**Updating the properties file, DriverScript.java and excelUtilities.java:**

Now we need to update the DriverScript.java and excelUtilities.java.

Since we have special keywords here like checkusersdifference, checkvehiclesdifference, checkbookingsdifference, checkbrandsdifference etc (we call these by passing the expected result like checkbrandsdifference(1) etc., in the actual test case), we need to handle these special keywords in our DriverScript.

For this we put all these special keywords in .properties file first as CHECK\_KEYWORDS and these are used for checking something and these are not data driven and won’t have any object repository.

**config.properties file:**

WB\_PATH\_TESTS = C:\\Users\\padal\\eclipse-workspacenew\\HybridKeywordDriven\\tests\\tests\_main.xlsx

WB\_PATH\_TESTS\_SHEET = TestCases

WB\_PATH\_TEST\_CASES = C:\\Users\\padal\\eclipse-workspacenew\\HybridKeywordDriven\\tests\\

MASTERKW\_PATH = C:\\Users\\padal\\eclipse-workspacenew\\HybridKeywordDriven\\data\\masterkeywords.xlsx

MASTERKW\_SHEET = globalkeywords

DATA\_PATH = C:\\Users\\padal\\eclipse-workspacenew\\HybridKeywordDriven\\data\\data.xlsx

SPECIAL\_KEYWORDS = adminpostvehicle,adminlogin,adminpostbrand

CHECK\_KEYWORDS = checkusersdifference,checkvehiclesdifference,checkbookingsdifference,checkbrandsdifference,checksubscribersdifference,checkqueriesdifference,checktestimonialsdifference

ADMIN\_UNAME = admin

ADMIN\_PWD = Test@12345

USER\_UNAME = subbu123

USER\_PWD = subbu123

Now in our DriverScript.java, we first load these CHECK\_KEYWORDS and verify if the keywords in test case are any one of these CHECK\_KEYWORDS. If the keyword is a part of this check keywords, then we call a new function which is specialcheckfunction(), by passing this special check keyword. Before doing this we split the special keyword based on the open parenthesis and get the expected result and pass this as well to specialcheckfunction(). We pass driver also to this.

**DriverScript.java executeTC() method:**

**public** **static** **void** executeTC(List<String> originalkeywords, Set<String> actionclass, Map<String, String> keywordvsac, Set<String> objectrepository, Map<String, String> keywordvsor) **throws** ClassNotFoundException, NoSuchMethodException, SecurityException, IllegalAccessException, IllegalArgumentException, InvocationTargetException, InstantiationException, IOException {

System.*setProperty*("webdriver.gecko.driver", "C:\\BrowserDrivers\\geckodriver.exe");

WebDriver driver = **new** FirefoxDriver();

Properties gldata = **new** Properties();

InputStream input = **new** FileInputStream("src/executionEngine/config.properties");

gldata.load(input);

**for**(String str : originalkeywords) {

String keyword = str;

String[] specialkeywords = gldata.getProperty("SPECIAL\_KEYWORDS").split("\\,");

**boolean** skstatus = **false**;

**for**(**int** i=0; i<specialkeywords.length; i++) {

**if**(keyword.indexOf(specialkeywords[i]) != -1) {

skstatus = **true**;

**break**;

}

}

**String[] checkkeywords = gldata.getProperty("CHECK\_KEYWORDS").split("\\,");**

**boolean ckstatus = false;**

**for(int i=0; i<checkkeywords.length; i++) {**

**if(keyword.indexOf(checkkeywords[i]) != -1) {**

**ckstatus = true;**

**break;**

**}**

**}**

**if**(skstatus) {

**if**(keyword.indexOf("(") != -1){

String[] parts = keyword.split("\\(");

String[] dataelements = (parts[1].split("\\)"))[0].split("\\,");

*specialfunction*(parts[0],dataelements,driver);

}

**else** {

String parts = keyword;

String[] dataelements = **null**;

*specialfunction*(parts,dataelements,driver);

}

}

**else if(ckstatus) {**

**if(keyword.indexOf("(") != -1){**

**String[] parts = keyword.split("\\(");**

**String checkkeyword = parts[0];**

**int expectedvalue = Integer.*parseInt*((parts[1].split("\\)")[0]));**

***specialcheckfunction*(checkkeyword,expectedvalue,driver);**

**}**

**}**

**else** {

//System.out.println("going to normal");

String actioncl = keywordvsac.get(keyword);

String objectcl = keywordvsor.get(keyword);

Class<?> cls = Class.*forName*("actions."+actioncl);

Class<?> orc = Class.*forName*("objectrepository."+objectcl);

Method[] methodcall = cls.getDeclaredMethods();

**for**(Method m : methodcall) {

**if**(keyword.equalsIgnoreCase(m.getName()) && m.getParameterCount() == 0)

{

Method mc = cls.getDeclaredMethod(keyword);

Constructor<?> constructor = cls.getConstructor(WebDriver.**class**);

mc.invoke(constructor.newInstance(driver));

}

**else** **if**(keyword.equalsIgnoreCase(m.getName()) && m.getParameterCount() == 1)

{

Method morc = orc.getDeclaredMethod(keyword);

Constructor<?> orconstructor = orc.getConstructor(WebDriver.**class**);

WebElement we = (WebElement) morc.invoke(orconstructor.newInstance(driver));

Method mc = cls.getDeclaredMethod(keyword,WebElement.**class**);

Constructor<?> constructor = cls.getConstructor(WebDriver.**class**);

mc.invoke(constructor.newInstance(driver),we);

}

}

}

}

}

Now in the special function we call a excelUtilities class method getCheckKeywordActionsClass() method to get the action class for the special check keyword. Once we get the action class for the special check keyword, we use java reflections to execute the method for getting the difference in brands or users etc.,

**DriverScript.java specialcheckfunction():**

**public** **static** **void** specialcheckfunction(String checkkeyword, **int** expectedvalue, WebDriver driver) **throws** IOException, ClassNotFoundException, NoSuchMethodException, SecurityException, IllegalAccessException, IllegalArgumentException, InvocationTargetException, InstantiationException{

Properties gldata = **new** Properties();

InputStream input = **new** FileInputStream("src/executionEngine/config.properties");

gldata.load(input);

excelUtilities ecu = **new** excelUtilities();

String actioncl = ecu.getCheckKeywordActionsClass(checkkeyword, gldata.getProperty("MASTERKW\_PATH"), gldata.getProperty("MASTERKW\_SHEET"));

Class<?> cls = Class.*forName*("actions."+actioncl);

Method mc = cls.getDeclaredMethod(checkkeyword, **int**.**class**);

Constructor constructor = cls.getConstructor(WebDriver.**class**);

mc.invoke(constructor.newInstance(driver),expectedvalue);

}

**ExcelUtilities.java getCheckKeywordActionsClass() method:**

//This function will return the action class for the special check keywords

**public** String getCheckKeywordActionsClass(String keyword, String wbpath, String sheetname) **throws** IOException{

FileInputStream fis = **new** FileInputStream(wbpath);

XSSFWorkbook workbook = **new** XSSFWorkbook(fis);

XSSFSheet sheet = workbook.getSheet(sheetname);

//Get all the rows

Iterator<Row> rows = sheet.iterator();

String actionclassvalue = "";

**while**(rows.hasNext()) {

**try** {

Row rowno = rows.next();

Cell kwvalue = rowno.getCell(0);

Cell acvalue = rowno.getCell(2);

String keywordvalue;

DataFormatter df = **new** DataFormatter();

keywordvalue = df.formatCellValue(kwvalue);

**if**(keyword.equalsIgnoreCase(keywordvalue)) {

actionclassvalue = df.formatCellValue(acvalue);

**break**;

}

}

**catch**(Exception e) {

}

}

workbook.close();

**return** actionclassvalue;

}