**Selenium WebDriver Course – Cheat Sheet:**

**TestNG:**

@BeforeClass

**public** **void** openBrowser()

{

// Implement This

}

@BeforeMethod

**public** **void** Before()

{

// Implement This

}

@AfterMethod

**public** **void** After()

{

// Implement This

}

@AfterClass

**public** **void** coseBrowser()

{

// Implement This

}

@Test

**public** **void** Test1()

{

// Implement This

}

**Drivers:**

WebDriverManager.*chromedriver*().setup();

WebDriverManager. *firefoxdriver()*.setup();

WebDriverManager. *internetExplorerDriver* ().setup();

**WebDriver Object Methods:**

driver.get("http://google.com");

driver.getCurrentUrl();

driver.getPageSource();

driver.close();

driver.quit();

driver.getWindowHandle();

driver.getWindowHandles();

driver.manage().window().maximize();

driver.manage().window().getSize();

driver.manage().window().setSize(dimension);

driver.manage().deleteAllCookies();

driver.manage().getCookieNamed("Cookie Name");

**Locators:**

WebElement ElemId = driver.findElement(By.*id*("Test"));

WebElement ElemName = driver.findElement(By.*name*("Test"));

WebElement ElemClass = driver.findElement(By.*className*("Test"));

WebElement ElemLink = driver.findElement(By.*linkText*("Test"));

WebElement ElemParLink = driver.findElement(By.*partialLinkText*("Test"));

WebElement ElemTagName = driver.findElement(By.*tagName*("p"));

WebElement ElemCss = driver.findElement(By.*cssSelector*("div[key='Test']"));

WebElement ElemXPath = driver.findElement(By.*xpath*("//table/tr/td/div"));

List<WebElement> AllLinks = driver.findElements(By.*tagName*("a"));

WebElement ElemFromList = driver.findElements(By.*tagName*("a")).get(0);

**Controllers:**

driver.findElement(By.*id*("btn")).click();

WebElement ElemBtn = driver.findElement(By.*id*("btn"));

ElemBtn.click();

driver.findElement(By.*id*("btn")).submit();

driver.findElement(By.*id*("txtArea")).sendKeys("myName");

driver.findElement(By.*id*("myText ")).clear();

Select mySelection = **new** Select(driver.findElement(By.*id*("myComboBox")));

mySelection.selectByVisibleText("My Tea");

mySelection.selectByIndex(2);

mySelection.selectByValue("Tea");

driver.findElement(By.*id*("datepicker")).click(); // Click to Open DatePicker Controller

WebElement dateWidget = driver.findElement(By.*id*("ui-datepicker-div"));

List<WebElement> cells = dateWidget.findElements(By.*tagName*("td"));

**for** (WebElement cell : cells)

{

**if** (cell.getText().equals("18")) // Selects 18th Date

{

cell.click();

**break**;

}

}

**Asserts and Verifications:**

**boolean** isDisp = driver.findElement(By.*id*("Test")).isDisplayed();

**boolean** isEna = driver.findElement(By.*id*("Test")).isEnabled();

**boolean** isSel = driver.findElement(By.*id*("Test")).isSelected();

**int** Width = driver.findElement(By.*id*("Test")).getSize().getWidth();

**int** Height = driver.findElement(By.*id*("Test")).getSize().getHeight();

**int** x = driver.findElement(By.*id*("Test")).getLocation().getX();

**int** y = driver.findElement(By.*id*("Test")).getLocation().getY();

String tagName = driver.findElement(By.*id*("Test")).getTagName();

String txt = driver.findElement(By.*id*("Test")).getText();

String Value = driver.findElement(By.*id*("Test")).getAttribute("value");

*assertEquals*("Expected" , "Actual");

*assertEquals*("Error Message", "Expected" , "Actual");

*assertNotEquals*("Expected" , "Actual");

*assertNotEquals*("Error Message", "Expected" , "Actual");

*assertTrue*(**true**);

*assertTrue*("Error Message", **true**);

*assertFalse*(**false**);

*assertFalse*("Error Message", **false**);

*assertNull*(Elem);

*assertNotNull*(Elem);

**Switch and Navigation:**

driver.navigate().to("http://google.com");

driver.navigate().refresh();

driver.navigate().back();

driver.navigate().forward();

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

popup.accept();

popup.dismiss();

String alertText = popup.getText();

popup.sendKeys("Hello World !");

WebElement iFrameElement = driver.findElement(By.*id*("test"));

driver.switchTo().frame(iFrameElement);

String handle = driver.getWindowHandle();

Set<String> handles= driver.getWindowHandles();

**for**(String winHandle : driver.getWindowHandles())

{

driver.switchTo().window(winHandle);

}

**Synchronization:**

Thread.*sleep*(1000);

driver.manage().timeouts().pageLoadTimeout(5, TimeUnit.***SECONDS***);

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

WebDriverWait wait = **new** WebDriverWait(driver, 10);

wait.until(ExpectedConditions.*presenceOfElementLocated*(By.*id*("Test")));

wait.until(ExpectedConditions.*elementToBeClickable*(By.*id*("Test")));

wait.until(ExpectedConditions.*visibilityOfElementLocated*(By.*id*("Test")));

wait.until(ExpectedConditions.*invisibilityOfElementLocated*(By.*id*("Test")));

wait.until(ExpectedConditions.*textToBePresentInElementLocated*(By.*id*("Test"), "Text to Insert"));

wait.until(ExpectedConditions.*titleIs*("My title"));

wait.until(ExpectedConditions.*titleContains*("My title Contains"));

**Error Handling:**

**try**

{

*assertEquals*("Expected", "Actual");

}

**catch** (Exception exp)

{

System.***out***.println("Element Was Not Found: " + exp);

}

**catch** (AssertionError assr)

{

System.***out***.println("Assert Failed: " + assr);

}

**finally**

{

System.***out***.println("Block to be Executed anyway");

}

**External Files:**

**public** **static** String getData (String nodeName) **throws** ParserConfigurationException, SAXException, IOException

{

File fXmlFile = **new** File("c://temp/config.xml");

DocumentBuilderFactory dbFactory = DocumentBuilderFactory.*newInstance*();

DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();

Document doc = dBuilder.parse(fXmlFile);

doc.getDocumentElement().normalize();

**return** doc.getElementsByTagName(nodeName).item(0).getTextContent();

}

**Extent Reports:**

ExtentReports extent = **new** ExtentReports("C://Test/Report.html", **true**);

ExtentTest test = extent.startTest("Test Name", "Test Descrption");

test.log(LogStatus.***PASS***, "Test Passed");

test.log(LogStatus.***FAIL***, "Test Failed" + test.addScreenCapture("C://Test/image.jpg"));

extent.endTest(test);

extent.flush();

extent.close();

**Actions:**

// drag & drop

WebElement draggable = driver.findElement(By.*id*("test"));

WebElement droppable = driver.findElement(By.*id*("test"));

**new** Actions(driver).dragAndDrop(draggable, droppable).build().perform();

//select multiple

List<WebElement> listItems = driver.findElements(By.*cssSelector*("test"));

Actions builder = **new** Actions(driver);

builder.clickAndHold(listItems.get(1)).clickAndHold(listItems.get(2)).click();

Action selectMultiple = builder.build();

selectMultiple.perform();

// double click

WebElement doubleClickElement = driver.findElement(By.*id*("dbl\_click"));

Actions action = **new** Actions(driver);

action.doubleClick(doubleClickElement);

action.perform();

// Mouse Hover

Actions action1 = **new** Actions(driver);

WebElement element = driver.findElement(By.*id*("mouse\_hover"));

action1.moveToElement(element).click().build().perform();

// Scroll

WebElement element1 = driver.findElement(By.*id*("scrolled\_element"));

((JavascriptExecutor) driver).executeScript("arguments[0].scrollIntoView(true);", element);