

- Here is how the sheet looks like-

Testcase Name	Keywords	Object Name	Object Type	Value
Reset Login In Application				
	GOTOURL			url
	SETTEXT	username	name	Demo
	SETTEXT	password	name	testPassword
	CLICK	resetButton	name	
Login In Application				
	GOTOURL			url
	SETTEXT	username	name	Demo
	SETTEXT	password	name	testPassword
	CLICK	loginButton	name	

Excel Sheet For Keyword Driven Test

- According to the keywords written in Excel file, the framework will perform the operation on UI.
- For example, we need to click a button 'Login.' Correspondingly, our Excel will have a keyword 'Click.' Now the AUT can have hundreds of button on a page, to identify a Login button, in Excel we will input Object Name as loginButton & object type as a name (see highlighted the row in above image). The Object Type could be Xpath, name CSS or any other value
- For all of our UI web elements, we need to create an object repository where we will place their element locator (like Xpath, name, CSS path, class name etc.)

```

object.prop...  UIOperation....
url=http://demo
username=uid
password=password
title=barone
loginButton=btnLogin
resetButton=btnReset
  
```

This will be our object repository

NOTE: You may think why do we need to create an object repository. The answer helps in code maintenance. For example, we are using the button with name = btnlogin in 10 different test cases. In future, the developer decides to change the name from btnlogin to submit. You will have to make a change in all the 10 test cases. In the case of an object repository, you will make the change just once in the repository.