验证码、定位和复现

● 常用测试类

- xUnit 测试用例，体系：JUnit、UnitTest、TestNG
- 测试用例 testcase (名字、过程：单元测试，UI 自动化测试 Appium Selenium，接口自动化测试 RestAssured 等、断言：JUnit Assert, hamcrest)、测试类 class、测试套件 suite
- 执行顺序：Default 取决于反射方法获得的列表，顺序固定  
@FixMethodOrder(MethodSorters.JVM)顺序可能变化  
@FixMethodOrder(MethodSorters.NAME\_ASCENDING)按名字排序
- 带标签的测试类 (基本的测试类、方法级别的标签、类级别的标签@Category({类名 class,类名.class}))、注释忽略等。
- 数据驱动：改造类，如下图。可在本地批量同时运行(模拟)设备。

```
@RunWith(Parameterized.class)
public class SampleTest {
    Parameterized.Parameters(name="{index}")
    public static List<String[]> data(){
        return Arrays.asList(new String[][]{ {"a","11.0"}, {"b","11.1"}, {"c","11.3"} });
    } //数据也可以来源于外部文件

    @Parameterized.Parameter
    public String deviceName;
    @Parameterized.Parameter(1)
    public String version;

    @Before
    public void setUp() throws MalformedURLException {
        .....
        capabilities.setCapability("deviceName", deviceName);
        capabilities.setCapability("platformVersion", version);
        .....
    }
}
```

- 测试套件：多个测试类时，可定义所有类的执行顺序等。添加 TestSuite 类，如下图左。

```
@RunWith(Suite.class)
@Suite.SuiteClasses( {
    SampleTest.class,
    SampleTest_PageObject.class
} )
public class TestSuite { }
```

```
@RunWith(Categories.class)
@Categories.IncludeCategory(类名.class)
@Suite.SuiteClasses( {
    SampleTest.class,
    SampleTest_PageObject.class
} )
public class TestSuite { }
```

- 基于标签运行：上图代码中添加@Categories.IncludeCategory(类名.class)，则只运行带有@Category(类名.class)的类，如上图右。exclude 则为排除。
- 执行顺序控制  
class 与 method 的执行顺序：@BeforeClass setUpClass (配置 Capability，初始化 driver，安装 app)、@Before setUp (启动并进入特定界面)、@Test test1()、@After tearDown (关闭 app)、@Before setUp、@Test test2()、@After tearDown、@AfterClass teardownClass (driver.quit)
- 按照功能模块分组，不同的产品和模块进行分包，不同包下要有一个运行套件，考虑尽可能并行
- mvn 执行测试用例  
mvn -Dtest=TestCircle test  
mvn -Dtest=TestSquare,TestCircle test  
mvn -Dsfire.returnFalingTestsCount=2 test  
mvn 调试测试用例  
mvn -Dmaven.surefire.debug tset 等  
mvn 测试结果 (mvn test 默认所有测试用例的 class 以 Test 开头)  
mvn site  
mvn surefire-report:report