**Creating User Login Module:**

Let’s develop the functionality of user login. We develop it as a module so when we want to login as a user we can directly call the special keyword “userlogin” instead of mentioning all the steps.

Let’s look at the master keywords list for a user login. The following is the functionality on a login popup.

|  |  |  |  |
| --- | --- | --- | --- |
| **Keywords** | **Description** | **actionclass** | **objectrepository** |
| Useropenhomepage | user comes to homepage | userlogin | userloginOR |
| Userclickloginregister | user clicks on login/register | userlogin | userloginOR |
| Userenteremailaddress | user enters email address | userlogin | userloginOR |
| Userenterpassword | user enters password | userlogin | userloginOR |
| Userclicklogin | user clicks on login button | userlogin | userloginOR |
| Userclciksignuphere | user clicks on sign up here link | userlogin | userloginOR |
| Registerfullname | Enter full name in the register form | userlogin | userloginOR |
| Registermobileno | Enter mobileno in the register form | userlogin | userloginOR |
| Registeremailaddress | Enter email address in the register form | userlogin | userloginOR |
| Registerpassword | Enter password in the register form | userlogin | userloginOR |
| Registerconfirmpassword | Enter password in the confirm field of register form | userlogin | userloginOR |
| Userclicksignup | user clicks on sign up button | userlogin | userloginOR |
| Registersuccess | user gets a successful message on javascript alert | userlogin | userloginOR |
| Userclickloginhere | user clicks on login here link on sign up form | userlogin | userloginOR |
| Userclickforgotpassword | user clicks on forgot password link | userlogin | userloginOR |
| Userforgotemailaddress | user enters email address on forgot password form | userlogin | userloginOR |
| Userforgotmobileno | user enters mobile number on forgot password form | userlogin | userloginOR |
| Userforgotpassword | user enters new password on forgot password form | userlogin | userloginOR |
| Userforgotconfirmpassword | user enters password again on forgot password form | userlogin | userloginOR |
| Userresetmypassword | user clicks on button forgot my password | userlogin | userloginOR |
| userbacktologin | user clicks on back to login link on forgot password form | userlogin | userloginOR |
| userclosebrowser | user close browser | userlogin | userloginOR |

The functionality we need for a user login is highlighted in yellow color above. We have already developed the functionality for all the above keywords.

Since we would like to develop a module for login, lets name the group of keywords in yellow color as “userlogin” and make it as a separate sheet.

**userlogin group:**

|  |  |  |  |
| --- | --- | --- | --- |
| **keywords** | **description** | **actionclass** | **objectrepository** |
| userclickloginregister | user clicks on login/register | userlogin | userloginOR |
| userenteremailaddress | user enters email address | userlogin | userloginOR |
| userenterpassword | user enters password | userlogin | userloginOR |
| userclicklogin | user clicks on login button | userlogin | userloginOR |

Let’s add this name to the SPECIAL\_KEYWORDS in config.properties file.

**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,registeruser,**userlogin**

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

ADMIN\_UNAME = admin

ADMIN\_PWD = Test@12345

USER\_UNAME = subbu123

USER\_PWD = subbu123

Now Let’s develop the test case

**Test Case:**

|  |  |  |  |
| --- | --- | --- | --- |
| **testcase** | **teststep** | **keywords** | **description** |
| TC15 | 1 | useropenhomepage | user comes to homepage |
| TC15 | 2 | userlogin | User logs in by using the module keyword. Clicks on login  button, enter user name, password and clicks on login button. |
| TC15 | 3 | userclicksonname | User clicks on name drop down to signout |
| TC15 | 4 | userclicksonsignout | user clicks on sign out link |
| TC15 | 5 | userclosebrowser | Close the browser |

Out of the above, useropenhomepage and userclosebrowser are under userloginOR object repository and userlogin action class. We already defined these two. “userlogin” is also a module which we have already defined the functionality in the above two classes.

“userclicksonname: and “userclicksonsignout” are under userhomeOR and userhome action call. Lets define these two.

**userhomeOR.java under object repository:**

**package** objectrepository;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**public** **class** userhomeOR {

WebDriver driver;

**public** userhomeOR(WebDriver driver){

**this**.driver = driver;

}

**public** WebElement userclicksonname() {

WebElement userclicksonname = driver.findElements(By.*xpath*("//ul")).get(1);

**return** userclicksonname;

}

**public** WebElement userclicksonsignout() {

WebElement userclicksonsignout = driver.findElement(By.*xpath*("//a[contains(text(),'Sign Out')]"));

**return** userclicksonsignout;

}

}

**userhome.java actions class under actions package:**

**package** actions;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**public** **class** userhome {

WebDriver driver;

**public** userhome(WebDriver driver){

**this**.driver = driver;

}

**public** **void** userclicksonname(WebElement clicksonname) {

clicksonname.click();

}

**public** **void** userclicksonsignout(WebElement clicksonsignout) {

clicksonsignout.click();

}

}