*Excel sheet for keyword driven framework*

|  |  |  |  |
| --- | --- | --- | --- |
| *Testcase ID* | *TS\_ID* | *Description* | *Action\_Keyword* |
| Login\_01 | TS\_001 | open the browser | openBrowser |
| Login\_02 | TS\_002 | navigate to website and login | login |
| Login\_03 | TS\_004 | operations in dashboard | View |
| Login\_04 | TS\_005 | operations in postjobs window | post |
| Login\_05 | TS\_006 | operations in job respond window | jobRespond |
| Login\_06 | TS\_007 | operations in fetchresume window | fetchResume |
| Login\_07 | TS\_008 | operations in fetchjobs window | fetchJobs |
| Login\_08 | TS\_009 | operations in changesettings window | changeSettings |
| Login\_09 | TS\_010 | logout from the site and close the browser | Logout |

*package executionEngine;*

import java.io.File;

import java.util.concurrent.TimeUnit;

import org.apache.log4j.Logger;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import keyword.ActionKeywords;

import utility.ExcelUtils;

public class Testmain {

// Keyword driven Framework

/\*\*

\* @author Theo

\*\*/

public static void main(String[] args) throws Exception {

// Declaring the path of the Excel file with the name of the Excel file

String nPath = System.*getProperty*("user.dir")+"\\Resources\\toolsqa.xlsx";

String s="Test Steps";

// Here we are passing the Excel path and SheetName as arguments to connect with Excel file ExcelUtils.*setExcelFile*(nPath, s)

for (int i=1;i<=9;i++){

//Storing the value of excel cell in sActionKeyword string variable

String sActionKeyword = ExcelUtils.*getCellData*(i, 3);

//Comparing the value of Excel cell with all the project keywords

if(sActionKeyword.equals("openBrowser")){

//This will execute if the excel cell value is 'openBrowser'

//Action Keyword is called here to perform action

ActionKeywords.*openBrowser*();}

else if(sActionKeyword.equalsIgnoreCase("login")){

ActionKeywords.*logIn*();}

else if(sActionKeyword.equalsIgnoreCase("view")){//dashboard

ActionKeywords.*view*();}

else if(sActionKeyword.equalsIgnoreCase("post")){//postjob

ActionKeywords.*post*();}

else if(sActionKeyword.equalsIgnoreCase("jobRespond")){//Jobresponse

ActionKeywords.*jobRespond*();}

else if(sActionKeyword.equalsIgnoreCase("fetchResume")){//fetchresume

ActionKeywords.*fetchResume*();}

else if(sActionKeyword.equalsIgnoreCase("fetchJobs")){//Jobs

ActionKeywords.*fetchJobs*();}

else if(sActionKeyword.equalsIgnoreCase("myaccount")){//myaccount

ActionKeywords.*changeSettings*();}

else if(sActionKeyword.equalsIgnoreCase("closeBrowser")){//logout

ActionKeywords.*logout*();}

}

}

}

*package utility;*

import java.io.FileInputStream;

import org.apache.poi.xssf.usermodel.XSSFCell;

import org.apache.poi.xssf.usermodel.XSSFSheet;

import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class ExcelUtils {

// Keyword driven Framework

/\*\*

\* @author Theo

\*\*/

private static XSSFSheet ExcelWSheet;

private static XSSFWorkbook ExcelWBook;

private static XSSFCell Cell;

// This method is to set the File path and to open the Excel file

// Pass Excel Path and SheetName as Arguments to this method

public static void setExcelFile(String Path, String SheetName) throws Exception {

FileInputStream ExcelFile = new FileInputStream(Path);

ExcelWBook = new XSSFWorkbook(ExcelFile);

ExcelWSheet = ExcelWBook.getSheet(SheetName);

}

// This method is to read the test data from the Excel cell

// In this we are passing parameters/arguments as Row Num and Col Num

public static String getCellData(int RowNum, int ColNum) throws Exception {

Cell = ExcelWSheet.getRow(RowNum).getCell(ColNum);

String CellData = Cell.getStringCellValue();

return CellData;

}

}

*package keyword;*

import java.io.File;

import java.io.FileInputStream;

import java.io.IOException;

import java.util.Iterator;

import java.util.List;

import java.util.Properties;

import java.util.Set;

import java.util.concurrent.TimeUnit;

import org.apache.log4j.BasicConfigurator;

import org.apache.log4j.Logger;

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.support.ui.Select;

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

public class ActionKeywords {

public static WebDriverWait wait;

//public static final Logger log= Logger.getLogger(ActionKeywords.class.getName());

public static Logger log = Logger.getLogger("myLogger");

public static WebDriver d;

public static String workingDir = System.getProperty("user.dir");

//public static Logger log = Logger.getLogger("mylogger");

protected static WebElement xp(String xpath) {

return d.findElement(By.xpath(xpath));

}

protected static WebElement id(String id) {

return d.findElement(By.id(id));

}

protected static WebElement css(String cssSelector) {

return d.findElement(By.cssSelector(cssSelector));

}

public static void openBrowser() {

BasicConfigurator.configure();

System.setProperty("webdriver.chrome.driver",workingDir+ "\\Resources\\chromedriver.exe");

d = new ChromeDriver();

//d.manage().window().maximize();

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

System.out.println("open browser");

}

public static void logIn() throws Exception {

File file = new File(workingDir + "\\Resources\\Base.properties");

FileInputStream fileInput;

fileInput = new FileInputStream(file);

Properties prop = new Properties();

prop.load(fileInput);

// opening techfetch webiste

d.get(prop.getProperty("URL"));

log.info("Opening Techfetch webiste");

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

//click recuriterlogin

xp(prop.getProperty("Recuriterloginclick")).click();

log.info("Opening Recuriter menu");

//click login button

xp(prop.getProperty("loginbutton")).click();

log.info("Clicking Login button");

//switch to empcontentframe

d.switchTo().defaultContent();

d.switchTo().frame("empcontentframe");

wait = new WebDriverWait(d, 30);

//enter username

xp(prop.getProperty("MailidField")).sendKeys(prop.getProperty("username"));

log.info("Entering mailid");

//enter password

xp(prop.getProperty("PasswordField")).sendKeys(prop.getProperty("password"));

log.info("Entering password");

//click submit button

xp(prop.getProperty("submitbutton")).click();

log.debug("Clicking submit button");

String mytitle = "Top Jobs for Smart Talent | TechFetch.com";

if (!d.getTitle().equals(mytitle)) {

wait = new WebDriverWait(d, 30);

//alert accept

xp(".//\*[@id='alertify-ok']").click();

//enter password

xp(prop.getProperty("PasswordField")).sendKeys(prop.getProperty("password"));

//click submit button

xp(prop.getProperty("submitbutton")).click();

//d.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

}

System.out.println("login completed");

}

public static void view() throws Exception {

/\*

\* @author Sakthi Praveen Kumar.k work name : DashBoard

\*/

String mytitle = "Top Jobs for Smart Talent | TechFetch.com";

// System.out.println(d.getTitle());

if (d.getTitle().equals(mytitle)) {

d.navigate().refresh();

}

Properties prop;

FileInputStream fileInput = new FileInputStream(workingDir + "\\Resources\\dashBoard.properties");

prop = new Properties();

prop.load(fileInput);

//switch to iframe1

d.switchTo().defaultContent();

d.switchTo().frame("Iframe1");

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

//click availables of menu

css(prop.getProperty("avaliableAsof")).click();

log.info("Clicking Avaliable as of menu");

//get table boder

// To display all the element from the WebPage Table WebElement table =

xp(prop.getProperty("tableBorder"));

log.info("getting table border");

// To Display All The Resumes according To Month Using For Loop:

String[] months = new String[] { "Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov",

"Dec" };

for (int i = 0; i < 12; i++) {

// getting avaliableasof field

xp(prop.getProperty("avaliableAsofFieldbox")).sendKeys(months[i]);

log.info("getting avaliableasoffield");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

WebElement table1 = xp(prop.getProperty("tableBorder"));

log.info("getting tableborder");

List<WebElement> rows4 = table1.findElements(By.tagName("tr"));

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

System.out.println("Resumes" + months[i]);

log.info("printing resumes");

System.out.println(rows4.size() - 1);

log.info("printing rowsize");

xp(prop.getProperty("avaliableAsofFieldbox")).clear();

log.info("clearing avaliableasof field");

}

// Search by Name

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

//enter name to search

xp(prop.getProperty("searchBynameFieldbox")).sendKeys(prop.getProperty("searchBynameValue"));

log.info("getting fieldname");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

//clear the name field

xp(prop.getProperty("searchBynameFieldbox")).clear();

log.info("clearing fieldbox");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

// Search By Work Authorization

String[] workAuthorization = new String[] { "US Citizen", "Green Card", "US Security Clearance" };

for (int i = 0; i < 3; i++) {

xp(prop.getProperty("workAuthorizationfieldBox")).sendKeys(workAuthorization[i]);

log.info("getting workauthorization");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

//clear work authorization field

xp(prop.getProperty("workAuthorizationfieldBox")).clear();

log.info("clearing workauthorization");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

} // Search By Location(City or State)

//enter location search value

xp(prop.getProperty("locationFieldbox")).sendKeys(prop.getProperty("locationSearchvalue"));

log.info("getting location field");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

//clear location fieldbox

xp(prop.getProperty("locationFieldbox")).clear();

log.info("clearing locationfieldbox");

// Search By Relocation

String[] reLocation = new String[] { "Yes", "No" };

for (int i = 0; i < 2; i++) {

// enter relocation value

xp(prop.getProperty("relocationFieldbox")).sendKeys(reLocation[i]);

log.info("getting relocationfield");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

//clear relocation Fieldbox

xp(prop.getProperty("relocationFieldbox")).clear();

log.info("clearing relocationfield");

}

// search By Specialized Area

//enterspecializedAreavalue

xp(prop.getProperty("specializedAreafieldBox")).sendKeys(prop.getProperty("specializedAreasearchValue"));

log.info("getting specializedarea");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

// clear specialized Area fieldBox

xp(prop.getProperty("specializedAreafieldBox")).clear();

log.info("clearing specializedarea");

// Search By Experience

//enter experience value

xp(prop.getProperty("experienceFieldbox")).sendKeys(prop.getProperty("experienceSearchvalue"));

log.info("getting experiencefield");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

// clear experience Fieldbox

xp(prop.getProperty("experienceFieldbox")).clear();

log.info("clearing experiencefieldbox");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

// Search By Rate

//enter rate search value

xp(prop.getProperty("rateFieldbox")).sendKeys(prop.getProperty("rateSearchvalue"));

log.info("getting ratefieldbox");

wait = new WebDriverWait(d, 30);

// Profile Click

xp(prop.getProperty("profileClick")).click();

log.debug("clicking profile");

//window handling

String window1 = d.getWindowHandle();

System.out.println(window1);

log.debug("printing window1 output");

Set<String> windows = d.getWindowHandles();

Iterator iterator = windows.iterator();

String x;

while (iterator.hasNext()) {

x = iterator.next().toString();

System.out.println(x);

// for(String x:windows){

if (x.equals(window1) == false) {

//switch to new window

d.switchTo().window(x);

d.manage().window().maximize();

// click Email

Thread.sleep(1000);

id(prop.getProperty("email")).click();

log.debug("clicking email");

//switch to empchkResumecontentframe

d.switchTo().defaultContent();

d.switchTo().frame("empchkResumecontentframe");

//enter the subject value

xp(prop.getProperty("subject")).sendKeys(prop.getProperty("subjectValue"));

log.debug("getting subject");

//click send email

xp(prop.getProperty("emailSend")).click();

log.debug("clicking emailsend");

xp(prop.getProperty("emailNotificationclose")).click();

//close email notification

log.debug("clicking emailnotification");

wait = new WebDriverWait(d, 30);

//switch to new window

d.switchTo().defaultContent();

d.switchTo().window(x);

Thread.sleep(1000);

// click

//switch to empforwardprofilecontentframeForward

xp(prop.getProperty("forwardProfile")).click();

log.debug("clicking forward profile");

d.switchTo().defaultContent();

d.switchTo().frame("empforwardprofilecontentframe");

wait = new WebDriverWait(d, 30);

//enter to mail id address

xp(prop.getProperty("toMailid")).sendKeys(prop.getProperty("toMailidValue"));

log.debug("Getting mailid");

//click forward button

xp(prop.getProperty("forwardButton")).click();

log.debug("clicking forward button");

//close forward notification

xp(prop.getProperty("forwardClosebutton")).click();

log.debug("Clicking forward close");

//switch to default frame

d.switchTo().defaultContent();

//switch to window

d.switchTo().window(x);

Thread.sleep(1000);

// Add Feedback

xp(prop.getProperty("addFeedback")).click();

log.debug("Clicking add feedback");

//switch to employerfeedbackframe

d.switchTo().defaultContent();

d.switchTo().frame("employerfeedbackframe");

wait = new WebDriverWait(d, 30);

//click open To relocate

xp(prop.getProperty("openTorelocate")).click();

log.debug("Clicking opento relocate");

//click save and close

xp(prop.getProperty("save&Close")).click();

log.debug("Clicking Save button");

//switch to window

d.switchTo().defaultContent();

d.switchTo().window(x);

// d.close();

// Add notes

//switch to window

d.switchTo().defaultContent();

d.switchTo().window(x);

Thread.sleep(1000);

//click add note

xp(prop.getProperty("addNote")).click();

//switch to addnotesframe

d.switchTo().defaultContent();

Thread.sleep(1000);

d.switchTo().frame("addnotesframe");

//enter textbox value

xp(prop.getProperty("textBoxfield")).sendKeys(prop.getProperty("textBoxvalue"));

//click save and close notes

xp(prop.getProperty("save&Closenotes")).click();

//switch to window

d.switchTo().defaultContent();

d.switchTo().window(x);

d.close();

}

}

//switch to parent window

d.switchTo().window(window1);

System.out.println("end of view");

}

public static void post() throws Exception {

/\*\*

\* @author surya Balapriya work name : postjobs

\*/

String workingDir;

//fetching file from resources

workingDir = System.getProperty("user.dir");

File file = new File(workingDir + "\\Resources\\postJob.properties");

FileInputStream fileInput;

fileInput = new FileInputStream(file);

Properties prop = new Properties();

prop.load(fileInput);

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

d.switchTo().defaultContent();

// switching to default content

id(prop.getProperty("postJobs")).click();

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

// switching to default content

d.switchTo().defaultContent();

// switching to iframe 2

d.switchTo().frame("Iframe2");

wait = new WebDriverWait(d, 10);

id(prop.getProperty("jobTitleName")).clear();

//clear the value in the text box

log.info("clearing title");

id(prop.getProperty("jobTitleName")).sendKeys(prop.getProperty("jobTitle"));

log.info("getting title");

//sending title to job title box

// d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

id(prop.getProperty("requiredSkillsName")).clear();

//clearing the value in the text box

log.info("clearing skills");

id(prop.getProperty("requiredSkillsName")).sendKeys(prop.getProperty("requiredSkills"));

//sending skills name to text box

log.info("getting skills");

// wait = new WebDriverWait(d, 30);

//clearing the value in the box

id(prop.getProperty("preferredSkillName")).clear();

log.info("clearing preferredskills ");

// d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

id(prop.getProperty("preferredSkillName")).sendKeys(prop.getProperty("preferredSkill"));

//entering preferred skill in text box

log.info("getting preferredskills");

//clicking the experience level area

id(prop.getProperty("expLevel")).click();

log.info("clicking explevel");

//clearing the value in the box

id(prop.getProperty("jobDescriptionName")).clear();

log.info("clearing jobdescription");

//entering job description in text box

id(prop.getProperty("jobDescriptionName")).sendKeys(prop.getProperty("jobDescription"));

log.info("getting jobdescription");

// d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clearing the value in the box

id(prop.getProperty("clientName1")).clear();

log.info("clearing clientname");

id(prop.getProperty("clientName1")).sendKeys(prop.getProperty("clientName"));

//entering the client name

log.info("getting clientname");

// wait = new WebDriverWait(d, 30);

id(prop.getProperty("buttonClick1")).click();

//clicking the button

log.info("clicking button");

// wait = new WebDriverWait(d, 30);

id(prop.getProperty("cityName")).sendKeys(prop.getProperty("city"));

//entering city name

log.info("getting city");

Thread.sleep(1000);

//clicking the city in the list

css(prop.getProperty("cityList")).click();

log.info("clicking citylist");

wait = new WebDriverWait(d, 30);

new Select(id(prop.getProperty("jobExp1"))).selectByVisibleText("1 Week");

//selecting job experience from dropdown

log.info("selecting jobexp");

wait = new WebDriverWait(d, 30);

//clicking consulting button

id(prop.getProperty("consulting")).click();

log.info("clicking consulting");

//clicking work authorization button

id(prop.getProperty("workAuthori")).click();

log.info("clicking workauthor");

wait = new WebDriverWait(d, 30);

//click preferred experience

id(prop.getProperty("preferEmp")).click();

log.info("clicking preferemp");

// clearing the duration

id(prop.getProperty("durationName")).clear();

log.info("clearing duration");

//entering duration

id(prop.getProperty("durationName")).sendKeys(prop.getProperty("duration"));

log.info("getting duration");

// d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//selecting special area

Select select = new Select(xp(prop.getProperty("splArea")));

log.info("selecting splarea");

//selecting special skill

Select select1 = new Select(xp(prop.getProperty("splSkills")));

log.info("selecting splskills");

for (int i = 0; i <= 27; i++) {

select.selectByIndex(i);

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

}

select.selectByIndex(1);

select1.selectByIndex(1);

//clicking save button

id(prop.getProperty("saveBtn")).click();

log.info("clicking savebtn");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking others

id(prop.getProperty("others")).click();

log.info("clicking others");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking domain experience

id(prop.getProperty("domainExpo")).click();

log.info("domainexpo");

//clicking domain experience1

id(prop.getProperty("domainExpo1")).click();

log.info("getting domainexpo1");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking domain experience2

id(prop.getProperty("domainExpo2")).click();

log.info("getting domainexpo2");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking postmyjob button

id(prop.getProperty("postMyJob")).click();

log.info("clicking postmyjob");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking close button

xp(prop.getProperty("closeBtn")).click();

log.info("clicking closebutton");

//window switching to deffaultcontent

d.switchTo().defaultContent();

System.out.println("end of post ");

}

public static void jobRespond() throws Exception {

/\*\*

\* @author Santhalakshmi work name :Jobresponse

\*/

File file = new File(workingDir + "\\Resources\\jobresponse.properties");

FileInputStream fileInput;

fileInput = new FileInputStream(file);

Properties prop = new Properties();

prop.load(fileInput);

// jobresponse tab

css(prop.getProperty("jobResponse")).click();

d.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS);

d.switchTo().defaultContent();

d.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS);

d.switchTo().frame("Iframe4");

System.out.println("end of jobRespond");

}

public static void fetchResume() throws InterruptedException, IOException {

/\*\*

\* @author Karthigavalli work name :Fetch Resume

\*/

File file = new File(workingDir + "\\Resources\\fetchResume.properties");

FileInputStream fileInput;

fileInput = new FileInputStream(file);

Properties prop = new Properties();

prop.load(fileInput);

//Switch to default content

d.switchTo().defaultContent();

// clicking postmyjob

xp(prop.getProperty("fetch")).click();

log.info("clicking postmyjob");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//Switch to default content

d.switchTo().defaultContent();

d.switchTo().frame("Iframe5");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

// getting key

id(prop.getProperty("key")).sendKeys(prop.getProperty("keyWord"));

log.info("getting key");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

// clicking all

id(prop.getProperty("all")).click();

log.info("clicking all");

// clicking workAuthority

id(prop.getProperty("workAuthority")).click();

log.info("clicking workAuthority");

// clicking workAuthority1

id(prop.getProperty("workAuthority1")).click();

log.info("clicking workAuthority1");

// clicking workAuthority2

id(prop.getProperty("workAuthority2")).click();

log.info("clicking workAuthority2");

// clicking workAuthority3

id(prop.getProperty("workAuthority3")).click();

log.info("clicking workAuthority3");

// clicking preferedEmployment

id(prop.getProperty("preferedEmployment")).click();

log.info("clicking preferedEmployment");

//clicking preferedEmployment1

id(prop.getProperty("preferedEmployment1")).click();

log.info("clicking preferedEmployment1");

// clicking preferedEmployment2

id(prop.getProperty("preferedEmployment2")).click();

log.info("clicking preferedEmployment2");

// clicking preferedEmployment3

id(prop.getProperty("preferedEmployment3")).click();

log.info("clicking preferedEmployment3");

// clicking preferedEmployment4

id(prop.getProperty("preferedEmployment4")).click();

log.info("clicking preferedEmployment4");

// clicking workAuthority

id(prop.getProperty("workAuthority")).click();

log.info("clicking workAuthority");

// clicking workAuthority1

id(prop.getProperty("workAuthority1")).click();

log.info("clicking workAuthority1");

// clicking preferedEmployment

id(prop.getProperty("preferedEmployment")).click();

log.info("clicking preferedEmployment ");

// clicking prefered employment1

id(prop.getProperty("preferedEmployment1")).click();

log.info("clicking prefered employment1");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

// getting area

new Select(id(prop.getProperty("area"))).selectByVisibleText("Java, J2EE");

log.info("getting area");

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

// getting skill

new Select(id(prop.getProperty("skill"))).selectByVisibleText("Java Server Side");

log.info("getting skill");

Thread.sleep(1000);

// clicking advanced search

id(prop.getProperty("advancedSearch")).click();

log.info("clicking advanced search");

Thread.sleep(1000);

// clicking domain1

xp(prop.getProperty("domain1")).click();

log.info("clicking domain1");

// clicking domain2

xp(prop.getProperty("domain2")).click();

log.info("clicking domain2");

// clicking country

id(prop.getProperty("country")).click();

log.info("clicking country");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

// getting degree

new Select(id(prop.getProperty("degree"))).selectByVisibleText("Bachelor");

log.info("getting degree");

// getting state

Select select = new Select(id(prop.getProperty("state")));

log.info("getting state");

List<WebElement> l = select.getOptions();

for (int i = 0; i < 6; i++) {

select.selectByIndex(i);

// clicking arrow

xp(prop.getProperty("arrow")).click();

log.info("clicking arrow");

}

// clicking experience

id(prop.getProperty("experience")).click();

log.info("clicking experience");

//clicking usExperience

id(prop.getProperty("usExperience")).click();

log.info("clicking usExperience");

//getting a skill1

id(prop.getProperty("aSkill1")).sendKeys(prop.getProperty("skill1"));

log.info("getting askill1");

// getting year1

new Select(id(prop.getProperty("year1"))).selectByVisibleText("4");

log.info("getting year1");

//getting a skill2

id(prop.getProperty("aSkill2")).sendKeys(prop.getProperty("skill1"));

log.info("getting askill2");

// getting year2

new Select(id(prop.getProperty("year2"))).selectByVisibleText("2");

log.info("getting year2");

// getting a skill3

id(prop.getProperty("aSkill3")).sendKeys(prop.getProperty("skill1"));

log.info("getting askill3");

// getting year

new Select(id(prop.getProperty("year3"))).selectByVisibleText("4");

log.info("getting year");

// getting save

xp(prop.getProperty("save")).sendKeys(prop.getProperty("fetching"));

log.info("getting save");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

// getting days

new Select(id(prop.getProperty("days"))).selectByVisibleText("90 Days");

log.info("getting days");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

// clicking fetched

id(prop.getProperty("fetched")).click();

log.info("clicking fetched");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//Switch to default content

d.switchTo().defaultContent();

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

// clicking matched resume

xp(prop.getProperty("tabClose")).click();

log.info("clicking logg");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking USA resume

xp(prop.getProperty("tabClose1")).click();

log.info("clicking log");

System.out.println("end of fetchResume");

}

public static void fetchJobs() throws Exception {

/\*\*

\* @author DurgaDevi work name :Jobs

\*/

File file = new File(workingDir + "\\Resources\\myJob.properties");

FileInputStream fileInput;

fileInput = new FileInputStream(file);

Properties prop = new Properties();

prop.load(fileInput);

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

d.switchTo().defaultContent();

//clicking the jobs icon

css(prop.getProperty("jobMenu")).click();

log.info("clicking jobmenu");

wait = new WebDriverWait(d, 30);

//switching to Iframe7

d.switchTo().defaultContent();

d.switchTo().frame("Iframe7");

//enter the updated date

xp(prop.getProperty("updatedDate")).sendKeys(prop.getProperty("month"));

log.info("getting updatedate");

//enter the job title

xp(prop.getProperty("jobTitle")).sendKeys(prop.getProperty("lang"));

log.info("getting jobtitle");

//enter the location

xp(prop.getProperty("location")).sendKeys(prop.getProperty("cty4"));

log.info("clicking location");

wait = new WebDriverWait(d, 30);

//click the preview option

xp(prop.getProperty("preview")).click();

log.info("clicking preview");

//switching to the child window from the parent window

String window1 = d.getWindowHandle();

System.out.println(window1);

log.debug("printing window1 output");

Set<String> windows = d.getWindowHandles();

Iterator iterator = windows.iterator();

String x;

while (iterator.hasNext()) {

x = iterator.next().toString();

System.out.println(x);

// for(String x:windows){

if (x.equals(window1) == false) {

d.switchTo().window(x);

Thread.sleep(5000);

//click the apply now icon

id(prop.getProperty("applyNow")).click();

log.info("clicking applynow");

//switch to the frame

d.switchTo().defaultContent();

d.switchTo().frame("applyemailframe");

wait = new WebDriverWait(d, 30);

//select the candidate from the profile

new Select(d.findElement(By.id(prop.getProperty("selectFromProfile"))))

.selectByVisibleText("Mithilesh Joshi (EXP) [Active]");

log.info("getting selectfromprofile");

//click the option value

// xp(prop.getProperty("optionValue")).click();

// log.info("clicking optionvalue");

//clear the cc

xp(prop.getProperty("cc")).clear();

log.info("clearing cc");

//enter the cc

xp(prop.getProperty("cc")).sendKeys(prop.getProperty("tooo"));

log.info("getting cc");

//click the covering letter

xp(prop.getProperty("coveringLetter")).click();

log.info("clicking coveringletter");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clear the uploaded file

id(prop.getProperty("fileUpload")).clear();

log.info("clearing fileupload");

// d.findElement(By.xpath(".//\*[@id='uploaddiv']/div/div/div")).click();

//fetch the resume from the resources file

id(prop.getProperty("fileUpload")).sendKeys(workingDir + "\\Resources\\ram.docx");

log.info("getting file");

//click the send email button

// d.findElement(By.id("uploadvalue2")).sendKeys("C:\\Users\\mag8\\Desktop\\res.docx");

id(prop.getProperty("sendEmail")).click();

log.info("clicking sendmail");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//click the forward job icon

//switch to the frame

d.switchTo().defaultContent();

d.switchTo().frame("applyemailframe");

//click the close button

xp(prop.getProperty("btnClose")).click();

log.info("clicking btnclose");

// forward

Thread.sleep(5000);

d.switchTo().defaultContent();

d.switchTo().window(x);

//click the forward job icon

//switch to the frame

xp(prop.getProperty("forwardJob")).click();

log.info("clicking fwdjob");

d.switchTo().defaultContent();

d.switchTo().frame("jsfwdjobcontentframe");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//enter the address

xp(prop.getProperty("toAddr")).sendKeys(prop.getProperty("tooo"));

log.info("getting toaddr");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//click the send mail

xp(prop.getProperty("sendMail")).click();

log.info("clicking sendmail");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//closing the driver

d.close();

}

}

// Edit

/\*

\* @author surya Balapriya work name : edit

\*/

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

d.switchTo().window(window1);

css(prop.getProperty("jobMenu")).click();

log.info("clicking jobmenu");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

d.switchTo().defaultContent();

d.switchTo().frame("Iframe7");

//click the edit option

xp(prop.getProperty("edit")).click();

log.info("clicking edit");

d.switchTo().defaultContent();

d.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

//switch to the Iframe8

d.switchTo().frame("Iframe8");

//click the job title

id(prop.getProperty("jobTitle5")).clear();

log.info("clicking jobtitle5");

//enter the job title name

id(prop.getProperty("jobTitle5")).sendKeys(prop.getProperty("titleName"));

log.info("getting jobtitle5");

d.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

//clear the skill required

id(prop.getProperty("skillReq")).clear();

log.info("clicking skillreq");

//enter the required skill

id(prop.getProperty("skillReq")).sendKeys(prop.getProperty("titleName"));

log.info("getting skillreq");

//click the experience level

id(prop.getProperty("exLevel")).click();

log.info("clicking exlevel");

//click the experience level1

id(prop.getProperty("exLevel1")).click();

log.info("clicking exlevel1");

//click the experience level2

id(prop.getProperty("exLevel2")).click();

log.info("clicking exLevel2");

//click the experience level3

id(prop.getProperty("exLevel3")).click();

log.info("clicking exlevel3");

//click the experience level4

id(prop.getProperty("exLevel4")).click();

log.info("clicking exlevel4");

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

//click the type of parameter

//click the work option

xp(prop.getProperty("permanentType")).click();

log.info("clicking Parametertype");

id(prop.getProperty("work1")).click();

log.info("clicking work1");

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

//click the preferred employment

id(prop.getProperty("preferredEmployment5")).click();

log.info("clicking PreferredEmployment5");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//select the specialized area

new Select(id(prop.getProperty("splArea1"))).selectByVisibleText("Java, J2EE");

log.info("getting splarear1");

//select the specialized skill

new Select(id(prop.getProperty("splSkill1"))).selectByVisibleText("Java, J2EE");

log.info("getting splskill1");

//click the domain name

id(prop.getProperty("domainList0")).click();

log.info("clicking domainlist0 ");

//click the domain name

id(prop.getProperty("domainList2")).click();

log.info("clicking domianlist2");

//click the preview option

id(prop.getProperty("preview1")).click();

log.info("clicking preview1");

Thread.sleep(1000);

//click the close button

d.findElement(By.linkText(prop.getProperty("closeButton"))).click();

log.info("clicking clsbutton");

Thread.sleep(1000);

d.switchTo().defaultContent();

//close the match

xp(prop.getProperty("matchClose")).click();

log.info("clicking matchclose");

//close the jobs window

xp(prop.getProperty("jobClose")).click();

log.info("clicking jobclose");

System.out.println("end of fetchjobs");

}

public static void changeSettings() throws Exception {

/\*\*

\* @author RamKumar Sastha work name :My Account

\*/

File file = new File(workingDir + "\\Resources\\myAccount.properties");

FileInputStream fileInput;

fileInput = new FileInputStream(file);

Properties prop = new Properties();

prop.load(fileInput);

d.switchTo().defaultContent();

//click my account icon

xp(prop.getProperty("myAccount")).click();

log.info("clicking myaccount");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//switch to Iframe9

d.switchTo().defaultContent();

d.switchTo().frame("Iframe9");

//clear the texted mail

xp(prop.getProperty("txt\_Mail")).clear();

log.info("clearing txtmail");

//enter the first name

xp(prop.getProperty("txt\_Mailid")).sendKeys(prop.getProperty("userName"));

log.info("getting txtmail");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

xp(prop.getProperty("firstName")).clear();

log.info("clearing firstname ");

xp(prop.getProperty("firstName")).sendKeys(prop.getProperty("firstNamevalue"));

log.info("getting firstname");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

xp(prop.getProperty("lastName")).clear();

log.info("clearing lastname");

//enter the last name

xp(prop.getProperty("lastName")).sendKeys(prop.getProperty("lastNamevalue"));

log.info("getting lastname");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clear the company name

xp(prop.getProperty("companyName")).clear();

log.info("clearing companyname");

//enter company name

xp(prop.getProperty("companyName")).sendKeys(prop.getProperty("companyNamevalue"));

log.info("getting companyname");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//enter the country name

xp(prop.getProperty("country")).sendKeys(prop.getProperty("countryValue"));

log.info("getting country");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clear the city name

xp(prop.getProperty("city")).clear();

log.info("clearing city");

//enter the city name

xp(prop.getProperty("city")).sendKeys(prop.getProperty("cityValue"));

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clear the personal id

xp(prop.getProperty("personalId")).clear();

log.info("clearing personalid");

//enter the personal mailid

xp(prop.getProperty("personalId")).sendKeys(prop.getProperty("personalIdvalue"));

log.info("getting personalid");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//select the area of phone number which we are going to provide

xp(prop.getProperty("ph\_Area")).sendKeys(prop.getProperty("ph\_Areavalue"));

log.info("getting ph area");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clear the given phone number

xp(prop.getProperty("ph\_No")).clear();

log.info("clearing phno");

//enter the phone number

xp(prop.getProperty("ph\_No")).sendKeys(prop.getProperty("ph\_Novalue"));

log.info("getting phno");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//click the update button

xp(prop.getProperty("update")).click();

log.info("clicking update");

Thread.sleep(5000);

//switching to default content window

d.switchTo().defaultContent();

//click the password from the menu

css(prop.getProperty("passClick")).click();

log.info("clicking Passclick");

//switching to Iframe10

d.switchTo().defaultContent();

d.switchTo().frame("Iframe10");

//enter the new password

xp(prop.getProperty("passReset")).sendKeys(prop.getProperty("password"));

log.info("getting passreset");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//enter the new password again for confirmation

css(prop.getProperty("passReset1")).sendKeys(prop.getProperty("password1"));

log.info("getting passreset1");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//click the update button

xp(prop.getProperty("resetButton")).click();

log.info("clicking resetbutton");

Thread.sleep(5000);

//switch to the default content window

d.switchTo().defaultContent();

//click the preferences icon from the menu

css(prop.getProperty("preference")).click();

log.info("clicking preferences");

//switch to the Iframe11

d.switchTo().defaultContent();

d.switchTo().frame("Iframe11");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking the radio button to select yes/no

css(prop.getProperty("radio2")).click();

log.info("clicking radio2");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking the radio button to select yes/no

css(prop.getProperty("radio1")).click();

log.info("clicking radio1");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking the radio button to select yes/no

css(prop.getProperty("radio4")).click();

log.info("clicking radio4");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking the radio button to select yes/no

css(prop.getProperty("radio3")).click();

log.info("clicking radio3");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking the radio button to select yes/no

css(prop.getProperty("radio6")).click();

log.info("clicking radio6");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

css(prop.getProperty("radio5")).click();

log.info("clicking radion5");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//clicking the radio button to select yes/no

css(prop.getProperty("radio\_Button")).click();

log.info("clicking radiobutton");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

//switch to the default content frame

d.switchTo().defaultContent();

//click the update button

xp(prop.getProperty("user")).click();

log.info("clicking user");

d.manage().timeouts().implicitlyWait(40, TimeUnit.SECONDS);

}

public static void logout() throws Exception {

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

//switch to defaultframe

d.switchTo().defaultContent();

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

File file = new File(workingDir + "\\Resources\\Base.properties");

FileInputStream fileInput;

fileInput = new FileInputStream(file);

Properties prop = new Properties();

prop.load(fileInput);

// Logout...

d.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

//click logout button

xp(prop.getProperty("logoutbutton")).click();

log.debug("Clicking logout button ");

System.out.println("end of logout");

d.quit();

}

}

***Properties files***

***Base.Properties***

*URL=http://uat.techfetch.com*

*username=trainee3@tiliconveli.com*

*password=TV@345*

*Recuriterloginclick=.//\*[@id='ucHeaderCtrl\_divRecruiter']/a*

*loginbutton=.//\*[@id='emploginpop']*

*MailidField=.//\*[@id='txtemailid']*

*PasswordField=.//\*[@id='txtpwd']*

*submitbutton=.//\*[@id='btnSubmit']*

*Recuritersetting=#WelcomeUser*

*logoutbutton=.//\*[@id='L*ogoutMTLTaskName']

***view.properties***

*avaliableAsof=.header-UpdDate>a*

*tableBorder=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/tbody*

*avaliableAsofFieldbox=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[1]/div/input*

*searchBynameFieldbox=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[2]/div/input*

*searchBynameValue=Raja Raj*

*workAuthorizationfieldBox= .//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[3]/div/input*

*locationFieldbox=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[4]/div/input*

*locationSearchvalue=Asbury, NJ*

*relocationFieldbox=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[5]/div/input*

*specializedAreafieldBox=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[6]/div/input*

*specializedAreasearchValue=Java*

*experienceFieldbox=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[7]/div/input*

*experienceSearchvalue=15*

*rateFieldbox=.//\*[@id='emp-col9']/div/div[2]/div[3]/div/div/div[2]/table/thead/tr[2]/td[8]/div/input*

*rateSearchvalue=Market*

*profileClick=.//\*[@id='JS6014\_16']/img*

*email=html/body/form/div[42]/div/div[3]/div[1]/div/div/div[1]/div/div/div/div/div[1]/div/div[2]/input[1]*

*subject=.//\*[@id='tbEmailSub']*

*subjectValue=Hai this is Updated Resume*

*emailSend=.//\*[@id='Submit1']*

*emailNotificationclose=.//\*[@id='btnPnlClose']*

*forwardProfile=.//\*[@id='forwardres']*

*toMailid=.//\*[@id='txtTo']*

*toMailidValue=sakthipraveen.k91@gmail.com*

*forwardButton=.//\*[@id='btnsendmail']*

*forwardClosebutton=.//\*[@id='btnPnlClose']*

*addFeedback=.//\*[@id='divfeedback']/a*

*openTorelocate=.//\*[@id='7']*

*save&Close=.//\*[@id='Button1']*

*addNote=.//\*[@id='notesarea']/a*

*textBoxfield=.//\*[@id='TxtNotes']*

*textBoxvalue=Hi Buddy*

*save&Closenotes=.//\*[@id='savenotes']*

***fetchResume.properties***

*fetch=.//\*[@id='LeftMenu11']/div/img*

*key=txtKeywords*

*keyWord=java*

*all=rbKeywordsAll*

*workAuthority=chkWorkAuthorization\_0*

*workAuthority1=chkWorkAuthorization\_1*

*workAuthority2=chkWorkAuthorization\_2*

*workAuthority3=chkWorkAuthorization\_3*

*preferedEmployment=chkPreferredEmployment\_0*

*preferedEmployment1=chkPreferredEmployment\_1*

*preferedEmployment2=chkPreferredEmployment\_2*

*preferedEmployment3=chkPreferredEmployment\_3*

*preferedEmployment4=chkPreferredEmployment\_4*

*area=ddlSpecializedArea*

*skill=ddlSpecializedSkill*

*advancedSearch=li1*

*domain1=.//\*[@id='chkBusinessDomain\_0']*

*domain2=.//\*[@id='chkBusinessDomain\_9']*

*country=ddlCountry*

*degree=ddlDegree*

*state=lbStates*

*arrow=.//\*[@id='AdvSearchOptions']/table/tbody/tr[2]/td/table[2]/tbody/tr[2]/td[2]/input[1]*

*experience=cbExpAny*

*usExperience=cbUSExpAny*

*aSkill1=txtSkill1*

*skill1=java*

*year1=ddlYears1*

*aSkill2=txtSkill2*

*skill2=dot net*

*year2=ddlYears2*

*aSkill3=txtSkill3*

*skill3=SQL Server*

*year3=ddlYears3*

*save=.//\*[@id='txtSearchName']*

*fetching=profilejob*

*days=ddlResumeDays*

*fetched=btnAdvancedSearch*

*tabClose=.//\*[@id='tabs']/ul/li[3]/span*

*tabClose1=.//\*[@id='tabs']/ul/li[2]/span*

***Job.properties***

*jobMenu=#LeftMenu10 > div.divLeftMenuTemplate.rc7 > img  
updatedDate=.//\*[@id='tblouterdiv']/table/thead/tr[2]/td[2]/div/input  
jobTitle=.//\*[@id='tblouterdiv']/table/thead/tr[2]/td[3]/div/input  
location= .//\*[@id='tblouterdiv']/table/thead/tr[2]/td[4]/div/input  
preview= .//\*[@id='previewjob33485']/img  
applyNow= quickapply  
selectFromProfile=lstResume  
optionValue=option[value=\\"DP5637\_7\\"]  
cc=html/body/form/div[3]/div[2]/div[1]/div/div[3]/div[2]/input  
coveringLetter=html/body/form/div[3]/div[2]/div[1]/div/div[7]/div[2]/div[1]/input  
fileUpload=fileCL  
sendEmail=btnsendmail  
btnClose= .//\*[@id='btnClose1']  
forwardJob= .//\*[@id='forwardjob']  
toAddr= .//\*[@id='txtTo']  
sendMail= .//\*[@id='btnsendmail']  
closeBtn= .//\*[@id='btnClose2']  
fwd=.//\*[@id='tblouterdiv']/table/tbody/tr[2]/td[9]/a/img  
toAddr= .//\*[@id='txtTo']  
sendMail= .//\*[@id='btnSend']  
forwardMain= //iframe[@id='forwardjobframe']  
month=Aug 28 2017  
lang=java  
cty4=Calabasas, CA*[*tooo=training2@tiliconveli.com*](mailto:tooo=training2@tiliconveli.com) *path8= C:\\\\Users\\\\mag8\\\\Desktop\\\\res.docx  
month8=Aug 28 2017  
lang8=java  
cty8=Calabasas, CA  
titleName=java  
edit= .//\*[@id='edt']  
jobTitle5=txtjobtitle2  
skillReq=txtsklrequired  
exLevel=chksenior  
exLevel1=chkmidle  
exLevel2=chkjunior  
exLevel3=chkconsulting  
contract= #txtDuration  
exLevel4= chkh1b  
permanentType= .//\*[@id='chkpermanenthire']  
work1=chkWorkAuthorization\_5  
preferredEmployment5=chkPreferredEmployment\_5  
preferredEmployment0=chkPreferredEmployment\_0  
splArea1=ddlsplarea  
splSkill1=ddlsplskill  
domainList0=chkdomainlist\_0  
domainList2=chkdomainlist\_2  
preview1=preview  
closeButton=Close  
updateButton=butupdate  
matchClose= .//\*[@id='tabs']/ul/li[3]/span  
jobClose= .//\*[@id='tabs']/ul/li[2]/span*

***jobresponse.properties***

*jobResponse=#AppliedResumes123HL*

*resumeTitle=ddlResumeTitle*

*resumeLink=.emp-gridtextlink*

*email=.//\*[@id='emlchkavail']*

*subject=.//\*[@id='tbEmailSub']*

*subjectValue=Hai this is Updated Resume*

*emailSend=.//\*[@id='Submit1']*

*emailNotificationclose=.//\*[@id='btnPnlClose']*

*forwardProfile=.//\*[@id='forwardres']*

*toMailid=.//\*[@id='txtTo']*

*toMailidValue=sakthipraveen.k91@gmail.com*

*forwardButton=.//\*[@id='btnsendmail']*

*forwardClosebutton=.//\*[@id='btnPnlClose']*

***myAccount.properties***

*userName=trainee3@tiliconveli.com*

*passWord=TV@345*

*txt\_Mail=.//\*[@id='txtemailid']*

*txt\_Mailid=.//\*[@id='txtemailid']*

*firstName=.//\*[@id='txtFName']*

*firstNamevalue=Trainee*

*lastName=.//\*[@id='txtLName']*

*lastNamevalue=Three*

*companyName=.//\*[@id='txtComname']*

*companyNamevalue=Techpark*

*country=.//\*[@id='ddsCountry']*

*country1=India*

*city=.//\*[@id='txtCity']*

*cityValue=Mumbai, Maharashtra*

*personalId=.//\*[@id='txtPersonalEmailID']*

*personalIdvalue=qatraining1@tiliconveli.com*

*ph\_Area=.//\*[@id='txtPhoneArea']*

*ph\_Areavalue=+91*

*ph\_No=.//\*[@id='txtPhone']*

*ph\_Novalue="9487525100"*

*update=.//\*[@id='btnUpdate']*

*passClick=#PasswordHL*

*passReset=.//\*[@id='txtResetPwd']*

*passReset1=#txtConPwd*

*resetButton=.//\*[@id='btnResetPwd']*

*preference=#PreferencesHL*

*radio1=#radioHotResumes\_0*

*radio2=#radioHotResumes\_1*

*radio3=#radioHotRequirement\_0*

*radio4=#radioHotRequirement\_1*

*radio5=#radioIM\_0*

*radio6=#radioIM\_1*

*radio\_Button=#btnUpdate*

*user=html/body/div[6]/div[1]/div[9]*

*fb=#PreferencesHL*

*myAccount=.//\*[@id='LeftMenu6']/div/div*

***myJob.properties***

*jobMenu=#LeftMenu10 > div.divLeftMenuTemplate.rc7 > img*

*updatedDate=.//\*[@id='tblouterdiv']/table/thead/tr[2]/td[2]/div/input*

*jobTitle=.//\*[@id='tblouterdiv']/table/thead/tr[2]/td[3]/div/input*

*location=.//\*[@id='tblouterdiv']/table/thead/tr[2]/td[4]/div/input*

*preview=.//\*[@id='previewjob33485']/img*

*applyNow=html/body/form/div[42]/div/div/div/div/div/div/div[1]/div/div/div/div/div[1]/div/div[2]/input[1]*

*selectFromProfile=lstResume*

*optionValue=option[value=\\"DP5637\_7\\"]*

*cc=html/body/form/div[3]/div[2]/div[1]/div/div[3]/div[2]/input*

*coveringLetter=html/body/form/div[3]/div[2]/div[1]/div/div[7]/div[2]/div[1]/input*

*fileUpload=.btn.btn-upload.btn-file*

*sendEmail=btnsendmail*

*btnClose=.//\*[@id='btnClose1']*

*forwardJob=.//\*[@id='forwardjob']*

*toAddr=.//\*[@id='txtTo']*

*sendMail=.//\*[@id='btnsendmail']*

*closeBtn=.//\*[@id='btnClose2']*

*fwd=.//\*[@id='tblouterdiv']/table/tbody/tr[2]/td[9]/a/img*

*toAddr=.//\*[@id='txtTo']*

*sendMail=.//\*[@id='btnSend']*

*forwardMain=//iframe[@id='forwardjobframe']*

*month= Aug 28 2017*

*lang=java*

*cty4=Calabasas, CA*

*tooo=training2@tiliconveli.com*

*path8=C:\\\\Users\\\\mag8\\\\Desktop\\\\res.docx*

*month8=Aug*

*lang8=c++*

*cty8=Calabasas, CA*

*titleName=java*

*edit=.//\*[@id='edt']*

*jobTitle5=txtjobtitle2*

*skillReq=txtsklrequired*

*exLevel=chksenior*

*exLevel1=chkmidle*

*exLevel2=chkjunior*

*exLevel3=chkconsulting*

*contract=#txtDuration*

*exLevel4=chkh1b*

*permanentType=.//\*[@id='chkpermanenthire']*

*work1=chkWorkAuthorization\_5*

*preferredEmployment5=chkPreferredEmployment\_5*

*preferredEmployment0=chkPreferredEmployment\_0*

*splArea1=ddlsplarea*

*splSkill1=ddlsplskill*

*domainList0=chkdomainlist\_0*

*domainList2=chkdomainlist\_2*

*preview1=preview*

*closeButton=Close*

*updateButton=butupdate*

*matchClose=.//\*[@id='tabs']/ul/li[3]/span*

*jobClose=.//\*[@id='tabs']/ul/li[2]/span*

***postJob.properties***

*//send keys*

*jobTitle=java*

*requiredSkills=java*

*preferredSkill=java*

*jobDescription=available*

*clientName=stephen*

*city=Calabasas, CA*

*duration=1year*

*//xpath*

*postJobs=PostJobs123MTLTaskName*

*jobTitleName=txtTitle*

*requiredSkillsName=txtsklrequired*

*preferredSkillName=txtprefsklrequired*

*expLevel=chkarchitect*

*jobDescriptionName=txtDescription*

*clientName1=txtClientName*

*buttonClick1=btnClick*

*cityName=txtcity*

*cityList=li*

*jobExp1=ddljobexpiry*

*consulting=chkconsulting*

*workAuthori=chkWorkAuthorization\_4*

*preferEmp=chkPreferredEmployment\_5*

*durationName=txtDuration*

*splArea=.//\*[@id='ddlsplarea']*

*splSkills=.//\*[@id='ddlsplskill']*

*saveBtn=savedetailbtn*

*others=rdyes*

*domainExpo=chkdomainlist\_0*

*domainExpo1=chkdomainlist\_2*

*domainExpo2=chkdomainlist\_8*

*postMyJob=btnpostjob*

*closeBtn=.//\*[@id='tabs']/ul/li[2]/span*

***Log4j.Properties***

*#Define loggers*

log4j.rootLogger=INFO,file

#Define rolling file appender

log4j.appender.file=org.apache.log4j.RollingFileAppender

log4j.appender.file.File=E:\\EclipseOxygenWorkSpace\\RecruiterPa\\src\\selenium.html

log4j.appender.file.Append=true

log4j.appender.file.ImmediateFlush=true

log4j.appender.file.MaxFileSize=500MB

log4j.appender.file.MaxBackupIndex=5

log4j.appender.file.layout=org.apache.log4j.HTMLLayout

log4j.appender.file.layout.ConversionPattern=%d %d{Z} [%t] %-5p (%F:%L) - %m%n

#Define loggers

log4j.logger.Logger=INFO, file1

log4j.appender.file1=org.apache.log4j.RollingFileAppender

log4j.appender.file1.File=E:\\EclipseOxygenWorkSpace\\RecruiterPa\\src\\manual.html

log4j.appender.file1.Append=true

log4j.appender.file1.ImmediateFlush=true

log4j.appender.file1.MaxFileSize=500MB

log4j.appender.file1.MaxBackupIndex=5

log4j.appender.file1.layout=org.apache.log4j.HTMLLayout

log4j.appender.file1.layout.ConversionPattern=%d %d{Z} [%t] %-5p (%F:%L) - %m%n