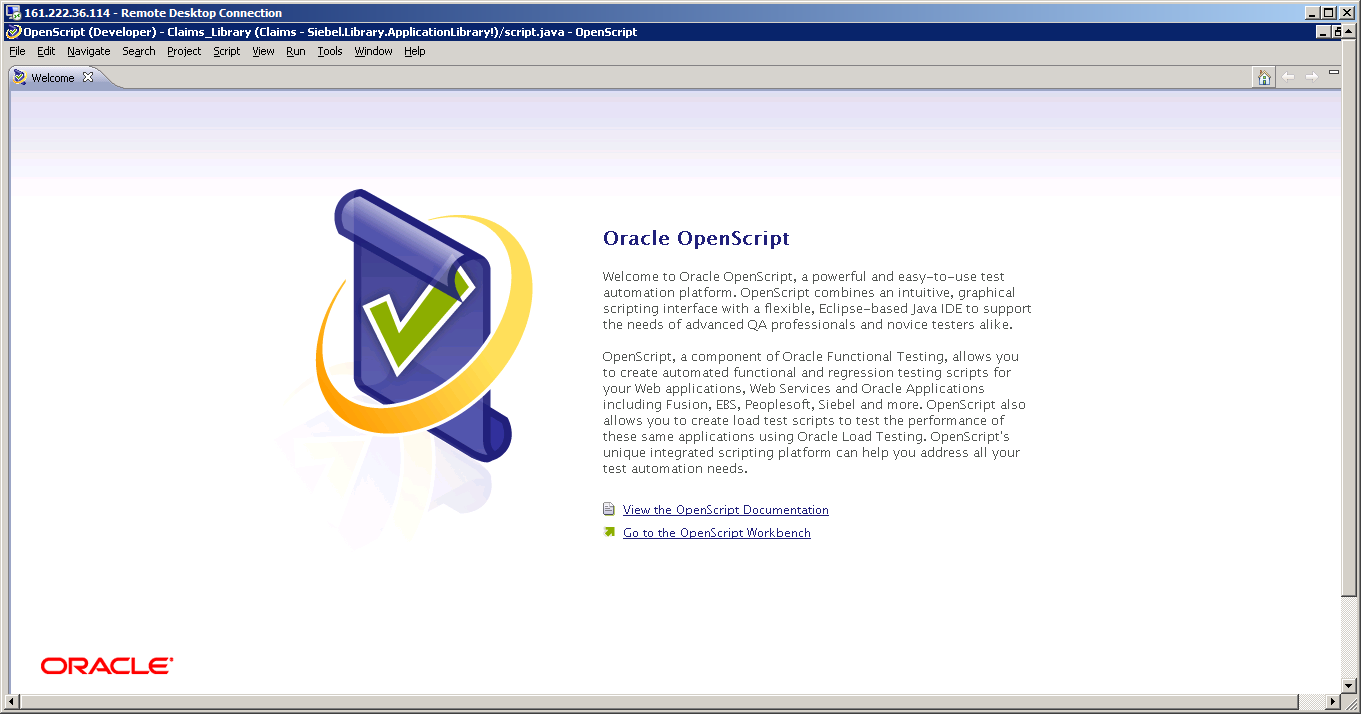
# A Help to programming in OpenScript



It is based upon the popular Eclipse IDE, and allows users/testers to record testing scripts (both load testing and functional testing) for a number of popular technologies.

Types of applications recognized by OATS:

|  |  |
| --- | --- |
| **Application type** | **Object** |
| Siebel | siebelFT |
| Web | Web |
| Oracle forms | Forms |

How to create a project?

Steps to create a new project:

1. Click on file->new->functional testing ->Siebel/web/oracle forms
2. Name the script

Create a repository:

1. Click on tools->Manage repository->Add
2. Enter the path and name

Open an existing script:

1. Click File->Open->select the file

Delete an existing script:

1. Click on Tools->Manage scripts
2. Select the script->Click delete

Edit repository path

1. Click on Tools->Manage repositories
2. Select the repository->Edit
3. Edit the repository path/name

**Working in Siebel applications**

1. **Table in Siebel**

There is no table concept available in Siebel.

**List** behaves like table here. We can get the no: of records available and can also read those values by using the list properties

siebelFT.list(<path>).getRowsCount ();

//will retrieve the no:of available rows.

siebelFT.list(<path>).activate(<rowno>);

// will activate the row no: specified and hence can read the value of text/picklistt available in the row

Code snippet:

int intNoOfRows = siebelFT.list(<Path>).getRowsCount();

if (intNoOfRows >0){

for(int intRowNo=0;intCount< intNoOfRows; intRowNo ++){

siebelFT.list(<Path>).activateRow(intRowNo);

siebelFT.picklist<Path>).select("Open");

}

1. **Reading value from a text/textArea or element**

siebelFT.text(<path>).getattribute(“text”);

or

siebelFT.text(<path>).gettext();

or

siebelFT.text(<path>).gettext();

//The above functions are available for textbox also.

1. **Reading the value slected in a siebList**

siebelFT.picklist (<path>).getActiveItem();

or

sieblFT.picklist(<path>)..getattribute(“activeitem”);

siebelFT.text(<path>).gettext();

1. **Logout from Siebel**

There is no log out button/link available in Siebel. This causes an issue when switching from one user to another. Even after logging in with a different credential, it still goes to the previous user’s account.

One method is to navigate to the log in screen of Siebel and use the following code.

web.clearSessionCookies();

This will clear the session cookies, so you can proceed to log in under a different username.

1. **Link in Siebel**

There is no link as such in Siebel. Instead, DrillDownColumn is available.

**D**rillDownColumn is equivalent to link.click in web.

siebelFT.list(<path>)

.drillDownColumn("Claim Number", intRowNo);

// intRowNo denotes the row number on which the link is present.

**Working in Web applications**

1. **Read the value slected in a selectbox**

String[] strValue= Web.slectbox(<path>).getSelectedText();

// strValue[0] gives the selected value of the selectBox

1. **Select value in a select box**

web.selectBox(<path>)

.selectOptionByText(<value>);

1. **Read value from text/textbox/element**

web.text(<path>).getattribute(“text”);

1. **Avoid recognizing the incorrect buttons/links/text**

To avoid the incorrect identification of fields/buttons specify the object property that has the highest weightage.

E.g.

"/web:window[@index='0' or @title='Siebel Call Center']/web:document[@index='12' ]" +

"/web:button[@value=’NEXT’ or @index=’2’]”

Can be modified as:

"/web:window[@index='0' or @title='Siebel Call Center']/web:document[@index='12' ]" +

"/web:button[@value=’NEXT’]”

// The index property has been removed

1. **Check for an object existence within a specific amount of time.**

web.exists(<path>, <timeout>)

This will avoid the script from becoming slow waiting for a long time/default time set.

1. **Right click on an element to show up a menu.**

web.element(<path>).showContextMenu();

1. **The name ,text and index property of an objects changes**

In Case, all properties of an object keeps changing as below:

@name=’pt1:r1:1:pc2:t1:0:it17’ or@ index=’10’

@name=‘pt1:r1:12:pc2:t1:3:it17’ or index=’13’ Etc.;

[In the above case only ‘it17’ remains the same throughout ]

We can make use of the below code:

//iterating to see if there is any textbox that has index between 0 and 100

for (int index =0;index<100;index++){

//Checking for the existence of a textBox

if(web.exists(<path>, 1)){

//Fetches the name of the property fetched

String tempEA\_ArrivalBaseName = web.textBox("/web:window[@index='0' or @title='OTM EVENTS EXTENSION'] /web:document[@index='0']/web:form[@id='pt1:ptf1' or @name='pt1:ptf1' or @index='0']/web:input\_text[@index='"+index+"']").getAttribute("name");

intEA\_ArrivalBaseIndex = index;

//Checking if the name of the property fetched ends with it17

if(tempEA\_ArrivalBaseName.indexOf("it117")>0){

strEA\_ArrivalBaseName=tempEA\_ArrivalBaseName;

break;

}

}

}

// intEA\_ArrivalBaseIndex gives the index and strEA\_ArrivalBaseName gives the name .

1. **Navigate through the hierarchical flow of elemental nodes and click on the appropriate element of the *tree* view mentioned in the Node Elemental Path .**

     //strNodeElementPath will contain the hierarchical flow of the base elemental node

String strNodeElementPath=”Fleet Management;Drivers;AssignDrivers”;

//Splits the hierarchical flow with delimiter as “;” and puts it into an array

String[] strClickMenuItems = strNodeElementPath.split(";");

Robot rRob = new Robot();

// strFormatSpecifier will contain 1st character of element id

String strFormatSpecifier = "1";

// fnDocumentIndex will resolve the document index in which the tree view is currently located

int intDocumentIndex = fnDocumentIndex(intWindowIndex, "/web:div[@id = 'actionTree.1\_1.l']");

if (intDocumentIndex !=999){

// Iterates until traverse is being made for no of levels which strNodeElementPath contains delimited by “;”

      for (int intMenuItemIter = 0 ; intMenuItemIter < strClickMenuItems.length ; intMenuItemIter++) {

            boolean isElementFound = false;

             //outer:

      try {

// Iterates through all an infinite loop until the desires element is found

  for (int intElementIter = 1 ; ; intElementIter++) {

//Fetch the text of the element based on the value of 'actionTree."+ strFormatSpecifier +"\_"+ intElementIter +".l' and update to the value of strElementText

String strElementText = web.element(       "/web:window[@index='"+intWindowIndex+"']/web:document[@index='"+intDocumentIndex+"']" +

"/web:div[@id='actionTree."+ strFormatSpecifier +"\_"+ intElementIter +".l']") .getAttribute("text");

//If strElementText is equal to strClickMenuItems then click on the link matched and update the isElementFound flag to //TRUE

if((strElementText.trim()).equalsIgnoreCase((strClickMenuItems[intMenuItemIter].trim()))) {

isElementFound = true;

"/web:window[@index='"+intWindowIndex+"']/web:document[@index='"+intDocumentIndex+"']" +  "/web:div[@id='actionTree."+ strFormatSpecifier +"\_"+ intElementIter +".l']").click();

think(2.00);

//Key press Right arrow when intMenuItemIter+1 values does not equals strClickMenuItems.length else BREAK the loop

if ((intMenuItemIter+1) != strClickMenuItems.length) {

                rRob.keyPress(KeyEvent.VK\_RIGHT);

                rRob.keyRelease(KeyEvent.VK\_RIGHT);

  } else break;

strFormatSpecifier += "\_"+ intElementIter;

//If isElementFound flag is TRUE then reset to FALSE and break the loop

             if (isElementFound) {

                    isElementFound = false;

                  break;

                        }

                 }

      }

// When ObjectNotFoundException exception is occurred then decrement intMenuItemIter by 1 and reset the isElementFound to FALSE

} catch(ObjectNotFoundException e) {

intMenuItemIter = strClickMenuItems.length - 1;

isElementFound = false;

}

// If intMenuItemIter+1 is equal to strClickMenuItems.length and isElementFound is TRUE post information into console about object is found

if (((intMenuItemIter+1) == strClickMenuItems.length) && (isElementFound))

            info ("Element: '"+ (strClickMenuItems[intMenuItemIter].trim()) +"' found in path: '"+ strNodeElementPath +"'");

// If intMenuItemIter+1 is equal to strClickMenuItems.length and isElementFound is FALSE post Warning into console about object is NOT found

else if (((intMenuItemIter+1) == strClickMenuItems.length) && (!(isElementFound)))

     warn ("Element: '"+ (strClickMenuItems[intMenuItemIter].trim()) +"' not found in path: '"+ strNodeElementPath +"'");

     }

//Return DONE upon success

      } return "Done";

//Return ERROR upon any Exception

 } catch (Exception e) {

      e.printStackTrace();

  return "Error";

  }

}

**Working in Oracle Forms**

1. **‘OK” button not recorded.**

On pressing the OK button in Oracle forms after performing a query, it gets recorded with some hard coded data and no ‘OK’ button at all. E.g.:

forms.listOfValues("//forms:listOfValues").select("MAIN|SNIUSOperatingUni|POBOX8100|DUBLIN|OH||430162100||US");

Ans: After entering the value for search, use the below keyboard ‘enter’ clicking process.

Robot robo = new Robot();

robo.keyPress(KeyEvent.VK\_ENTER);

robo.keyRelease(KeyEvent.VK\_ENTER);

1. **Find from a list of values**

Find button is not recognized here. Instead the code can be read as:

forms.listOfValues(<path>).find(strLegalEntity);

1. **Dialog box not recognized-Click on OK**

In case a dialog box is not recognized but have to click on the highlighted button say OK button, can use the below code:

Robot robot = new Robot();

// Press the Key board key - ENTER

robot.keyPress(KeyEvent.VK\_ENTER);

//Release the Key board key - ENTER

robot.keyRelease(KeyEvent.VK\_ENTER);

1. **Tab through fileds.**

forms.textField(<path>)

.invokeSoftKey("NEXT\_FIELD");

1. **Read status bar message**

String strStatusMSg = forms.getStatusBarMessage();

**General**

**1. Select a multiple value in Select Box in OpenScript**

* Select First Value
* Send the CTRL Key
* Select The Second Value

Code Template :

web.selectBox (Path)

. selectOptionByText(FirstValue);

web.selectBox (Path). keyPress("<CTRL>");

web.selectBox (Path). selectOptionByText(SecondValue);

Note : The word CTRL should be given within ‘<’ & ‘>”

1. **Maximize the browser in OpenScript ?**

web.window("/web:window[@index='0']")

.maximize();

1. **Unable to split a string using ‘.’**

Java will not split a string using ‘ .’ . hence first replace it with ‘ ;’ or some other special characters and go for the split function.

String strName=”Schneider.nationals”;

String strName1=strName .replace(".", ";");

String []strarrName2=strname1.split(;)

1. **Error writing to csv file**

Error may result if we try to write a null value. Hence require to check for null values before writing to CSV files. E.g.:

If(!strName!=null){

//Write to CSV

}else{

//Throw error msg

}

1. **Retrieve system date**

Calendar cal = Calendar.getInstance();

SimpleDateFormat sdf = new SimpleDateFormat(strDateFormat);

String strdate=sdf.format(cal.getTime());

\*strDateFormat- it’s the format in which we need the date. E.g.: "hhmmss" , "MMddyyss" , "MMddyyhhmmss" ,

“dd-MMM-yyyy” etc

1. **Different ways to handle exceptions:**

Any of the below code can be inserted in the catch block:

1. To print error on console:

e. printStackTrace ();

1. To throw customized exceptions:

throw new Exception ("<statement>");

1. To pass the exception to the calling script so as to continue with the execution rather than handling it in the current function:

return "Error";

Try catch format:

try{

<Code>

}catch (Exception e) {

// Any of the above code can be inserted here

}

1. **Common variables**

Common variables are variables initialized in java class. This can be done by creating a java class file from the package explorer of the script. Class variables do not get initialized during the iterations unless reinitialized inside the initialize function of a script. Hence can be used to store values across different iteration or scenarios.

1. **Read numbers from an alphanumeric data.**

Pattern p = Pattern.compile("[0-9]");

Matcher m = p.matcher(<alphanumeric data>);

StringBuffer refCode = new StringBuffer();

while (m.find()) {

refCode.append(m.group() );

}

String strNumber=refCode.toString();

1. **To get the path where the script lies**

getScriptPackage().getScriptPath();

1. **To get the repository path**

getScriptPackage().getRepository();

**OpenScript Tips**

1. **Object not captured on a browser opened by OpenScript**

Restart OpenScript

File->Restart

1. **OpenScript becomes idle after executing few iterations throwing port not foound error**

Increase the port No:

View->OpenScript preferences…->General->Record playback port range

1. **Capture a table and its values with ease**

Go to tree view. Select a step. Then Add object test icon becomes active. Click this and the capture button  . Now spy the table. After selecting the object by clicking F10 , click on OK. This populates the table of OpenScript with all the data from application table.

1. **Line to line debug**

Adding a breakpoint. Right click on a line and select Toggle breakpoint

Pressing F6 allows you to move line by line.

Press F5 allows you to step into any function specified

Press F8 to resume

1. **Remove all breakpoints**

Got to ‘debug view’-> Run->Remove all break points

1. **Delete a script**

Never delete a script directly from its folder.

Go to tools->Manage script->Select the script->Delete

1. **To proceed to next scenario/exception in case of exceptions caused by pop-up**

Restarting the browser can be done in case of Exceptions where the application is being blocked from navigating any further .

browser.close();

browser.launch ();

1. **To leave comma as it is while writing to CSV file:**

CSV file separates out data to different columns using commas (,) .In case, comma is the output to be written in CSV ,use the below format:

"\""+<value>+"\""

1. **What happens when there is no data available after splitting a string value**

When a string is split and assigned to an array, if no data is available after a split with (;) or some other special characters, the array size get automatically trimmed to the last available value.

Eg:

String strValue=”;new;openscript;;;;;;”;

String [] arrSplitArray=strValue.split(;);

int intLength=arrSplitArray.length();

//intLength will return value 3

Common function which deals with CSV files

Date format and Date Add function

1. **How to convert from array to a vector and vise versa**

String[] arrData={“a”.”b”,”C”};

//Converting array to a list

List list =Arrays.asList(arrData);

//Convert list to a vector

Vector vector = new Vector(list);

//Add an extra value to the vector

vector.add(“newValue”);

//Creating a new array with the dimension of the vector's size

String[] criteria = new String[vector.size()];

//Converting vector to array and writing to the destination file

vector.toArray(criteria));

1. **To perfom date/time addition to time present**

/  \* To perfom date/time addition to time present

      \* @param field eg: CALENDAR.MINUTE

      \* @param interval eg: 15

      \* @param strOTMExtTime eg: 3/2/2011 08:00 AM

      \*/

public String DateAdd (@Arg("field") int field,  @Arg("interval") int interval,         @Arg("strTime ") String strTime, @Arg("strFormat ") String strFormat) throws Exception {

//creating instance for Date

            Date date = new Date();

            Date postDate = new Date();

            SimpleDateFormat sdf = new SimpleDateFormat(strFormat);

//Parsing strTime to the given date format

            date = sdf.parse(strTime);

//creating a calendar instance

            Calendar cal = Calendar.getInstance();

            //cal.set(date.getYear(), date.getMonth(), date.getDate(), date.getHours(), date.getMinutes());

            //or

            cal.setTime(date);

//Add the interval to the field specified(date/hr/min/sec)

            cal.add(field, interval);

            postDate = cal.getTime();

            System.out.println("fnOTMExtTimeAdd - > Before : " + strTime);

            System.out.println("fnOTMExtTimeAdd - > After : " + sdf.format(postDate));

            return sdf.format(postDate);

      }

//DateAdd (Calendar.MINUTE, 15, strOTMExtTime) will call the above function

1. **File handling functions**

* Check existence of a file:

boolean blnFileExists = false;

File objFile = new File(Path);

if (objFile.exists()) {

blnFileExists= true;

}else{

blnFileExists = false;

}

* Write to CSV

BufferedWriter out;

if(!blnFileExists){

//you declare which file to write to add the boolean true which will stop it from replacing the helloworld.txt with a new one.

out = new BufferedWriter(new FileWriter(Path, false ));

} else{

out = new BufferedWriter(new FileWriter(Path, true ));

}

out.write(<Value>);

out.write(",");

out.newLine();

out.close();//flushes and closes the stream

* Read from CSV

//Load CSV

Table table = utilities.loadCSV(Path);

//Reading column values

String array[] = table.getColumnNames();

System.*out*.println(array.length);

//get no:of columns

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

{

**if** (array[i].compareToIgnoreCase(ColumnName)==0){//comparing with the required column name

//getting the row of data

Row row = table.getRow(RowNumber);

//getting the column value in the particular row

String StrReturnValue = row.get(ColumnName);

System.*out*.println(" StrReturnValue " + StrReturnValue);

**return** StrReturnValue;

}

}

Read numbers from a string

String strMessage = forms.choiceBox(810, "//forms:choiceBox").getAlertMessage();

Pattern p = Pattern.compile("[0-9]");

Matcher m = p.matcher(strMessage);

StringBuffer refCode = new StringBuffer();

while (m.find()) {

refCode.append(m.group() );

}

System.out.println(refCode.toString());

Regular Expression code snippet to Pick The First Number Occurance leaving other digits

**import** java.util.regex.Matcher;

**import** java.util.regex.Pattern;

**public** **class** RegExp {

**public** **static** **void** main(String args[]){

            Pattern p = Pattern.*compile*("[\\d+](file:///\\d+)");

            Matcher m = p.matcher("187401 ADDISON IL 60101-3117");

**if** (m.find())

            {     System.*out*.println(m.group(0)); }

      }

}