

Image Based Testing

Automate the app, not the code!



Email: Assi.Ivomir@gmail.com
Facebook: [Facebook.com/Ivomir.Assi](https://www.facebook.com/Ivomir.Assi)
LinkedIn: [LinkedIn.com/in/Ivomir-Assi](https://www.linkedin.com/in/Ivomir-Assi)

About me

- ~10 years of experience as a Quality Assurance Engineer
 - Cross-Platform Automation with: Web, Desktop, Mobile
 - Significant expertise in CI/CD and Jenkins
 - Pragmatic Academy Trainer
 - Udemy Trainer
-
- Hobbies: Sailing, Aviation, Ski, Travel



Ivomir Assi

Table of Contents

- Popular UI Tools
- Image Based Testing
 - Advantages
 - Limitations
- Intro to Popular Image Based Tools
 - Sikuli - demo
 - lackey
- Basic UI Framework
- Automation Demos
 - Calculator
 - Game
- Questions



Popular UI Tools

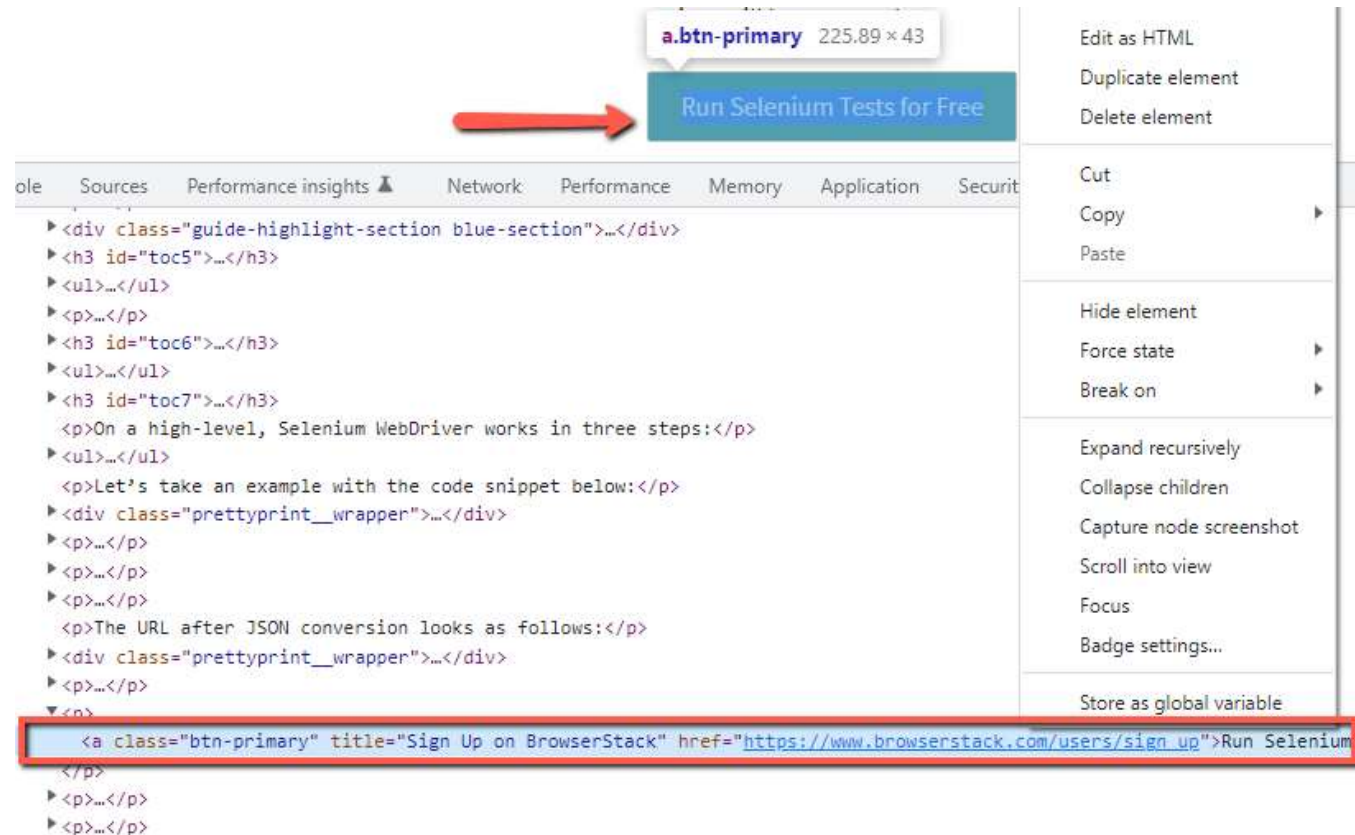
- Web App – Selenium
- Mobile App – Appium
- Windows Desktop App – Autoit



Web UI - DOM and Locators

driver.FindElement:

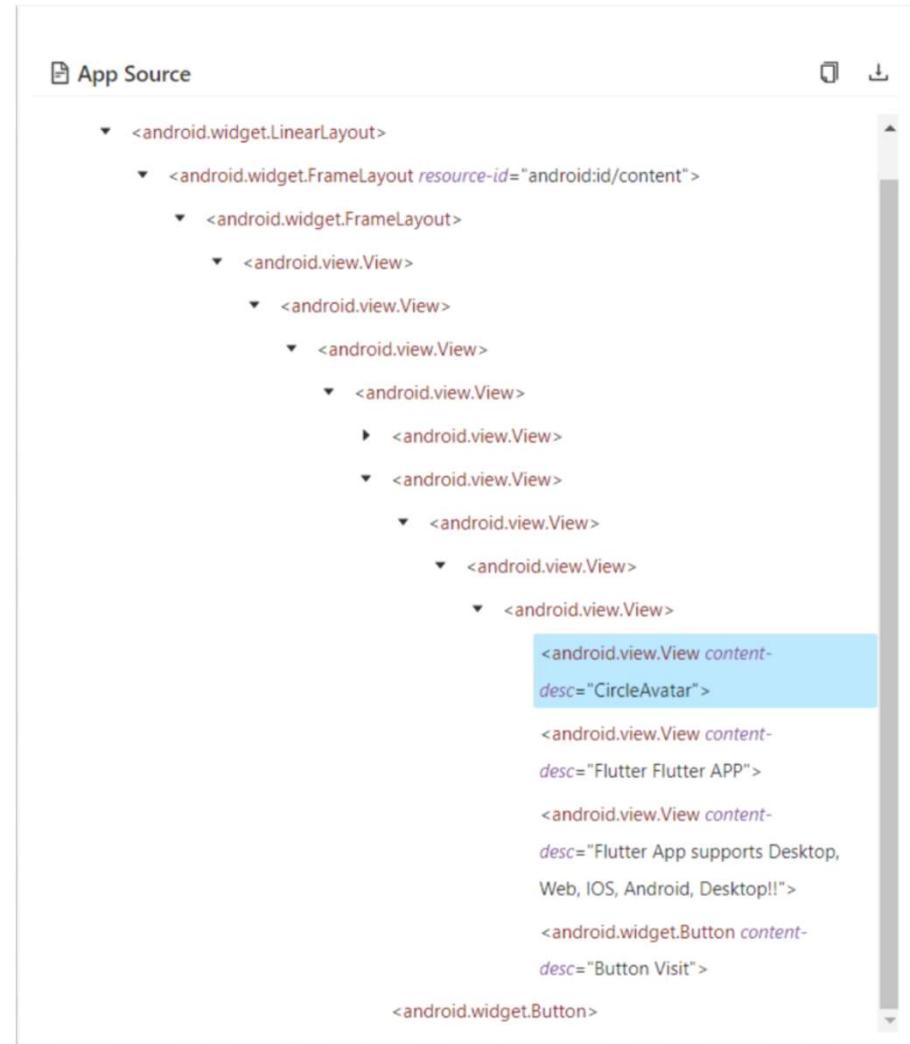
- By.Id("userName")
- By.ClassName("Example_Class")
- By.CssSelector("#userName")
- By.LinkText("Google")
- By.Name("DriverCheatSheet")
- By.PartialLinkText("Sikuli")
- By.TagName("a")
- By.XPath("//*[@id='panel']/div/h1")



Mobile (Android) – App Structure and Locators

driver.FindElement:

- ById("android:id/text1")
- ByClassName("android.widget.CheckBox")
- ByXPath("//*[@text='Views']")
- ByAccessibilityId("Views")



Desktop App – overview

Common things:

- Class
- Name
- ID

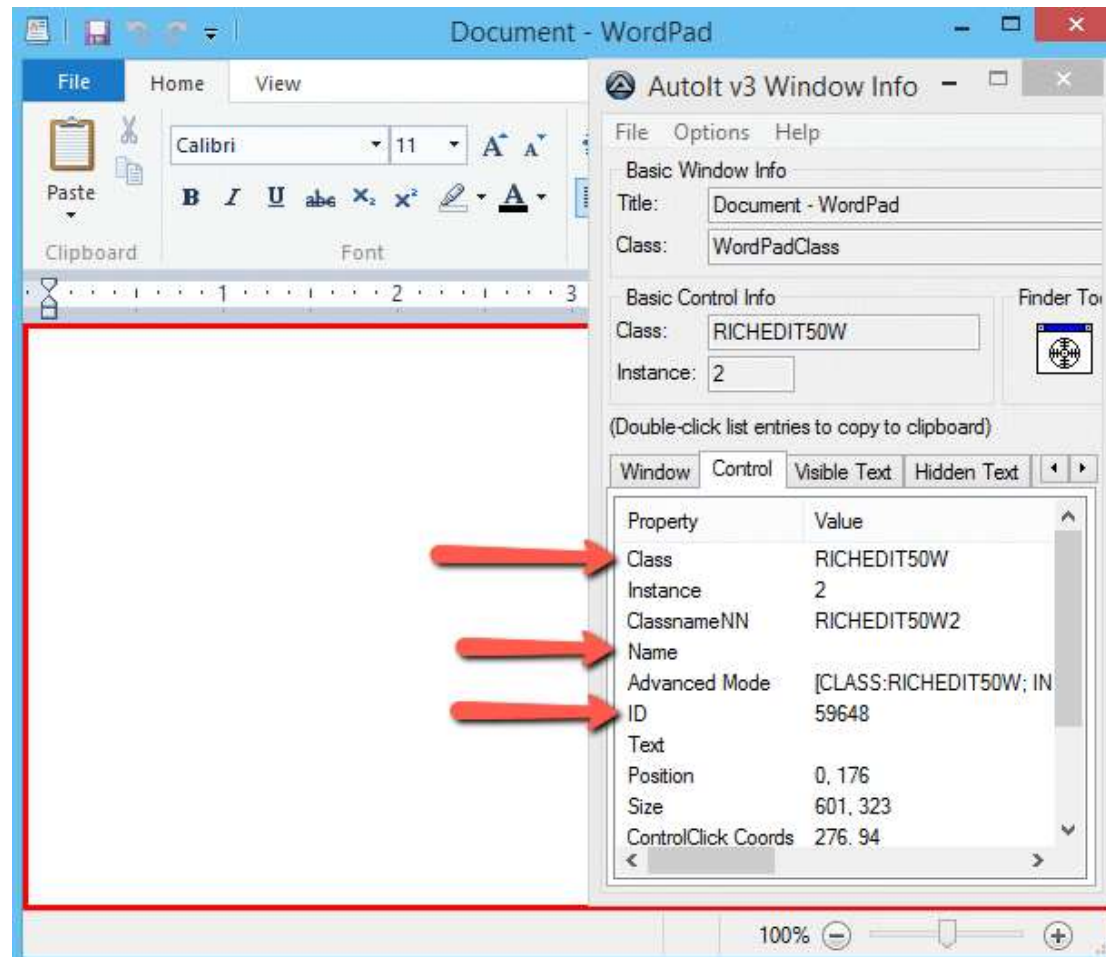
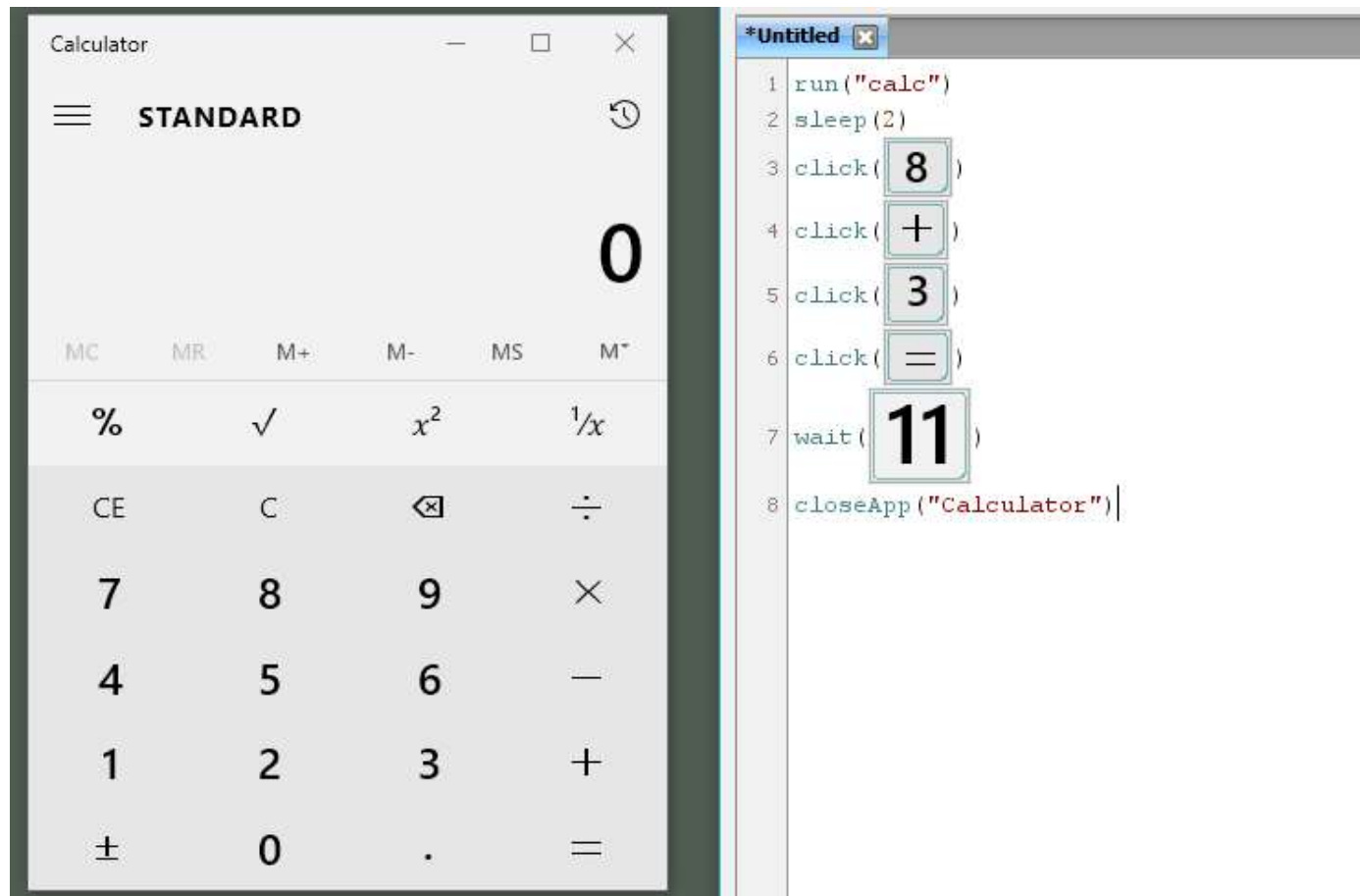


Image Based Testing

- Do you understand what this script does?



Advantages

- Cross-Platform tool
- Intuitive and natural approach
- Reproduce user/client actions



Limitations?

- Operates only in the visible screen
- Can be affected from: fronts, DPI
- Not working headless



Popular tools / libs

Free:

- Sikuli
- Lackey – Python(3.x) implementation of Sikuli

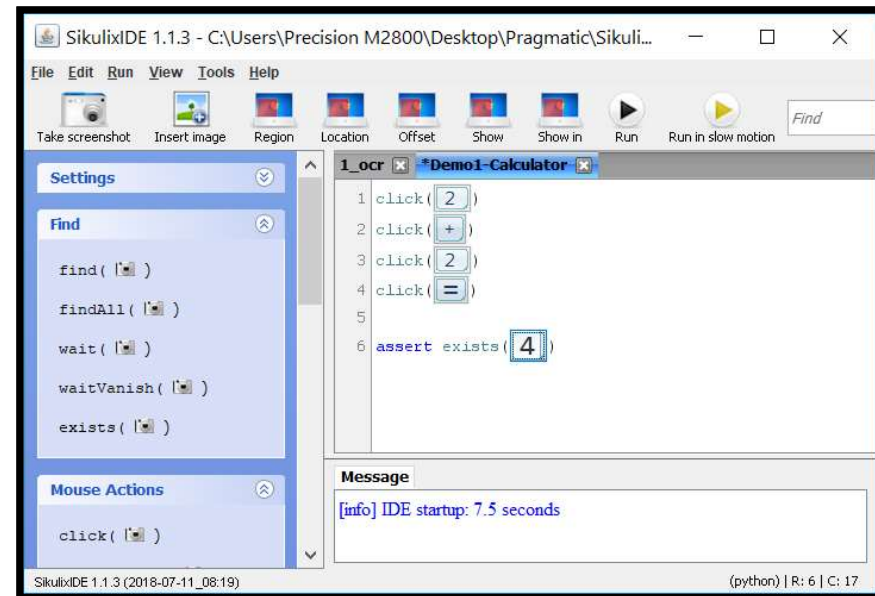
Paid:

- Eggplant
- Waldo
- Percy



Introducton to Sikuli

- **Screenshot driven** – it is a visual technology to search, recognize and automate GUI with images
- **Open-source visual scripting tool**
- Can **automate anything** which is displayed on the screen
- **Scripts are written in Jython** – which simply 'means' Python 2.7



Basic Sikuli Actions

- **Find** – find, wait, exists
- **Mouse Actions** – click, dragDrop, hover
- **Keyboard Actions** – type, paste
- **Others** – highlight, switchApp



Image Pattern Settings

- File
- Matching Preview – Similarity
- Target offset



Sikuli script directory

- **Sikuli script directory (.sikuli) contains:**
 - **HTML file (.html) – generated after save in Sikuli IDE**
 - **Python source file (.py)**
 - **Image files (.png)**



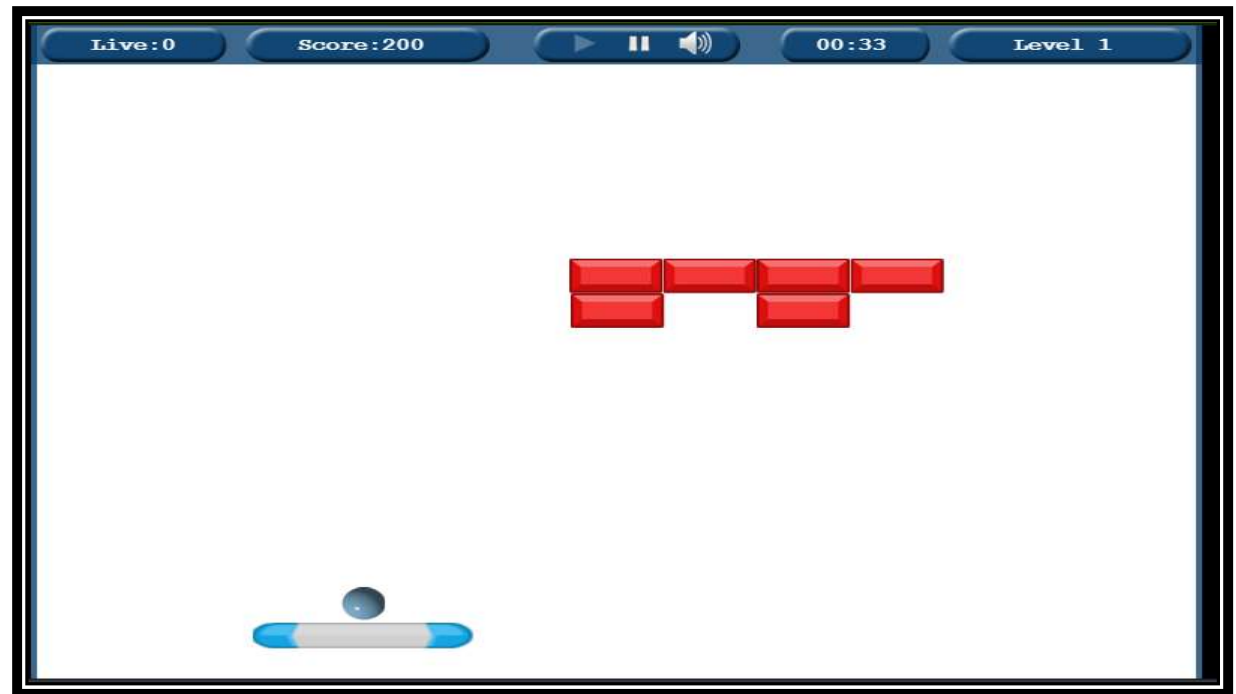
Sikuli is Cross-Platform Client-Agnostic

- Works on any GUI which can be displayed on Windows/Mac/Linux
 - Virtual machines
 - Remote Desktop
 - Simulators/Emulators
 - Desktop applications
 - Web
 - System menus
 - Flash
 - Video



Sikuli – Automate a game

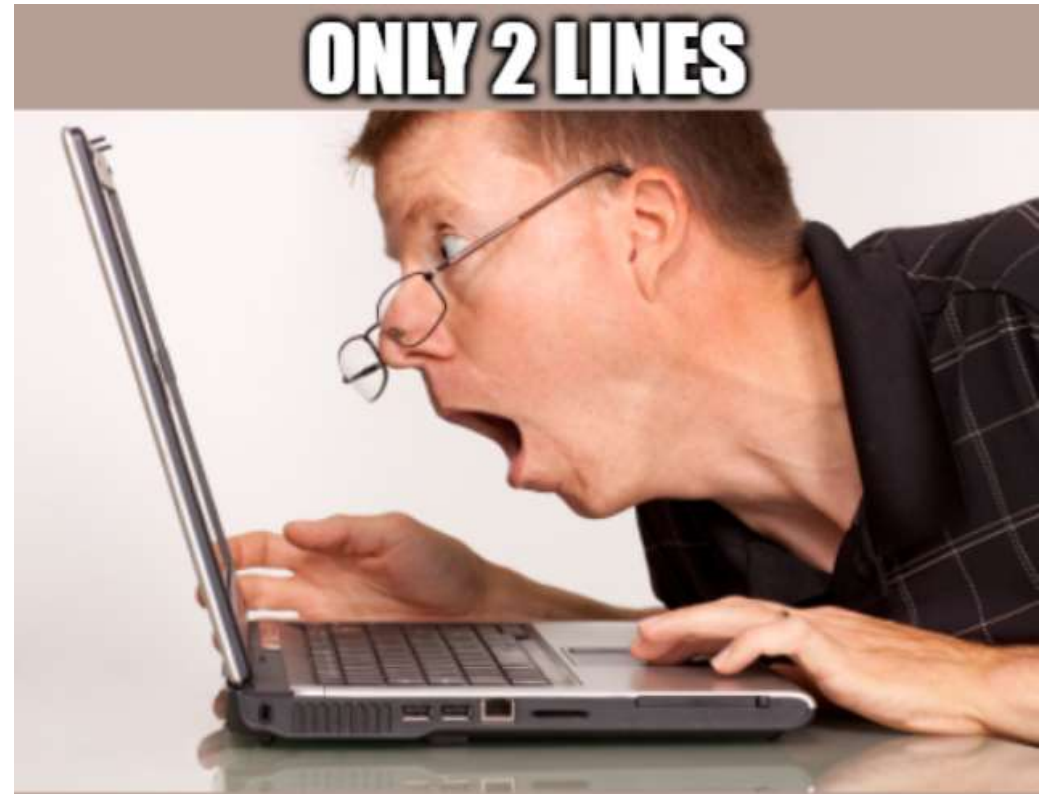
How many lines of code do we need to
Automate Bricks Game?



Sikuli Automate Game (2)

```
1 while (True) :  
2     hover (  )
```

ONLY 2 LINES



Python usage

Close 10 windows

```
1 click(✕)
2 click(✕)
3 click(✕)
4 click(✕)
5 click(✕)
6 click(✕)
7 click(✕)
8 click(✕)
9 click(✕)
10 click(✕)|
```

Close all Windows – smart way

```
1 button_Close = ✕
2 while exists(button_Close):
3     click(button_Close|)
```

Sikuli Demo



Basic Framework

Basic 3 component structure

- Test Set - tests
- UI Map – ui elements
- Library – useful actions



_lib.sikuli



_uimap.sikuli



smoke_tests.sikuli

Test Set

```
3
4 import unittest
5 from _lib import *
6
7 class SmokeTests(unittest.TestCase):
8     def setUp(self):
9         pass
10
11     def tearDown(self):
12         pass
13
14     def test_001_StartCalc(self):
15         Calc.Start()
16
17     def test_002_(self):
18         click(CalcUI.button_4)
19         click(CalcUI.button_plus)
20         click(CalcUI.button_6)
21         click(CalcUI.button_eq)
22         assert(exists(CalcUI.result_img))
23
24 if __name__ == '__main__':
25     suite = unittest.TestLoader().loadTestsFromTestCase(SmokeTests)
26
27     outfile = open("report.html", "w")
28     runner = HTMLTestRunner.HTMLTestRunner(stream=outfile, title='Report')
29     runner.run(suite)
30     outfile.close()
```

UI Map

```
#####  
# UI map for XYZ  
#####  
from sikuli import *  
#####
```

```
class CalcUI:
```

```
    button_4 = 
```

```
    button_6 = 
```

```
    button_plus = 
```

```
    button_eq = 
```

```
    result_img = 
```

Library

```
3
4 from sikuli import *
5 import HTMLTestRunner
6 from _uimap import *
7
8 class Calculator(object):
9     @classmethod
10     def Start(self):
11         type("r", KeyModifier.WIN)
12         type("calc")
13         type(Key.ENTER)
14
```

The Power of Sikuli

Cross Platform – Client Agnostic Automation

- Windows Calculator
- Android Calculator
- Web Calculator
- Virtual Machine – using RDP



Thank you

- Email: Assi.Ivomir@gmail.com
 - Facebook: Facebook.com/Ivomir.Assi
 - LinkedIn: LinkedIn.com/in/Ivomir-Assi
 - GitHub: AssiNET
 - [Udemy Course Sikuli](#)
- Automate Anything: Using Sikuli and Python

Free coupon code: SOFTUNI_OCT_22

