# Quick Reference Book

<http://narsingraoa.wordpress.com/basic-manual-testing-concepts/>

**TestScenarios:**

<http://www.softwaretestinghelp.com/sample-test-cases-testing-web-desktop-applications/>

**Unicode**

[http://unicode-table.com/en/#mandaic](http://unicode-table.com/en/%23mandaic)

**Span Shiksha**

<http://spanshiksha.skillport.com/skillportfe/main.action?id=CDE$642&content=summaryPage>

**Span Harate**

<http://sig.spanservices.com/TestingGroup/Lists/Testing%20Harate/AllItems.aspx>

<http://www.csmantra.com/NCERT-Books.html>

<http://careernewsweekly.com/india/finance/?sub1=t>

**VB script**

<http://w3schools.com/vbscript/default.asp>

<http://msdn.microsoft.com/en-us/library/t0aew7h6(v=vs.84).aspx>

Me5g<2$,I4:^-5!

**Java**

<http://docs.oracle.com/javase/tutorial/java/javaOO/objectcreation.html>

<http://www.vogella.com/java.html>

<http://www.tutorialspoint.com/java/java_overriding.htm>

<http://www.javatpoint.com>

<http://www.javawithus.com/tutorial/taking-input-from-the-user>

**TestNG**

<http://www.tutorialspoint.com/testng/index.htm>

**Auto IT**

<http://learnwebdriver.blogspot.in/p/autoit.html>

**CSS Selector**

<http://www.w3schools.com/cssref/css_selectors.asp>

<http://net.tutsplus.com/tutorials/html-css-techniques/the-30-css-selectors-you-must-memorize/>

**Xpath**

[http://release.seleniumhq.org/selenium-core/1.0/reference.html#locators](http://release.seleniumhq.org/selenium-core/1.0/reference.html%23locators)

**Xpath to CSS Selector:**

<http://cssify.appspot.com/>

**Groovy**

<http://www.robert-nemet.com/2011/11/groovy-xml-parsing-in-soapui.html>

**RegEX**

<http://msdn.microsoft.com/en-us/library/3206d374(v=vs.110).aspx>

<http://www.jdocs.com/javase/7.b12/java/util/regex/Pattern.html>

**Soap and WSDL:**

[http://my.safaribooksonline.com/book/web-development/web-services/0131401602/soap-and-wsdl/ch03lev1sec11#X2ludGVybmFsX0h0bWxWaWV3P3htbGlkPTAtMTMtMTQwMTYwLTIlMkZjaDAzbGV2MXNlYzMmcXVlcnk9](http://my.safaribooksonline.com/book/web-development/web-services/0131401602/soap-and-wsdl/ch03lev1sec11%23X2ludGVybmFsX0h0bWxWaWV3P3htbGlkPTAtMTMtMTQwMTYwLTIlMkZjaDAzbGV2MXNlYzMmcXVlcnk9)

<http://onebyteatatime.wordpress.com/2009/04/18/soapui-tips-n-tricks-part-2/>

**SoapUI Eclipse Plug-in**

<http://www.soapui.org/IDE-Plugins/eclipse-plugin.html>

**-------------------------------------------------------------------------------------**

**Copy-Paste in Command**

<http://www.techsupportalert.com/content/how-copy-and-paste-windows-command-prompt.htm>

<http://rbi.org.in/commonman/English/scripts/FAQs.aspx?Id=5>

**-------------------------------------------------------------------------------------**

**Maven**

//Maven Execution (Repository)

mvn -Declipse.workspace=C:\Users\Eclipse\workspace1 eclipse:add-maven-repo

//Restart Eclipse…

[https://maven.apache.org/guides/mini/guide-ide-eclipse.html#Maven\_2\_repository](QRB.docx)

**-------------------------------------------------------------------------------------**

**Selenium**

<http://www.youtube.com/watch?v=VwExSBTFGjQ>   
**-------------------------------------------------------------------------------------**

**Jenkins**

java -jar jenkins.war --ajp13Port=-1 --httpPort=8082

**-------------------------------------------------------------------------------------**

**Last Modified**

Javascript:alert(document.lastModified)

**-------------------------------------------------------------------------------------**

<https://fswiki.evry.com/display/TC/CEOS+ECC,+CoreWs+generation+2+FD+test+environment>

<https://fswiki.evry.com/display/TC/CEOS+ECC,+CoreWs+generation+2+BF+Preprod>

**MY WORKSPACE**

**Groovy:**

**log.info ((holder.getNodeValues("//\*:City")).toString())**

**----------------------------------------------------------------------**

Pattern p = Pattern. [**compile**](http://www.jdocs.com/link/java/util/regex/Pattern.html?m=M-compile(String)) ("a\*b");

Matcher m = p. [**matcher**](http://www.jdocs.com/link/java/util/regex/Pattern.html?m=M-matcher(CharSequence)) ("aaaaab");

boolean b = m. [**matches**](http://www.jdocs.com/link/java/util/regex/Matcher.html?m=M-matches()) ();

OR

boolean b = Pattern.matches("a\*b", "aaaaab");

**-----------------------------------------------------------------------------------------------------------------------------------------**

For all value”//\*” -- holder.getNodeValues("//\*:WeatherID")

**-------------------------------------------------------------------------------------**

def runnerComplete = testRunner.testCase.testSteps.DataSink

runnerComplete.setPropertyValue("property\_name","$variable\_holding\_result")

------------------------------------------------------------------------------------------------------------------------------------------

def groovyUtils = new com.eviware.soapui.support.GroovyUtils ( context )

//testrequest xml holder

def request = groovyUtils.getXmlHolder("TestRequest#Request")

------------------------------------------------------------------------------------------------------------------------------------------

EX:[1,2,3,4] = 1,2,3,4

------------------------------------------------------------------------

class CommonUtils {

def static String getStringValue(String[] listValue){

if(null == listValue){

return "";

}

if(0 == listValue.size()){

return "";

}

def String str = listValue.toString();

str = str.replace("[", "");

str= str.replace("]", "");

str= str.replace("null,", "");

str= str.replace("null", "");

return str;

}

}

-----------------------------------------------------------------------------

package com.google;

import java.io.File;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

import org.openqa.selenium.support.ui.Select;

public class BaseClass {

public static WebDriver driver;

public void browserInstantiate(String BrowserName) {

String browserName = BrowserName.toLowerCase();

if (browserName.contains("mozilla") || browserName.contains("firefox")

|| browserName.contains("ff")) {

driver = new FirefoxDriver();

} else if (browserName.contains("google")

|| browserName.contains("chrome") || browserName.contains("gc")) {

System.setProperty("webdriver.chrome.driver",

"D://chromedriver.exe");

driver = new ChromeDriver();

} else if (browserName.contains("internet")

|| browserName.contains("explorer") || browserName.contains("ie")) {

File file = new File("D://Selenium/IEDriverServer.exe");

System.setProperty("webdriver.ie.driver", file.getAbsolutePath());

driver = new InternetExplorerDriver();

}

}

public String getPageUrl(String Url) {

driver.get(Url);

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

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

String title = driver.getTitle();

if (title.equals("Error 404 (Not Found)!!1")) {

return "ERROR : Page not found";

} else if (title.equals("Problem loading page")) {

return "ERROR : Page not found";

} else if (title.contains("is not found")) {

return "ERROR : Page not found";

} else {

return "SUCCESS";

}

}

public WebElement getWebElement(By by) {

WebElement webelement = null;

webelement = driver.findElement(by);

return webelement;

}

public void myClick(WebElement WE) {

WE.click();

}

public void typeText(WebElement WE, String Typetext) {

WE.clear();

WE.sendKeys(Typetext);

}

public void mySelect(WebElement WE, String VisibleText) {

Select select = new Select(WE);

select.deselectAll();

select.selectByVisibleText(VisibleText);

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

}

}

-----------------------------------------------------------------------------

ArrayList<String> GooglearrayList = **new** ArrayList<String>();

**for** (WebElement webElement : link) {

GooglearrayList.add(webElement.getText());

}

System.*out*.println(GooglearrayList.get(1));

-------------------------------------------------------------------------------------

<td width="14%" height="21" align="center" bgColor="#efeef4">

<a onclick="self.opener.document.aspnetForm.ctl00$cphHRMSPages$FromDateTextBox.value='01/20/2014'; if(self.opener.aspnetForm.ctl00$cphHRMSPages$changeFlag) { self.opener.aspnetForm.ctl00$cphHRMSPages$changeFlag.value='true';self.opener.aspnetForm.submit(); }window.close();" href="#">

<a onclick="self.opener.document.aspnetForm.ctl00$cphHRMSPages$FromDateTextBox.value='01/26/2014'; if(self.opener.aspnetForm.ctl00$cphHRMSPages$changeFlag) { self.opener.aspnetForm.ctl00$cphHRMSPages$changeFlag.value='true';self.opener.aspnetForm.submit(); }window.close();" href="#">

-----------------------------------------------------------------------------

def randomNo = context.expand('${=(int)(Math.random()\*100000)}')

log.info randomNo

for (i in 1..10){

//request.getDomNode("//\*:ZIP")

request.setNodeValue("//\*:ZIP", randomNo)

request.updateProperty()

testRunner.runTestStepByName("GetCityForecastByZIP")

testRunner.runTestStepByName("Validation")

testRunner.runTestStepByName("Groovy Script 2")

testRunner.runTestStepByName("DataSink")

}

------------------------------------------------------------------------------------

//KeyPress in TestNG

driver.findElement(By.*className*("gbqfif")).sendKeys(Keys.*chord*(("\u0915"),("\u092E"),("\u093E"),("\u0932"),Keys.*SPACE*,("\u0939"),("\u0948"),("\u0964")));

Kamal hai(hindi)

("\u0915")- 2325

("\u092E")-2350

("\u093E")-2366

("\u0932")-2354

(“\u0020”) SPACE.0032

("\u0939")-2361

("\u0948")-2376

("\u0964") |line.-2404

Super(kannada)-("\u0CB8"),("\u0CC1"),("\u0CAA"),("\u0CB0"),("\u0CCD")

//DIV[@id='cst']/DIV

TimeUnit

D:\Maven\apache-maven-3.1.1\lib

Point hoverItem =driver.findElement(By.xpath("Value")).getLocation();  
((JavascriptExecutor)driver).executeScript("return window.title;");      
Thread.sleep(6000);  
((JavascriptExecutor)driver).executeScript("window.scrollBy(0,"+(hoverItem.getY())+");");

Locatable hoverItem = (Locatable) driver.findElement(By.xpath("//li[text()='Reklama w Google']"));

int y = hoverItem.getCoordinates().getLocationOnScreen().getY();

((JavascriptExecutor)driver).executeScript("window.scrollBy(0,"+y+");");

JavascriptExecutor js = (JavascriptExecutor)driver;

js.executeScript("window.scrollTo(0,Math.max(document.documentElement.scrollHeight," +

"document.body.scrollHeight,document.documentElement.clientHeight));");

public void scrollToElement(By by) {

Locatable element = (Locatable) selenium.findElement(by);

Point p= element.getCoordinates().getLocationOnScreen();

JavascriptExecutor js = (JavascriptExecutor) selenium;

js.executeScript("window.scrollTo(" + p.getX() + "," + (p.getY()+150) + ");");

}

//\*[@id="rso"]/li[1]/div/h3/a

Actions action = new Actions(driver); action.doubleClick(myElemment); action.perform();

<a href="javascript:window.opener.Build('aspnetForm.ctl00$cphHRMSPages$ToDateTextBox', '10','2', '2014', 'MM/DD/YYYY');">

<a onclick="self.opener.document.aspnetForm.ctl00$cphHRMSPages$ToDateTextBox.value='02/01/2014'; if(self.opener.changeFlag) self.opener.changeFlag='true';window.close();" href="#">

//DIV[@id='page2']

//FORM[@id='aspnetForm']/DIV[1]

WebDriverWait wait = new WebDriverWait(driver, 300);

WebElement triggerDropDown = driver.findElement(By

.className("ui-helper-hidden"));

triggerDropDown.click();

WebElement selectElement = wait.until(ExpectedConditions

.visibilityOfElementLocated(By.id("formLevel:levels\_input")));

Select select = new Select(selectElement);

select.selectByVisibleText("SECURITY");

"All the desirable things in life are either illegal, expensive, fattening or in love with someone else!"