

Automation Examples



Test Automation Design

- Using Robot Framework examples
- Every good test framework will allow similar design
- Show how we extract duplication
- Making tests maintainable



Copyright 2010: Lisa Crispin



Example One: Simple Test

- Opens browser, brings up page, checks title, logs in with valid account info
- Everything hardcoded



Example One

*** Test Cases ***

Test Login

open browser http://127.0.0.1:7272 firefox
title should be Login Page

input text username_field demo
input text password_field mode
Click Button login_button

Title should be Welcome Page
Page should contain Login succeeded

*** Settings ***

Library SeleniumLibrary

Test Teardown close browser



Example Two: Extracting Duplication

- Good tools provide a way
- Robot Framework has “keywords”
- Other tools/languages have “includes”, “macros”, “classes”, “modules”
- Now we have two test cases
- Browser and server name in variables
- Defined keywords for actions that may be repeated



Example Two Test Cases

invalid account

navigate to the login page
type in username `invalid`
type in password `xxx`
click submit
verify the invalid account error message

valid account

navigate to the login page
type in username `demo`
type in password `mode`
click submit
verify the valid account message
click logout



Example Two Keywords (1)

*** Keywords ***

navigate to the login page

log hello, world!

open browser `${LOGIN PAGE}` `${BROWSER}`

title should be Login Page

`${title}` = Get Title

should start with `${title}` Login

type in username [Arguments] `${username}`

input text username_field `${username}`

type in password [Arguments] `${password}`

input text password_field `${password}`



Example Two Keywords (2)

click submit

Click Button login_button

click logout

Click Link logout

verify the invalid account error message

Title should be Error Page

Page should contain Invalid user name and/or password

verify the valid account message

Title should be Welcome Page

Page should contain Login succeeded



Example Two Settings/Variables

*** Settings ***

Library SeleniumLibrary

Test Teardown close browser

*** Variable ***

\${LOGIN PAGE} http://localhost:7272

\${BROWSER} firefox



Example Three: More Flexibility

- What if we could run the test with any username and password?
- We could pass in variable values as runtime arguments
- Let's do some refactoring



Example Three Test Cases

login account

navigate to the login page

type in username `${username}`

type in password `${password}`

click submit

verify the title

verify the message



Example Three Keywords (1)

navigate to the login page

open browser `${LOGIN PAGE}` `${BROWSER}`

title should be Login Page

`${title}` = Get Title

should start with `${title}` Login

type in username [Arguments] `${username}`

input text `username_field` `${username}`

type in password [Arguments] `${password}`

input text `password_field` `${password}`

click submit

Click Button `login_button`



Example Three Keywords (2)

click logout

Click Link logout

verify the title

Title should be `${title}`

verify the message

Page should contain `${message}`



Specifying the Variables at Runtime

```
./runDemo.py --variable username:demo  
--variable password:mode  
--variable message:'Login succeeded'  
--variable title:'Welcome Page' ExampleThree.txt
```



Example Four: Re-use

- Split keywords used by more than one test into a separate file
- In Robot Framework this is called a “resource”



navigate_to_login.txt – “resource”

*** Keywords ***

navigate to the login page

open browser `${LOGIN PAGE}` `${BROWSER}`

title should be Login Page

`${title}` = Get Title

should start with `${title}` Login



Example Four Settings

*** Settings ***

Documentation A test for valid login

Resource [navigate_to_login.txt](#)



Example Four Test Cases

login account

navigate to the login page

type in username \${username}

type in password \${password}

click submit

verify the title

verify the message



Example Four Keywords

type in username [Arguments] \${username}
input text username_field \${username}

type in password [Arguments] \${password}
input text password_field \${password}

click submit
Click Button login_button

click logout
Click Link logout

verify the title
Title should be \${title}

verify the message
Page should contain \${message}



Example Four Variables

*** Variable ***

`${LOGIN PAGE}` http://localhost:7272

`${BROWSER}` firefox

`${username}` demo

`${password}` mode

`${title}` Welcome Page

`${message}` Login succeeded



Other Re-Use Approaches: Example 5

- Module to click a button or link – pass in button or link name
- Module to input text – pass in text to enter
- All keywords in a resource file – a “library”

Main script has:

Resource example_resources.txt



Copyright 2010: Lisa Crispin



example_keywords.txt

(navigate to login page same as before)

click a link [Arguments] \${link_name}

Click Link \${link_name}

click a button [Arguments] \${button_name}

Click Button \${button_name}

type in text [Arguments] \${field_name} \${text_string}

input text \${field_name} \${text_string}

verify the title [Arguments] \${title}

Title should be \${title}

verify the message [Arguments] \${message}

Page should contain \${message}



Example Five test cases

login account

navigate to the login page

type in text username_field demo

type in text password_field mode

click a button login_button

verify the title Welcome Page

verify the message Login succeeded

click a link logout

login should fail

navigate to the login page

type in text username_field invaliduser

click a button login_button

verify the title Error Page

verify the message Invalid user name and/or password

