**Getting the action classes and object repositories for keywords for individual test cases:**

Now we have the individual keywords so we have to get the action classes and object repositories for these keywords.

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

**public** **static** **void** getDependencies(List<String> keywords) **throws** IOException, ClassNotFoundException, NoSuchMethodException, SecurityException, IllegalAccessException, IllegalArgumentException, InvocationTargetException, InstantiationException {

List<String> originalkeywords = **new** ArrayList<String>(keywords);

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

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

gldata.load(input);

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

List<?> lis = ecu.getActionsObjects(keywords, gldata.getProperty("MASTERKW\_PATH"), gldata.getProperty("MASTERKW\_SHEET"));

Set<String> actionclass = (Set<String>) lis.get(0);

Map<String, String> keywordvsac = (Map<String, String>) lis.get(1);

Set<String> objectrepository = (Set<String>) lis.get(2);

Map<String, String> keywordvsor = (Map<String, String>) lis.get(3);

*executeTC*(originalkeywords,actionclass,keywordvsac,objectrepository,keywordvsor);

}

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

//This function will return the action class names and object repository names a particular test case needs

**public** List getActionsObjects(List<String> keywords, String wbpath, String sheetname) **throws** IOException{

//This set holds the action class names needed

Set<String> actionclasses = **new** HashSet<String>();

//This set holds the object repository names needed

Set<String> objectrepositories = **new** HashSet<String>();

//This map holds the keyword vs action class

Map<String, String> keywordsvsac = **new** HashMap<String, String>();

//This map holds the keyword vs object repositoy name

Map<String, String> keywordsvsor = **new** HashMap<String,String>();

//This ArrayList holds all the above and will be returned

List lis = **new** ArrayList();

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

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

XSSFSheet sheet = workbook.getSheet(sheetname);

//Get all the rows

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

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

**try** {

Row rowno = rows.next();

Cell kwvalue = rowno.getCell(0);

Cell acvalue = rowno.getCell(2);

Cell obvalue = rowno.getCell(3);

String keywordvalue;

String actionclassvalue;

String objectrepositoryvalue;

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

keywordvalue = df.formatCellValue(kwvalue);

**for**(**int** i=0; i<keywords.size(); i++) {

**if**(keywords.get(i).equalsIgnoreCase(keywordvalue)) {

actionclassvalue = df.formatCellValue(acvalue);

objectrepositoryvalue = df.formatCellValue(obvalue);

actionclasses.add(actionclassvalue);

keywordsvsac.put(keywords.get(i), actionclassvalue);

objectrepositories.add(objectrepositoryvalue);

keywordsvsor.put(keywords.get(i), objectrepositoryvalue);

}

}

}

**catch**(Exception e) {

}

}

workbook.close();

lis.add(actionclasses);

lis.add(keywordsvsac);

lis.add(objectrepositories);

lis.add(keywordsvsor);

**return** lis;

}