**一、等待处理**

**1.全局等待**

　　/\*全局设置，当元素识别不到的时候，可以接受的最长等待时间。\*/

　　driver.manage()timeouts().implicitlyWait(30, TimeUnit.SECONDS);

　　/\*全局设置，页面加载的最长等待时间。\*/

　　driver.manage()timeouts().pageLoadTimeout(30, TimeUnit.SECONDS);

　　/\*全局设置，关于JavaScript代码的异步处理的超时时间。AJAX请求。\*/

　　driver.manage()timeouts().setScriptTimeout(30, TimeUnit.SECONDS);

　　**2.元素等待**

|  |
| --- |
| public void waitForElement(By by) throws Exception{  for(int second = 1; second <= 30; second++){  try{  driver.findElement(by);  break;  }catch(Exception e){  System.out.println(e.getLocalizedMessage());  }  Thread.sleep(1000);  }  }  this.waitForElement(By.id("username"));//调用方法 |

　　系统自带的方法：

　　WebElement btnLogin = new WebDriverWait(driver, 10).until(

　　ExpectedConditions.presenceOfElementLocated(By.id("login")));

　　btnLogin.click();

　**二、断言**

**1.判断页面元素是否存在**

|  |
| --- |
| public boolean waitForElement(By by) throws Exception{  boolean isExsit = false;  for(int second = 1; second <= 30; second++){  try{  driver.findElement(by);  isExist = true;  break;  }catch(Exception e){  System.out.println(e.getLocalizedMessage());  }  Thread.sleep(1000);  }  return isExist;  }  if(this.waitForElement(By.id("username"))){  }else{  } |

　**2.判断页面上的元素的值/内容**

　　String result = driver.findElement(By.id("msg")).getText();

　　//对于内容的判断：

　　//1.严格匹配：result.equals("")

　　//2.模糊匹配：result.startsWith(""),result.endsWith, result.contains

　　//3.正则表达式：result.matches("正则表达式 No=.\*")

　　if(result.contains("aaa")){

　　}else{

　　}

　　**3.直接读取数据库的内容**

　　String sql = "SELECT ID FROM USERNAME ORDER BY ID "

　　int maxId =  thisgetMaxId(sql);

　　int postEqual = result.indexOf("=");//取=号在字符串中的位置

　　String newId= result.subString(postEqual + 1 );//从=号开始截取到最后,+1后移一位

　　if(maxId == Integer.parseInt(newId)){

　　}else{

　　}

　　**三、新窗口处理**

**1.对话框确认框的操作**

　　Alert alert = driver.switchTo().alert();

　　alert.accept();  //点击确定

　　alert.dismiss();  //点击取消

　**2.新窗口的操作**

|  |
| --- |
| //windowID切换  String loginID = driver.getWindowHandle();  for(String windowID : driver.getWindowHandles()){  if (!windowID.equals(loginID))  driver.switchTo.().window(windowID);  }  //windowTitle切换  for(String windowID : driver.getWindowHandles()){  driver.switchTo.().window(windowID);  Sring windowTitle = driver.getTitle();  if(windowTitle.contains("部分标题")){  break;  }  } |

　　**3.弹出窗口和Iframe**

　　driver.switchTo().frame("frameId");//切换到frame页面

　　driver.switchTo().window("windowhandle");//切换回到主页面

　**四、鼠标操作事件**

|  |
| --- |
| private Actions action;  //单击-click  public void click(){  action.moveToElement(drvier.findElement(By.linkText("login"))).perform();  action.click().perform(); //action的调用后面一定要加上perform，才能保证能真正的运行。  }  //双击-  public void doubleClick(){  action.doubleClick(drvier.findElement(By.linkText("login"))).perform();  }  //右键-  public void rightClick(){  action.contextClick(drvier.findElement(By.linkText("login"))).perform();  }  //悬停-  public void mouseHold(){  action.clickAndHold(drvier.findElement(By.linkText("login"))).perform();  }  //拖拽-  public void dragDrop(){  WebElement source = drvier.findElement(By.linkText("login"))  WebElement target = drvier.findElement(By.linkText("login"))  action.dragAndDrop(source, target);  action.dragAndDropBy(source, 200, 300);  } |

　**五、键盘事件处理**

**1.webDriver键盘操作-Action**

　　//webDriver键盘操作

　　action.sendKeys("A").perform(); //按下键盘A

　　action.sendKeys(Keys.Delete).perform();

　**2.Java键盘操作-Robot**

　　//Java键盘操作

　　//Java内含robot操作对象。throws Exception抛出异常给调用者。try{}catch(Excetion e){}//Excetion所有异常的父类，捕捉所有异常。

　　public void robotUsage(){

　　Robot robot = new Robot();

　　robot.mousePress(InputEvent.BUTTON1\_MASK); //鼠标左键点击

　　robot.mouseRelease(InputEvent.BUTTON1\_MASK); //鼠标左键释放

　　robot.mousePress(keyEvent.VK\_ENTER); //回车键

　　robot.mouseRelease(keyEvent.VK\_ENTER);

　　}

**3.组合键使用方法**

　　//标签页的切换,组合按键

　　public void tabSwitch() throws Exception {

　　for(String windowId: driver.getWindowHandles()){

　　driver.switchTo().window(windowId);

　　if(driver.getTitle().contains("title2"))

　　break;

　　}

　　action.keyDown(keys.CONTROL).sendkeys(keys.TAB).keyUp(keys.CONTROL).perform();//按下control,按tab,松开control

　　}

　**六、下拉框的操作**

　　Select selectshow = new Select(driver.findElement(By.id("test")));

　　int selectTotal = selectshow.getOptions().size(); //取得selectshow总长度，随机下拉值

　　int index = random.nextInt(selectTotal);

　　selectshow.selectByIndex(index);

　　selectshow.selectByVisibleText("a");

　　selectshow.selectByIndex(2);

　　**七、截图**

　　//现场截图

　　//设置保存路径

　　String path = System.getProperty("user.dir") + "/screenshot/";

　　//设置截图的文件名：随机数，使用当前时间，建议把模块名或测试用例编号附加上去

　　SimpleDateFormat formater = new SimpleDateFormat("yyyyMMdd\_hhmmss");

　　String time = formater.format(new Date().getTime());

　　String name = "Userstory(模块名）\_add（测试用例）\_" + time +".png";

　　TakesScreenshot tss = (TakesScreenshot)driver; //强制转换为TakesScreenshot类型

　　File image = tss.getScreenshotAs(OutputType.FILE);//java i/o的类

　　FileUtils.copyFile(image, new File(path + name));

　　**八、上传文件**

**1.用webDriver的sendKeys**

　　//driver.findElement(By.id("upload")).click(); //不执行打开命令，直接执行下面这句

　　driver.findElement(By.id("upload")).sendkeys("c://a.txt");//直接输入到无素里

　　**2.用JAVA自带的Robot按键功能操作**

　　driver.findElement(By.id("upload")).click();//打开上传窗口

　　Robot robot = new Robot();

　　robot.keyPress(KeyEvent.VK\_C);//用键盘输入C

　　robot.keyRelease(KeyEvent.VK\_C);//用键盘输入C

　　robot.keyPress(KeyEvent.VK\_COLON); //输入完整的路径，再按回车进行上传。

　　robot.keyRelease(KeyEvent.VK\_COLON);

　　**3.使用AutoIt工具上传**

　　上传详解：http://www.cnblogs.com/baihuitestsoftware/articles/5730133.html

　　1）用AutoIt生成exe

　　;ControlFocus("title","text",controlID) Edit1=Edit instance 1

　　ControlFocus("选择要加载的文件", "","Edit1")

　　; Wait 10 seconds for the Upload window to appear

　　WinWait("[CLASS:#32770]","",10)

　　; Set the File name text on the Edit field

　　ControlSetText("选择要加载的文件", "", "Edit1", "D:\\upload\_file.txt")

　　Sleep(2000)

　　; Click on the Open button

　　ControlClick("选择要加载的文件", "","Button1");

　　2）python调用例子

　　#coding=utf-8

　　from selenium import webdriver

　　import os

　　driver = webdriver.Firefox()

　　#打开上传功能页面

　　file\_path =  'file:///' + os.path.abspath('upfile.html')

　　driver.get(file\_path)

　　#点击打开上传窗口

　　driver.find\_element\_by\_name("file").click()

　　#调用upfile.exe上传程序

　　os.system("D:\\upfile.exe")

　　driver.quit()

　**九、WebDriver执行JavaScript**

　　1）简单实现调用js

　　WebDriver driver = new FireFoxDriver();

　　JavaScriptExecutor jse = (JavaScriptExecutor)driver;

　　jse.executeScript("usernameVar=document.getElementById('username');usernameVar.value='admin';");

　　jse.executeScript("passwordVar=document.getElementById('password');passwordVar.value='admin';");

　　jse.executeScript("document.getElementById('login').click();");

　　2)优化后调用js

|  |
| --- |
| WebDriver driver = new FireFoxDriver();  JavaScriptExecutor jse = (JavaScriptExecutor)driver;  String jsContent = "usernameVar=document.getElementById('username');"  +"usernameVar.style.borderWidth='3px';" //在输入用户名时加入特效，边框3像素，红色  +"usernameVar.style.borderColor='#f30';" //  +"usernameVar.value='admin';"  +"passwordVar=document.getElementById('password');"  +"passwordVar.value='admin';"  +"document.getElementById('login').click();";  jse.executeScript(jsContent);  //------------------  jsContent = "setKEContent('content', 'This is content!')"; //输入内容时，调用js里面的方法进行操作,content是ID  jse.executeScript(jsContent); |

　**十、webdriver在不同浏览器上运行**

|  |
| --- |
| public class BrowserUsage{  WebDriver driver;  public static void main (String[] args){  BrowserUsage bu = new BrowserUsage();  bu.firefox();  bu.ie();  bu.chrome();  }  public void firefox(){  System.setProperty("webdriver.firefox.bin", "C:\\firefoxdir\\firefox.exe");  driver = new FireFoxDriver();  }  public void ie(){  String path = System.getProperty("user.dir") + "lib"; //取得当前项目的目录  System.setProperty("webdriver.ie.driver", path + "IEDriverServer.exe"); //设置IEDriver的路径  DesiredCapabilities dc = DesiredCapabilities.internetExplorer();  dc.setCapability("ignoreProtectedModeSettings",true);  driver = new InternetExplorerDriver(dc);  }  public void chrome(){  String path = System.getProperty("user.dir") + "lib";  System.setProperty("webdriver.chrome.driver", path + "/chromedriver.exe"); //设置IEDriver的路径  driver = new ChromeFoxDriver();  }  } |