**Updating properties file with the special keywords and checking these keywords in the executeTC():**

In executeTC() of DriverScript.java we are checking the special keywords with the condition

**if**((keyword.indexOf("adminpostvehicle") != -1)||(keyword.indexOf("adminlogin") != -1))

Since we have two special keywords this is fine. But if we have more number of special keywords then we have to change the script every time. But this is not recommended. So we can maintain these keywords in properties file and check these names in executeTC() method. If we get more special keywords like these, we can directly add them in properties file.

Update the properties file with these keywords.

**.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

ADMIN\_UNAME = admin

ADMIN\_PWD = Test@12345

USER\_UNAME = subbu123

USER\_PWD = subbu123

Now we will modify the executeTC function to check these keywords.

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;**

**}**

**}**

**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** {

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);

}

}

}

}

}

We are loading the properties file first. Then we are getting the specialkeywords and then we are checking the keyword with these special keywords.

Now our framework is completed and in order to automate the remaining scenarios, we have to add action classes and object repositories. If there are any special keywords create separate sheets for them and add those names in the properties file in SPECIAL\_KEYWORDS. Also prepare the test cases with those module names.