15 Oct 23

Goal :	Z2M Selenium	MC Selenium
Selenium WebDriver	✓	✓
Eclipse	✓	✓
Java	✓	✓
Git	✓	✓
maven	✓	✓
jenkins ci/cd	✓	✓
log4j	✓	
Apache POI	✓	✓
TestNG	✓	✓
Cucumber/Gherkin		✓
Framework dev't	✓	
json/yaml		✓
Extent report	✓	✓

workspace : c:\jay\training\workspaces\eclipse-selenium
repos gh : github.com
ssh file : jayssh2
pass phrase : jaysshpassphrase2
repos local : c:\jay\training\repo\git\selenium-training

16 Oct 23 [Sel WD w J B2A]
pomos : |||| |||| ||||
Setting up mvn

Selenium -> org.seleniumhq.selenium ; selenium-java ; 4.14.1 ; compile
(mvnrepository)
Test NG -> org.testng ; testng ; 7.8.0 ; test
selenium.dev/downloads

- Tech Stack
- Project Elements + organization / framework design
- Packaging and Deployment
- test data prep and take down
- execution modes
- test results reporting
- Test scenario / cases organization

Practice Sites

-> https://rahulshettyacademy.com/client
-> https://www.phptravels.net/flights ☆

ChromeDriver implements org.openqa.selenium.WebDriver

- org.opensa.selenium.chrome.ChromeDriver, edge.EdgeDriver, firefox.FirefoxDriver

download, invoke in code chromedriver.exe

└ bypass Selenium mgr auto download

Selenium Web Driver Locators

Locator

- way to identify html element on a page
- ID, Xpath, CSS selector, name, Class Name, Tag Name, Link Text, Partial Link Text

e.g. driver. findElement (By.<locator type> (<value>)).<action>

driver.findElement (By.id ("inputUser")).sendKeys ("sometext");

CSS Selector

- <tagName>.<className>
- <tagName>#<id> ; or ; #<id>
- <tagName>[attribute='value']

e.g. <p class="error"> Wrong uid or pwd</p> -> p.error

ChroPath / SelectorsHub - browser plugins to check if selector is unique

implicit wait - put a delay before next loc is executed

e.g. driver.wait (3000);

- you can also get css via inspect > console e.g $('p.error')

18Oct23
Wed

p's ; ||||-||||-||||-||||-|

Selectors can't

Link Text - used for anchor element

e.g. <a href="#"> Forgot your pwd?</a>
↳ By.linkText("Forgot your pwd?")).click();

Xpath - <Tagname>[@<attrib>='<value>']
e.g <input type="text" placeholder="Username" value=" ">
→ input[@placeholder="Username"]

*↖ Selectorhub can also generate this!

to check in inspect > console:
> $x('<xpath str>')

If you have multiple elements w/ similar attrib & value, they can be referenced via "index" → via xpath or cssselector
e.g. <input type="text" >
<input type="text" > ←
→ input[@type='text'][2] (by xpath)
using css selector → input[type='text']:nth-child(2)
tip: prefer unique attrib vs indexing

Xpath via parent & child traverse
→ e.g. //form/input[3]

ElementClickInterceptedException
- means button is not yet clickable. need a wait
put in explicit wait

Using regex
- use if a name is dynamic.
e.g. regex in xpath → //button[contains(@class,'submit')]
//<tag>[contains(@class,'<str>')]

Locator : tagName

Locator : xpath based on button name

e.g. → //button[text()='Log Out']

//*[text()='Log Out']

Siblings xpath traverse

Abs - start fr. beginning

relative - middle ← preferred

- starts w/ double slashes //

e.g. //header/div/button[1]/following-sibling::button[1]

child to parent traverse (::)

//header/div/button[1]/following-sibling::button[1]/parent::div

- using xpath, css locator is unable

note: css is faster than xpath

Running in maximized mode

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

Navigating to another page → driver.navigate().to("<page>");

.get() ← loads the whole page before moving forward

to go back to previous page → driver.navigate().back();

" " .forward();

Drop Downs

Select html

- this can be referenced in selenium via org.openqa.selenium.support.ui.Select

e.g. WebElement webE = driver.findElement(By

id("<id of select element>"));

Select dropdwn = new Select(webE);

```
dropdown.selectByVisibleText("AED");
        - ByValue
```

install chropath

23 Oct Mon p's : ||||

Section 55 : Dynamic Dropdowns
Section 56 : Parent-Child relationship locator

e.g.

```
driver.findElement(By.xpath("//div[...]  //a[...]")).click();
                          parent ↗       child ↗
```

S58 : Auto suggestive Dropdown

If you need to search through an unordered list and you only have the text value available, use List<WebElement> and iterate through the list.

e.g.

```
List<WebElement> options = driver.findElements(By.cssSelector("li[...] a"));
 for (WebElement option : options){
       if (option.getText().equalIgnoreCase("...")){
             option.click();
       }}
```

S60 : Checkbox and getting size

checkbox could be clicked already so it would be a good idea to call WebElement.isSelected, then invoking .click() if its false

27 Oct Fri p's : ||||

S60 : To retrieve multiple elements use findElements()

e.g.

```
List<WebElement> elements = driver.findElements(By.xpath
        ("..."));
```

S61 : Using Assertions

In testng this API is used : org.testng.Assert

e.g. Assert.assertEquals(actual, expected);

.assertTrue , .assertFalse

- Assertions are usually invoked to test a state after an action

S63: Calendar UI

30-Oct-23
Monday

P's : ||||-||||-||||

S64: Validating UI Elements enabled/disabled w/ attributes

- Inspect the element to find out the attribute you need to investigate. Inspect would be done before an after an action has been triggered on the element.
- Once you have the attribute, you can use getAttribute to retrive the associated value
- Then in code, you would make calls to getAttribute() before and after the action, thus the difference can be observed

e.g. driver.findElement(By.id("")).getAttribute("style");

68 : Handling Java Alerts

Alerts are non-html elements, they are referenced via switchTo() method

driver.switchTo().alert().accept();

66 : To check for presence of element in page use isDisplayed()

Tip: Don't use dynamic / changing text for locators

Section 9 Synchronization

achieved by:

Implicit Wait, Explicit Wait, Thread.sleep, Fluent Wait

#77 Implicit Wait

- set globally, conditional, applies globally
- have to be cautious in using or setting it for too long

↳ can be used to assert for perf issues e.g. page not loading at expected time

let x be the max time on elem complete loading, the wait should not be set more than x

Explicit Wait

- targets specific element
- not global

31 Oct tuesday

p's: ||||-||||-||

```
e.g. WebDriverWait expWait = new WebDriverWait(driver, 5);
     expWait.until(ExpectedConditions.visibilityOfElementLocated(locator));
```

WebDriver Wait

Fluent Wait - a type of Explicit Wait

- searches for webelement repeatedly at intervals of time until timeout or object is found

2-Nov Thurs

p.s: ||||-||||-||||-||||-||||-|

Fluent Wait - need to build customized methods unlike WDW

```
e.g. Wait<WebDriver> wait = new FluentWait<WebDriver>(driver)
          .withTimeout(int).pollingEvery(int)
          .ignoring(NoSuchElementException);
     WebElement elem = wait.until(new Function<WebDriver, WebElement>(){
          public WebElement apply(WebDriver driver) {
               return getWebElement(); }
     });
```

Section 10 AJAX Calls, Child Windows, IFrames

#86 Ajax, Mouse

org.openqa.selenium.interactions.Actions
- used to perform actions on an element

e.g. mouse over an element

Actions ac = new Actions(driver);
ac.moveToElement(driver.findElement(By.cssSelect("."))).build().perform();

Actions.click() ← against Action
WebElement.click() ← against WebElement

Actions API follows a Builder Design Pattern click(), keydown(), sendKeys()
.doubleClick() returns an appended Actions object.

.contextClick() ← used to right click

#88 Handling Windows

driver.switchTo().window(child)
- used to switch to another window. To get child, use driver.getWindowHandles which returns a Set<String>

#91 Frames

Frames
- display content independent of its container
- elements inside iframe cannot be reached by a locator without switching to the frame first. Frame are identifiable by name, id or an attribute

e.g. driver.switchTo().frame(WebElement);
- iframes have tagname "iframe" so you can call findElements (By.tagname("iframe")) to return all iframe webElement

#94 Limiting Scope

If you need the scope of findElement(s) to be limited, you can call them against a WebElement object instead of the driver

e.g.
```
WebElement sectionB = driver.findElement(By.ID());
    sectionB.findElements(By.tagName("a")).size();
```
↳ returns count of <a> in specific section

another 'section' can also be extracted from sectionB

#95 Checking multiple links are working

e.g.
```
new Action(driver).moveToElement(links.get(i)).keyDown(Keys.CONTROL)
        .click().build().perform();
```
hold CTRL btn while clicking a link

Alternative
```
String clickOnTab = Keys.chord(Keys.CONTROL, Keys.Enter);
section.findElements(By.tagName("a")).get(i).sendKeys(clickOnTab);
```

3 Nov Friday

#96 Iterating through multiple open tabs

1. getWindowHandles, it will return a set of string
2. get an iterator and iterate through each one

```
Set<String> tabs = driver.getWindowHandles()
        Iterator iter = tabs.iterator();
```

#98 Calendar

"Element not clickable @ pt (x,y)" - when you get this error, consider if you need to scroll down the page

e.g.
```
JavaScriptExecutor js = (JavaScriptExecutor) driver;
    js.executeScript("window.scrollBy(x,y)");
```

Also advised to put a ExpWait till element is clickable prior to doing click action on an element

- grab common attrib, put in a list and iterate

for explicit waits use expected conditions that accept WebElement instances rather than By instances
At the most unstable pt. of the test script, instead of using selenium waits use Thread.sleep(n);

Section 12

#102 Scrolling within Table and Window level using JavaScriptExecutor

e.g. JavascriptExecutor js = (JavascriptExecutor) driver;
    js.executeScript(script);

Some js scripts;
    window.scrollBy(x, y) ← scroll window by the x,y coord.

Note: use the console in inspect tool to test js
    document.querySelector(Css locator).scrollTop = 5000

4 Nov Sat Section 13

|||| #105 SSL Check
|||| - To bypass SSL Checks use ChromeOptions (for Chrome)
|||| e.g. ChromeOptions opts = new ChromeOptions();
|||| opts.setAcceptInsecureCerts(true);
|||| WebDriver driver = new ChromeDriver(opts);

other browsers have counterparts

#106 other ChromeOptions
→ addExtensions() adds chrome extension you need to run for automation
→ Proxy - when site uses a proxy you need to configure
    .setHTTPProxy
e.g. Proxy prox = new Proxy();
    Prox.setHttpProxy("ipaddress:4444");
    opts.setCapabilities("proxy", prox);
see: chromedriver.chromium.org/capabilities

→ start chrome maximized
→ block pop-up windows
→ setup default download dir

→ deleting cookies: driver.manage().deleteAllCookies();
- might be useful when testing/verifying login url

#109 Taking Selenium Screenshots

→ selenium.TakesScreenshot

e.g. File src = ((TakesScreenshot) driver).getScreenshotAs(OutputType.FILE);
FileUtils.copyFile(src, new File("path"));

#110 Broken Links

→ links can be debugged via chrome devtools > Network tab

programatically: get url → open connection → chk response code

#113 Soft Assertions

→ SoftAssert a = new SoftAssert()
a.assert... a.assertAll()
→ allow execution to continue, a.assertAll() call at the end will display the assertion results

Section 14

#116 Java Streams

Java Streams
→ perform aggregate operations on java collections

Note Install testNG for Eclipse Marketplace

→ → java collections have an API .stream() that returns a Stream<T> and from the stream, .filter() can be called

.filter
→ accept a param 'predicate' that is used for filtering the collection

.forEach (s → action) Stream.concat

.map (s → { })
→ modifies the stream

Predicate
→ function that takes an input and returns boolean

.anyMatch (Pred) → return true if any element matches predicate

. collect (Collectors) → return a collection based on defined collectors

e.g.

Map<String, List<Person>> peopleByCity = personStream
    .collect(Collectors.groupingBy(Person::getState, Collectors
        .groupingBy(Person::getCity)));

e.g fruits.stream().filter().collect(Collectors.toList());

#117 WebTable Sorting using java streams

5Nov Sun → following-sibling::td[n] traverses xpath to the next sibling

||||

||||

||

Section 15: Selenium 4.0 features

136. #123 Relative Locators

→ To use, include the line below: // note: only w/ tagnames

import static org.openqa.selenium.support.locators.RelativeLocator.*;

→ e.g. driver.findElement(with(By.tagName("label"))
    .above(Webelement));

→ Allows elements to be located via its tagname and relation to its position w/ a nearby webelement

note; if the element is a 'flex' element, the relative locator does not recognize it and will proceed to the next

→ .above

.below

.toLeftOf

.toRightOf

#126 Invoking multiple Windows/Tabs using one drive instance

→ driver.switchTo().newWindow(WindowType.TAB)

→ get window handles and assign them w/ names

→ call switchTo().window(handle) for other window

→ call get, passing in url of other site

locator tip: for css locator, "a[href*='....']" ← the (*) asterisk allows search for the href's with values containing the given string

#127 Partial Screenshots

→ webElement.getScreenshotAs(OutputType.FILE)

e.g. File src = webElement.getScreenshotAs(OutputType.FILE);
FileUtils.copyFile(src, new File("path"));

#128 Validating (UX) WebElement height + width

→ webelement.getRect().getDimension().getHeight();
→ " " " .getWidth();

Section 16: Framework Part 1: TestNG

7 Nov Tues

→ testNG classes don't have main class
→ test cases are written as methods annotated w @Test

testNG.xml
→ to create in eclipse, right click on proj > TestNG > Convert to TestNG

Test NG hiehrarchy
Test Suite → Test Folder (Shell) → Test Cases
→ to run a test suite, right click on testNG.xml > run as testsuite

```
suite
  └ test
      └ classes
          └ class
```

#134 Include + Exclude → method level control for which tests to run or not
→ exclude - excludes methods from the suite
→ include - runs test case only under this

#135 Executing testcases at package level w/ regex

→ using regex relies on 'naming convention' → "run all test cases starting with 'mobile*' e.g. <exclude name="*API.*"/>

→ package ; test can also be run on package level in a suite

suite
  ↳ package // <package name="com.joy.test"/>

#136 TestNG Annotations

@BeforeTest

→ methods that execute first prior to any test case

@AfterTest

→ last method to execute after all tc methods in the class

@BeforeClass

→ 1st to execute when java class is run

→ usually for login e.g.

@AfterClass

→ last to execute e.g. deleting cookies at the end.

note: there can be multiple instances of these annotations - They would go FIFO

@BeforeSuite

@AfterSuite

include the

→ does as their name implies. They are methods executed before and after the suite

@BeforeMethod / @AfterMethod

→ executes before/after every test method in encompassing class

#138 Groups

→ added to test method annotation as attributes

e.g. @Test(groups={"Smoke"})

→ in the xml, the <groups> tag is added at the suite or test level

e.g.
```
<test name="regression">
    <groups>
        <run> <include name="Smoke"/> </run>
    </groups>
<classes> ... </classes>
</test>
```

#139 Annotation Helper Attributes

dependsOnMethods

- attribute for testNG annotations that sets method prerequisites for the method to execute
  e.g. @Test(dependsOnMethods = {"webloginCarLoan"})

→ also proceeds to execute the methods specified as prerequisite
must be annotated w/@Test

enabled

→ allows the testcase to be disabled e.g. tc failed for a release and bugfix being done e.g. @Test (enabled=false)

timeout

→ eg. @Test (timeout = 4000)

#140 Parameterizing from testng.xml

→ use <parameter> Tag

→ applied on suite, test, classes, class, methods

set in the testng.xml

→ Thus in the class method, add a @Parameters annotation. The value of the parameter name specified in the annotation is automatically passed to the method argument w/ the annotation

e.g

```
@Parameters({"URL"})
@Test
public void weblogin (String urlstr)
```

in the method

```
<parameter name="URL" value="google.com"/>
```

DataProvider Annotation

→ set on methods that returns data set to be used in other test methods.

e.g

```
@DataProvider
→ public HashMap<String, String> getData() {..}
@Test (dataProvider = "getData")
public void testDoSomething(String uid, String pwd){}
```

→ allowed return types : Object[][], Iterator<Object[]>, Object[], Iterator<Object>

#143 TestNG Listeners

→ reroute execution upon certain conditions met

→ listen to testcases

→ must implement ITestListener interface

- expose listener method which can be defined to perform specific action.

eg. public void onTestFailure (ITestResult) {
//... take snapshot }

→ in TestNg.xml, under suite, the listeners must be declared

eg. <listeners><listener class-name="com.java.TestListener"></listener>
</listeners>

Running tests in parallel

In the testng.xml, on the suite or test tag you can add the "parallel" attribute

e.g. <suite name="testsuite" parallel="classes" thread-count="2">
↳ refers to # of threads to run in parallel

Note: rather than use parallel, selenium grid is preferred.

Section 18: Test Automation Framework part 1

/ → create mvn structured framework w/ necessary dependencies

/ → implement POM to drive locators from classes

/ → drive object creation w/ in Page Object Classes encapsulating it for test

/ → create base test w/c sets browser config details + Global properties

/ → Decide the Test strategy how test are clubbed + distributed w/ annotation

/ → Create testng runner file to trigger test w/ single pt of execution control

/ → Introduce grouping in Testng.xml to categorize tests

/ → implement data-driven testing + parameterization using testng Dataproviders Hashmap + json file readers

/ → implement testng listeners to capture screenshot on automatic test failures and logging

/ → create extent report wrapper to generate excellent html reports for the application

/ → parally execution

/ → testng retry mechanism to rerun failed flaky tests

/ → Run framework tests w/ mvn commands w/ testng maven integration plugin

/ → implement Maven runtime variables to replace global parameters of test data at runtime

/ → integrate framework w/ jenkins w/ parameterized build pipeline jobs + schedule jobs on specific time frames

/ → add cucumber wrapper to existing framework w/ Cucumber testNG runner

8 Nov Wed

||||
|||
→||||
|||

Creating a framework I

tip stream().anyMatch()
→ if you just need a boolean

1. Setup project
   - mvn project, dependencies
2. StandAlone test

— ~ o ~

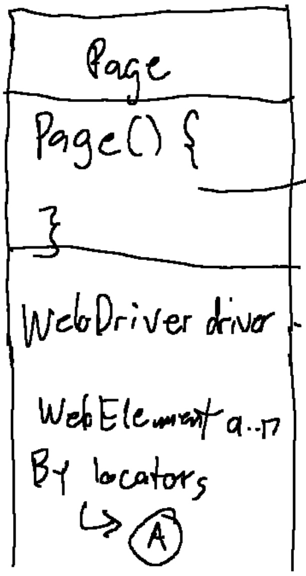
3. Create Page Objects

Section 19: Page Object Model + Factory Design Pattern

Page Object Model

eg. Login Page is designed as a class and all locators for the page are defined in the class

→



Constructor (driver)
→ initilize driver
→ initialize PageFactory e.g. Pagefactory.initElements(driver, this);

webelements are declared and defined using @FindBy Annotation e.g.:

@FindBy (id = "userEmail")
WebElement userEmail;

@FindBy (css = ".userPwd")
WebElement password;

called PageObjectFactory

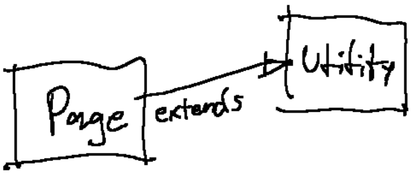
(A) → define locators used in action methods

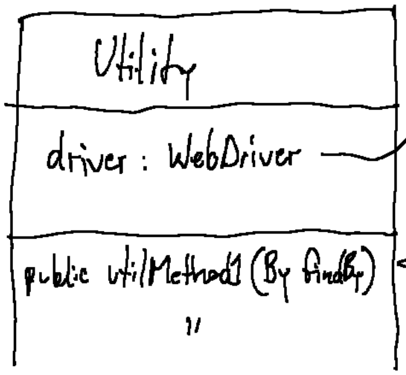
Note:
PageObject should not hold any data but only Elements + Actions

Utilities

→ ideally in src/main/java

→ separate package e.g joylabs. utils





passed via constructor of child classes

pass a 'By' parameter if the method requires By locators.

Elements common to all PO Classes should also be defined here

834

9 Nov Thu

|

1148

||||| 933

||||||||| 330

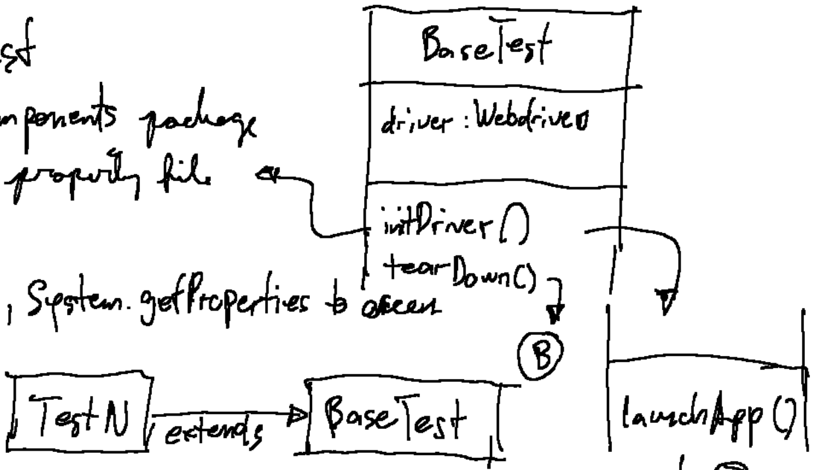
→ PageObject method leading to other pages should return page object type of that page

#166 Creating BaseTest

→ should be in test. components package

→ initDriver → reference property file to setup driver

→ use Properties API, System.getProperties to access



Properties
→ browser = chrome

(A) return type of launch app should be initial PageObject → (A)
eg. return new LandingPage();

#168

@BeforeMethod should be applied to (A) → ensures base page is loaded prior to any test method execution

@AfterMethod is applied to (B) → usually to close driver

#169 generate testng.xml by converting project to testng

12 Nov Sun #171 Running tests in parallel

|| → In testng.xml, <suite> has an attrib, 'parallel' that needs to be set (w/c can be

||||| 835p tests, methods, classes, instances

→ eg <suite parallel = "tests" ...

13 Nov Mon 926 Issue @BeforeMethod not running when specific group is run in testng.xml

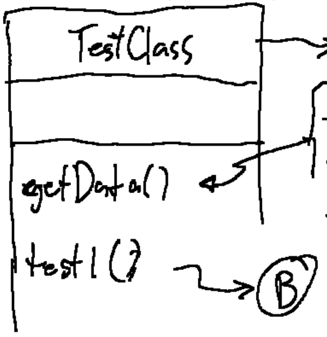
|||| → To solve this, you must include alwaysRun = true in BeforeMethod declare

|||| 412p e.g @BeforeMethod (alwaysRun = true)

test 555

grouping tests → is done via e.g. @Test (groups = {"ErrorHandling"}), then in the testng.xml, add the <groups> elements just below suite

#172 DataProvider + Parameterization II



→ [Basetest

→ @DataProvider

→ returns Object[][], or Iterator<Object[]>

→ to use, Ⓑ

Ⓑ ← the @Test annotation must specify the attrib 'dataProvider' value of which would be name of the dp method

e.g. @Test (dataProvider = "getData")

→ then, the test method arguments should declare the fields represented in the Object[].

eg. public void submitOrder (String uid, String pwd, String prodname)

→ the arguments above are automatically initialized by the dp

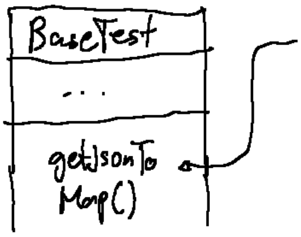
→ the test method will run as many times depending on the number of obj[] sets in the iterator.

#173 Using HashMap

→ for hashmap, one hashmap instance would represent a set of testdata for one test run. So if you have 3 different sets of data, you need 3 hm instances e.g. HashMap<String, String>

→ the argument of the test method using hm dataprovider must declare the hash map. value is retrieve via get()

#174 Reading data from JSON file



utility class to convert json file into java HashMap

note: see commit 4f8ede5

#175 Screenshot Utility

note: see commit # 0dbaa4e

14 Nov Tues

|||| 458a

|||||| 742

||| 1201

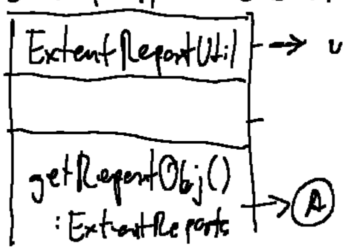
||||||| 525p

|||

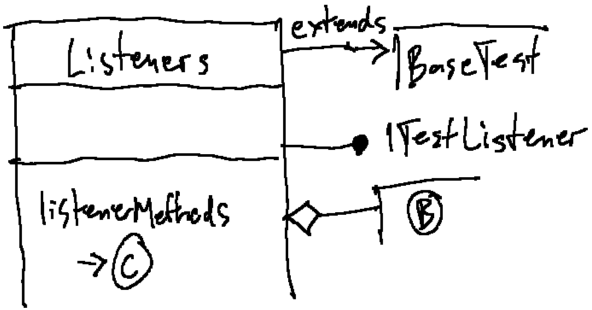
24p

#176 Extent Reports

commit # 3f898a7



→ utils package in src/main/java



instantiated via ExtentReportUtil

B - ExtentTest test

- ExtentReports

- ThreadLocal <ExtentTest>

C

→ onTestStart() - initialize listener state

→ onTestSuccess() - set test as passed

→ onTestFailure() - log test as fail and log throwable; take screenshot

→ onStart() - initialize driver (via BaseTest)

→ onFinish() - flush extent and do teardown

from BaseTest

#179 Issues w/ Concurrency

note: see commit # 4759cc0

IRetry Analyzer

→ rerun flaky test

→ have a Retry class implement IRetryAnalyzer

→ Retry class is referenced via argument in @Test, retryAnalyzer = Retry.class

#181 Running via mvn cmds

→ update pom.xml

→ add plugin for maven-surefire-plugin and adjust <suiteXmlFile> accordingly to point to the correct testng.xml file

→ create profile for each test suite

→ open a console and go to proj folder

→ run $ mvn test -P<profile id>

#182 mvn global parameters ; updating tests @ runtime

→ run $ mvn test -Dbrowser=firefox -PRegression

→ modify BaseTest, where property is read to check for parameter values passed via cmd line, -D

other info to pass via cmd : environment, profile name, etc.

#183 Jenkins

→ download Generic Java pkg war and unpack using cmd below :

→ java -jar jenkins.war -httpPort:9090

└ starts jenkins on this port

15 Nov Wed

#184 Creating a job in jenkins on jenkins folder

||||

→ download + install

|||| 1272 1367

→ new item ; freestyle project ; configure accordingly.

|

Note: if the project being built is not at the root of a git repo then you need to use the "use custom workspace" option and specify the exact directory

#185 Headless Mode

→ this prop is set in ChromeOptions

Note : refer to # a9668a1

#186 Scheduling Jenkins Job

→ configure > build triggers > build periodically then input schedule as approp.

e.g. 30 6 * * * → runs build at 6:30 everyday

Section 27 : Cucumber

→ BDD framework ; uses gherkin

Gherkin

→ Business readable , domain-specific language describing software behavior

Cucumber Scenario = TC

→ contains steps / test steps

→ uses ff. keyword : Given, When, Then, But , And

Given → preconditions

When → user action

But - used for negative scenarios

Then → expected output / result

And → aggregates other items for G, W, T

eg.

Scenario : Make minimum due payment

Given : User is on Pay Credit Card page

When : User fills in details

And : Select min amnt option

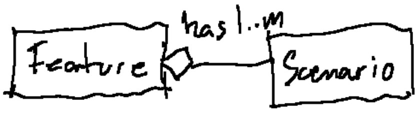
And : clicks pay button

Then : CC info page is displayed

Feature - a business requirement

Feature File - test suite of all scenarios

- .feature extension



Scenario Outline

- similar to a scenario but parameterized and has a dataset

#189 Setting-up

→ mvn needs 2 dependency - cucumber-java , cucumber-testng

→ in eclipse , install the cucumber plugin

→ create a feature file in a cucumber package under src/test/java

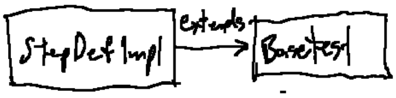
e.g. SubmitOrder.feature

```
@tag
  Feature: Purchase the order from ecommerce website

      Background:
      Given I landed on Ecommerce Page

      @Regression
      Scenario Outline: Positive test of Submitting the order
Ⓐ         Given Logged in with username <uid> and password <pwd>
          When I add product <prodname> to cart
          And Checkout <prodname> and submit the order
          Then "Thankyou for the order." message is displayed on
                  confirmation page

          Examples:
            | uid                   | pwd              | prodname
            | myemail@yahoo.com     | 123Surprise7456  | ZARA COAT3
```

#190 Step Definitions

16Nov Thu → create inside cucumber.stepdefinitions under src/test/java

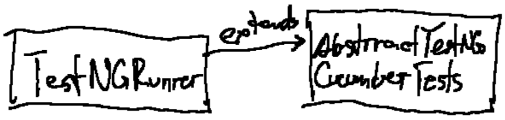
|| 7:107 → ties and maps keywords with java methods



| 929 Cucumber testng runner

→ ties cucumber resources w/ Testng

→ a java class, defined in cucumber package



note: refer to #06d3003

17 Nov Fri

|| 752 Ⓐ - tags can be referenced in TestNGRunner file

|||| 10:36 TestNGRunner

||| 258 → is annotated w/ @CucumberOptions making several tag attribs available::

features: path where feature files are
glue: package where stepdefinitions are
monochrome: boolean to enable console output as readable
plugin: where cumber report is generated
tags: references tag name in feature file to define scope of Scenarios to run.

e.g.

```
@CucumberOptions (features="src/test/java/jayslabs/cucumber",
                  glue = "jayslabs.cucumber.stepDefinitions", monochrome=true,
                  tags="@ErrorValidation",
                  plugin={"html:target/cucumber.html"})
    public class TestNGRunner extends AbstractTestNGCucumberTests {}
```

→ to integrate the cucumber runner in mvn, just add another profile

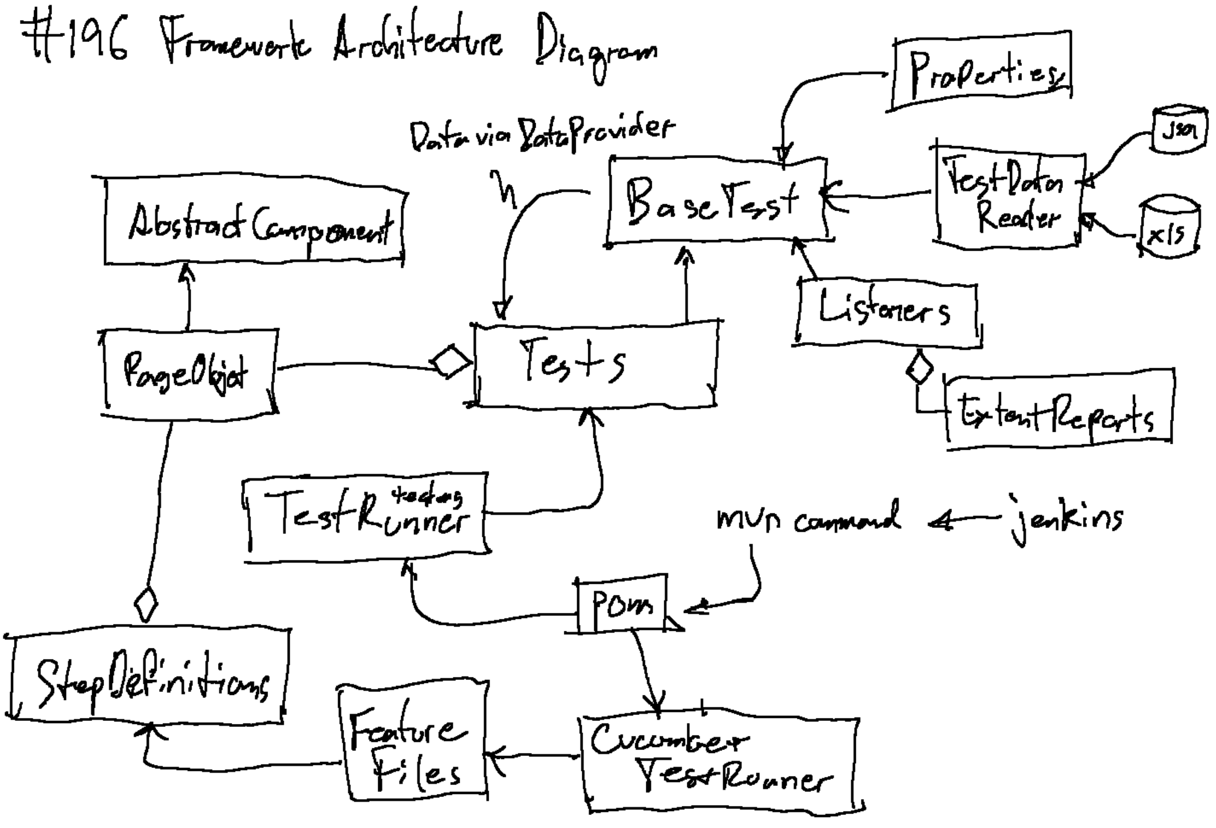
Note: There's a mvn issue where a profile defined to run cucumber doesn't run, in this case try to remove the surefire plugin default if its running the testNG runner instead, and replace it with the cucumber runner

#b81c9fd

20 Nov Mon
||||

21 Nov Tues
||| 100
| C29
|||||||| 710
12p

#196 Framework Architecture Diagram



#199: Apache POI

mvn dependencies : poi-ooxml & poi   refer to ExcelDataUtil

1. XSSFWorkbook class   4. access specific row   #89f6ccf

2. access xls sheet   5. access all cells of row

3. access all rows of sheet   6. convert data from cells into Arrays

#206 DataProvider & Excel

→ Its important to integrate xls data to dataprovider as dataprovider will provide results for each test execution of each test data represented in the xls

→ method annotated w/ DataProvider should return a List<HashMap<String,String>>

→ XSSFWorkbook, XSSFSheet, Row^I, Cell^I \ or an Object[][] use DataFormatter

22 Nov Wed → refer to # 7542964

644

||

||||||| 917

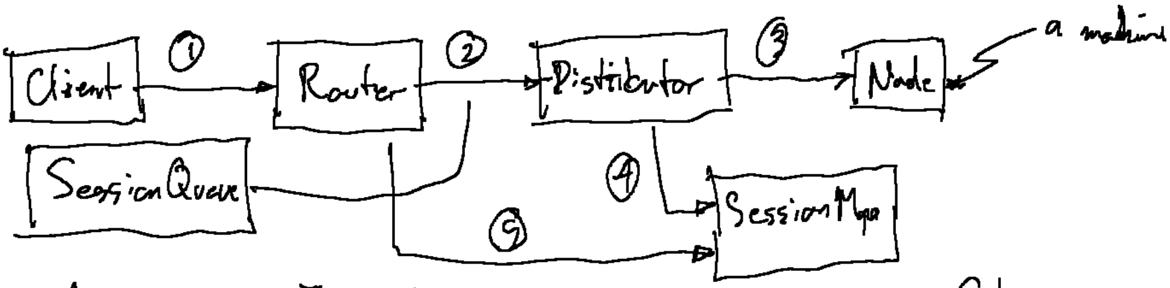
|| 314

Section 27 Selenium Grid

Selenium Grid

→ smart proxy server enables tests to run in parallel on multiple machines

→ allows testing on different browser versions



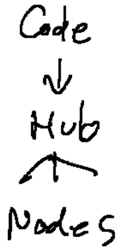
1. Sends request to router. Router receives

2. ask distrib to create new session ID

3. Assign session to particular node

4. Store session into the map e.g. sessionId / Node URI

5. Subsequent request to router is checked against map



Setting-up

- Grid roles : Standalone, Hub & Node, Hub, Node, Distributed

→ Hub + Node - for small to med size grids

1. Start Hub → <java -jar <sel jar> hub ← on grid folder

2. Start Node in same machine → java-jar <sj> node --detect-drivers true

→ prior to 1 & 2, selserver jar & browser drivers must be present in a folder. This is done for all machines in the grid

3. Start Node in different machine →

java -jar <seljar> node --detect-drivers true --publish-events tcp://192.168.68.67:4442 --subscribe-events tcp://192.168.68.67:4443

Setting up Client

-> the driver must be initialized using RemoteWebDriver passing to it, an URL object pointing to the grid server

e.g. driver = new RemoteWebDriver (new URL ("http://192.168.1.1:4444"), opts);

where opts is an instance of ChromeOptions

refer to #3c1c520

Section 29: DB connection to Selenium

-> mysql v8.0.35

-> open mysql workbench, connect to server

29 Nov Wed

|||| 525

||||| 813

||| 222

SQL -> "create database <db>;"

-> use <schema>;

-> create table <name> (var1 vartype, varn varntype ...);

-> describe <table>;

- to use mysql in java project, download mysql-connector-java-5.1.21.jar

-> in mvn : mysql:mysql-connector-java

-> insert into <table> values (val1, valn);

-> java.sql.DriverManager   i.e. use tester1 / 123K@must@k@456

.getConnection (url, user, pwd)

url : "jdbc:mysql//<host>:<port>/schema"

e.g "jdbc:mysql:// localhost:3306/jayqadbt"

-> Take note that the user you are using has proper access to the schema

refer to #b3ea2cf

#239 Window Authentication Pop-up

-> not browser based but tied to browser

-> to enter the uid + pwd to the pop-up, the values must be part of the string passed to driver.get(site);

e.g. site = "http://admin:admin@the-internet.herokuapp.com/";

refer to #4a9f2dc selenium-training:master

#241 File upload + ~~AutoIT~~ java.awt.Robot

30 Nov Thu #218 Chrome Devtools Protocal (CDP)

7~725 -> Chrome Dev Tools - built into browser

945 - 10:11 -> Selenium has wrapper around CDP

-> see chromedevtools.github.io/devtools-protocol

CDP + Selenium

- Capture + Monitor and Stub network request + responses
- Inject session cookies & perform basic auth
- Mock Device Coordinates for mobile / Tabs view
- Check & monitor site performance
- Mock geolocations of the user
- Block network requests to optimize Test execution
- Mock faster/slower network speeds
- Execute + debug js
- View console logs

Note: only for chromium based browsers. Excludes firefox

2 Dec Sat #219 Simulating browser as mobile

130-219 Chromium Driver

829p - 844p -> used to access CDP devtools

+ 1h 30m e.g. ChromeDriver driver = new ChromeDriver();

Devtools dtools = driver.getDevTools();

dtools.createSession();

-> at this point, you can send CDP commands (see CDP url above) via dtools.send() passing in command as a parameter

-> Eg. Emulation.setDeviceMetricsOverride() to emulate a browser on an iphone

-> e.g. Network.getRequestPostData

↳ selenium command wrapper for CDP tools

#220 Custom Commands

-> driver.executeCdpCommand

see #6a52c37

-> doesn't use dtools.send

e.g. dtools.send(
Emulation.setDeviceMetricsOverride(...)
);

→ after creating a session dtools.createSession();

```
→ ImmutableMap.Builder<String, Object> params = ImmutableMap.builder();
params.put("<attrib1>", <value>);
params.put("<attrib n>", <value>");
driver.executeCdpCommand("<cdp command>", params.build());
```

#221 Localization Testing w/ CDP

→ Emulation.setGeolocationOverride

→ use as per above, take note to set chromeoptions to enable geolocation

```
e.g. ChromeOptions options = new ChromeOptions();
     options.addArguments("--enable-features=Geolocation");
     options.addArguments("--lang=es");
```

Note: website might block CDP, in this case ChromeOptions should instead be used.

#222 Network Responses & Status Codes w/ CDP

3 Dec Sun
1h 44m 450-634
34m 637-711
36m 748-824
33m 827-9
= 3h 27m

→ cdp: Network.enable (Method) ← required for when Network traffic need to be accessed

→ cdp: Network.loadingFailed (Event)
Network.loadingFinished "

→ to use the cdp events, you have to add listener to dtools

```
e.g. dtools.addListener(Network.responseReceived(),
        response -> {
            Response res = response.getResponse();
        });
```

→ cdp: Network.requestWillBeSent (Event) - to track request

note:. the response can be tracked to log for codes starting in "4"

#70b9a4d

#223 Intercept Network resp w/CDP to simulate scenario

-> Fetch.enable (method)

-> Fetch.requestPaused (event)

- after Fetch.enable has been invoked, a listener has to be defined for the above event, the event is triggered when request URL matches the specified filter. Inside the listener definition the request can be evaluated, modified, and the request continued (Fetch.continueRequest)

e.g.
```
dtools.send(Fetch.enable(..));
dtools.addListener(Fetch.requestPaused(), request -> {
    Request rq = request.getRequest();
    if (rq.getUrl().contains("...")) {
        rq.getUrl().replace(" ", " ");
        dtools.send(Fetch.continueRequest(..));
    } else // send rq w/out modification
});
```
refer to # 032ce99

#224 Test failed Network request calls

-> pass an appropriate pattern on Fetch.enable

-> then, add a listener on Fetch.requestPaused event, inside it, invoke Fetch.failRequest

refer to # 6b80fff

#225 Block unwanted Network request calls

-> done to speed up test execution, reducing flaky tests

-> different from fail because the calls are prevented from executing

-> Network.setBlockedURLs ← param: List<String>

```
e.g dtools.send(Network.setBlockedURLs(ImmutableList.of("*.jpg", "*.css")));
```
↳ blocks jpg & css

refer to # 510014d

4 Dec Mon
1h 15m 553-708
1h 22m 10:50-1212
2h 8m 116-324
40m 330-410
27 724-751
5h, 52m

#226 Emulating Network speed

-> Network.emulateNetworkConditions (method)

-> Network.connectionType (type) : none, cellular2g, cellular3g, etc.

-> Network.loadingFailed (event) - add listener for this and read loadingFailed obj

#227 Selenium uriPredicate function for Basic Authentication

→ register uriPredicate w/ the driver, passing in values via org.openqa.selenium.UsernameAndPassword.

→ cast driver to HasAuthentication to invoke register() as above

e.g. Predicate<URI> uriPredicate = uri → uri.getHost().contains("...");
    ((HasAuthentication) driver).register(uriPredicate, UsernameAndPassword.of("uid","pwd"));

refer to #9dd05c8

#228 Logging JS errors

→ selenium.logging package exposes some classes: LogEntries, LogEntry, LogType

e.g. LogEntries entries = driver.manage().logs().get (LogType.BROWSER);
    List<LogEntry> logs = entries.getAll();
    & contains log info to log accordingly, typically via listeners

note #aee7565

#Section 31 : Cross Browser testing on Cloud using 3rd Party Venders

→ Vendors maintain infra w/c has varying browser flavors

→ On Browser stack:
  - → create Browser stack profile
  - → setup Browser stack on eclipse via marketplace
  - → add mvn dependancy for browser stack: com.browserstack:browserstack-java-sdk;
  - → in base test, init a RemoteWebDriver LATEST, passing in URL for browserstack:
    https://hub.browserstack.com/wd/hub
  - → in pom, add <argline> to profiles w/ javaagent pointing to full jar location

note # c60d375

~~#330 Bugzilla~~

Interview Q's

1. Design Pattern used in framework
    - Page Object Model
        - java class designed around pages; structured
    - Page Factory
        - creats / derive locators via annotations
        - WebElement or List of

2. How are reusable utilities handled w/in the framework?

→ 2 ways(1) via parent class abstract Components for selenium objects and BaseTest(2) parent class for test classes which handles driver init, data loading, teardown activities and common behavior across tests (classes, methods) e.g. screenshots, common page elements, etc.

(1) for AbstractComp this defines functions common across p.objs

(1) Utility for Selenium/Page Object related

(2) Utility for Test Classes related

3. How do you drive data from external files into the framework

json file → json reader class → BaseTest → @DataProvider in test method
↳ via jackson databind ↳ by testng

4. Interface use

→ Listeners ; ITestListener (testng) ; for common report entries at various test stages
WebDriver (selenium), WebElement

5. How are you achieving encapsulation in the framework

→ limiting access to fields via appropriate access modifiers

6. Process for running test in parallel

→ In testng.xml, set parallel attrib on the <suite>, use ThreadLocal wrapper on test variables enabling each thread to have its own local copy of the variable making the code threadsafe

7. Do you use static keyword in your test framework?

→ If so, not the driver objects else you cannot do parallel testing. Static elements/variables are shared across* all threads*. If it just variable w/ read access then they can be static

9. How do you send global props to your test at run time

→ using java.util.Properties. After instantiating a FileInputStream obj passing to it the path of properties file, the fis instance is used to instantiate the properties obj.

→ they can also be passed via cli as a parameter in a mvn command: "-D" and using System.getProperty()

10. What's the mechanism used to run only selected set of tests inside the framework?

→ testng groups annotation for tests and referencing this in testng.xml. Also create separate profile in POM to run specific suites

11. How do you handle flaky tests

→ IRetryAnalyzer org.testng

12. How do you implement screenshots in reports

→ through listeners, selenium has the screenshot functionality and this is exposed/defined via a method in Basetest.

→ org.testng.ITestListener