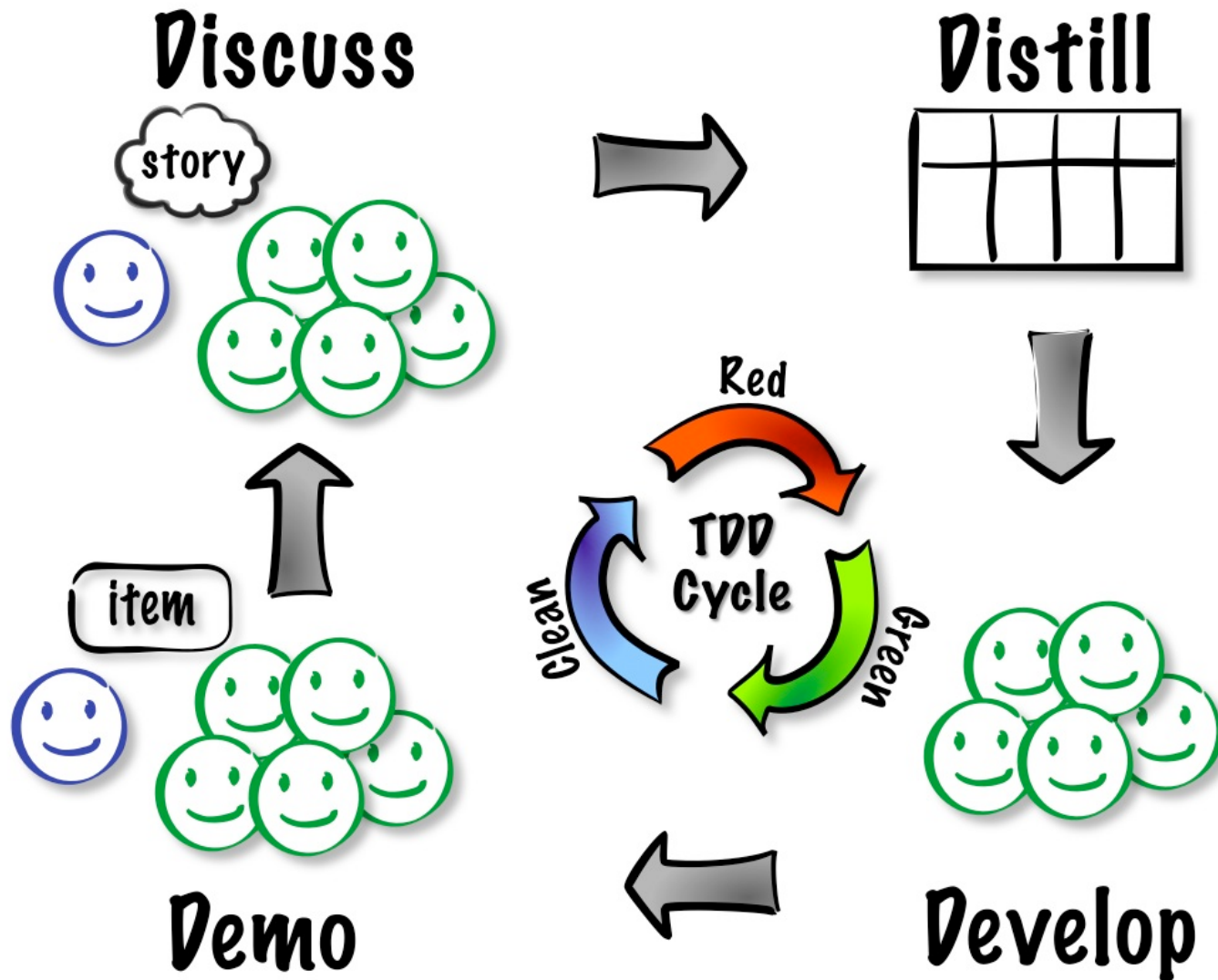


# Robot Framework

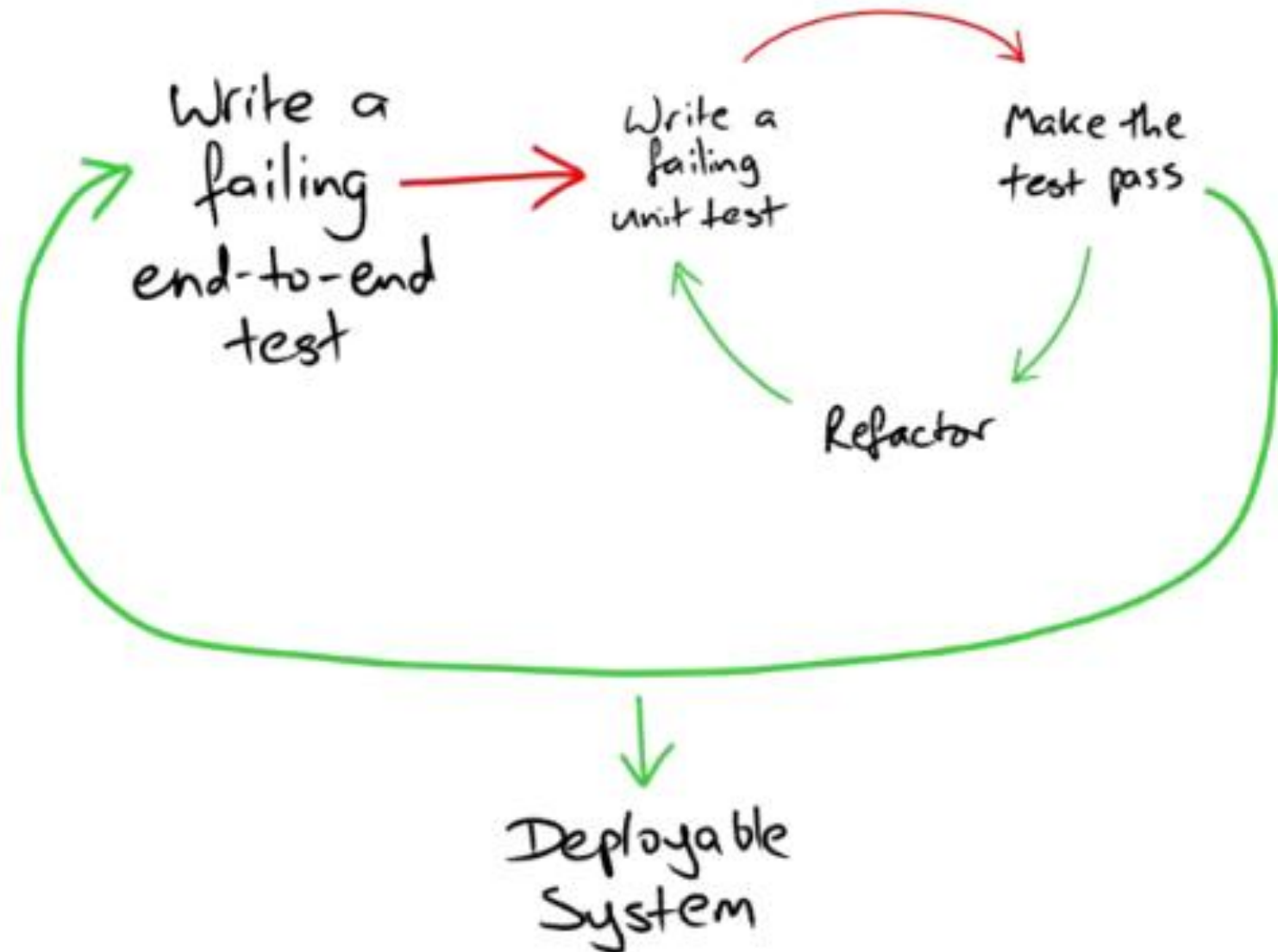
# Agenda

- Introduction to Robot Framework
- Installation
- Robot Framework - Standard Library
- Robot Framework - External Library
- Selenium2Library
- Simple Command
- Workshop
- Reviews Code

# Development Cycle



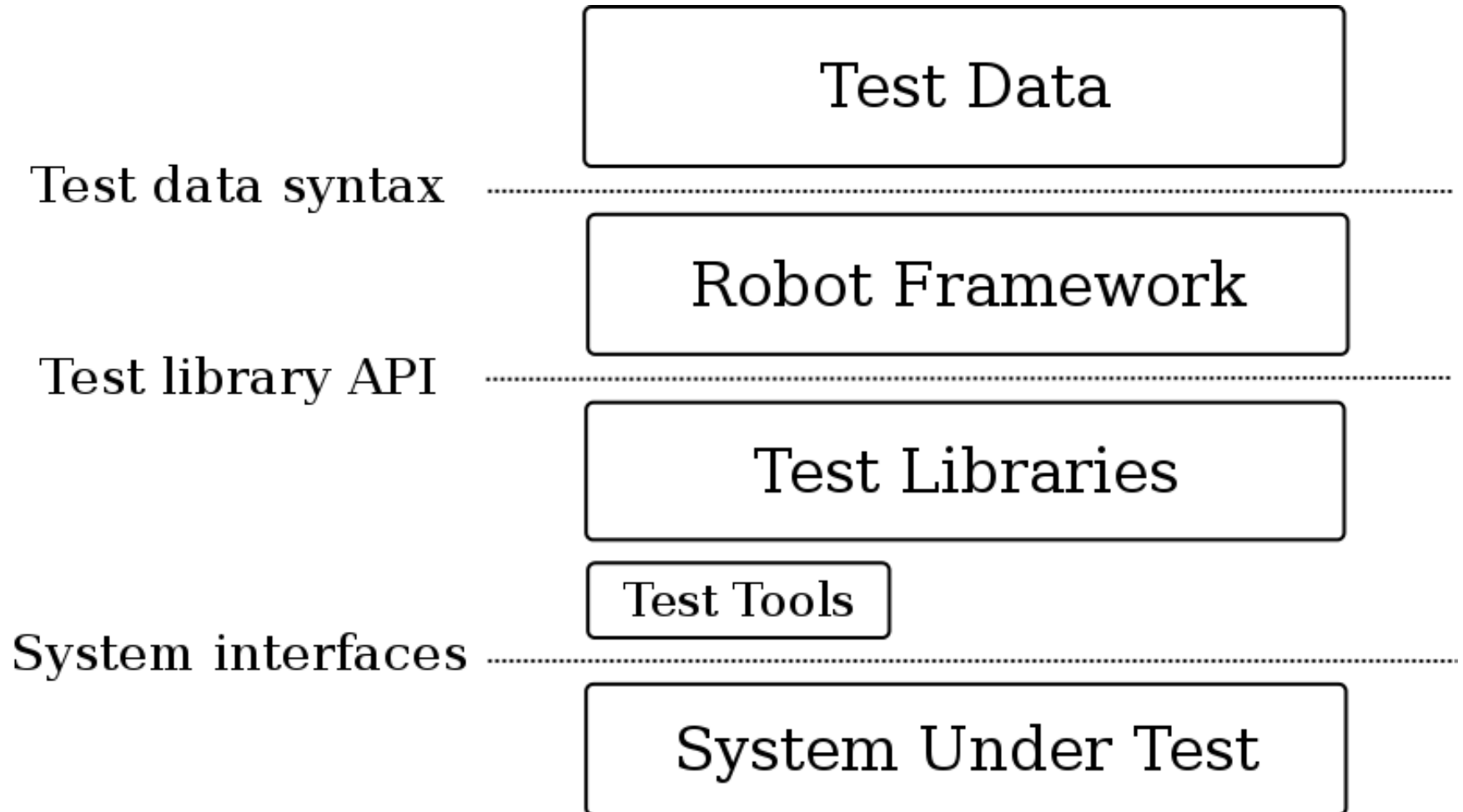
# Cycle



# Robot Framework

- Test automation framework
- <https://code.google.com/p/robotframework/>
- Keyword-Driven Development
- Based on Python
- Open source
- Sponsor by Nokia Siemens Networks

# Modular



# Installation

- Python 2.7/2.8
- pip
- Robot Framework
- Selenium2Library for Robot Framework

# Install python (1)

- Download from <https://www.python.org/>
- Version 2.7/2.8
- Run -> **python-2.7.12.msi**
- \$python



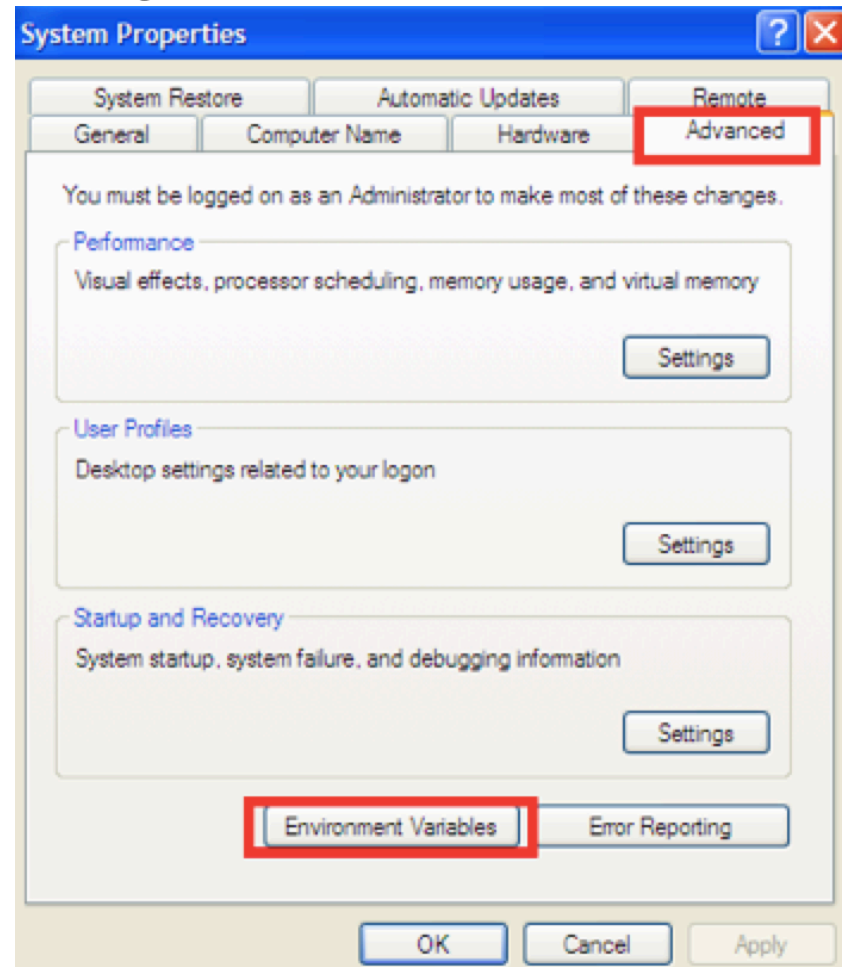
# Install python (2)

- **For Windows OS!**

- set PYTHON\_HOME=c:\python27
- set PATH=.;c:\Python27;c:\Python27\Scripts;  
%PATH%

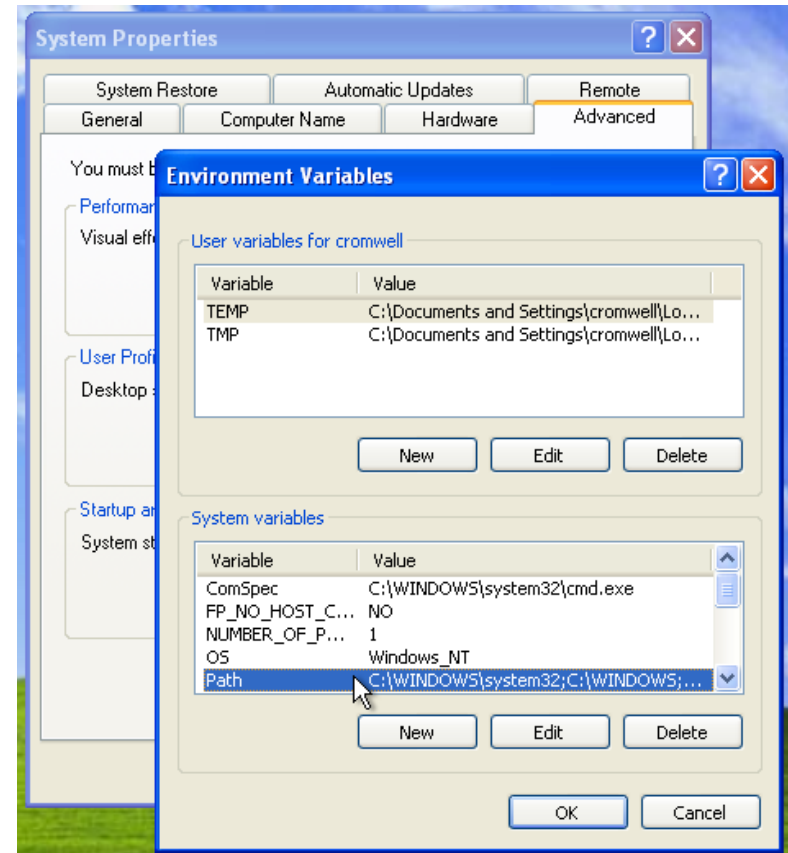
# Install python (3)

- Go to My Computer -> Right Click -> Properties
  - Choose tab Advanced



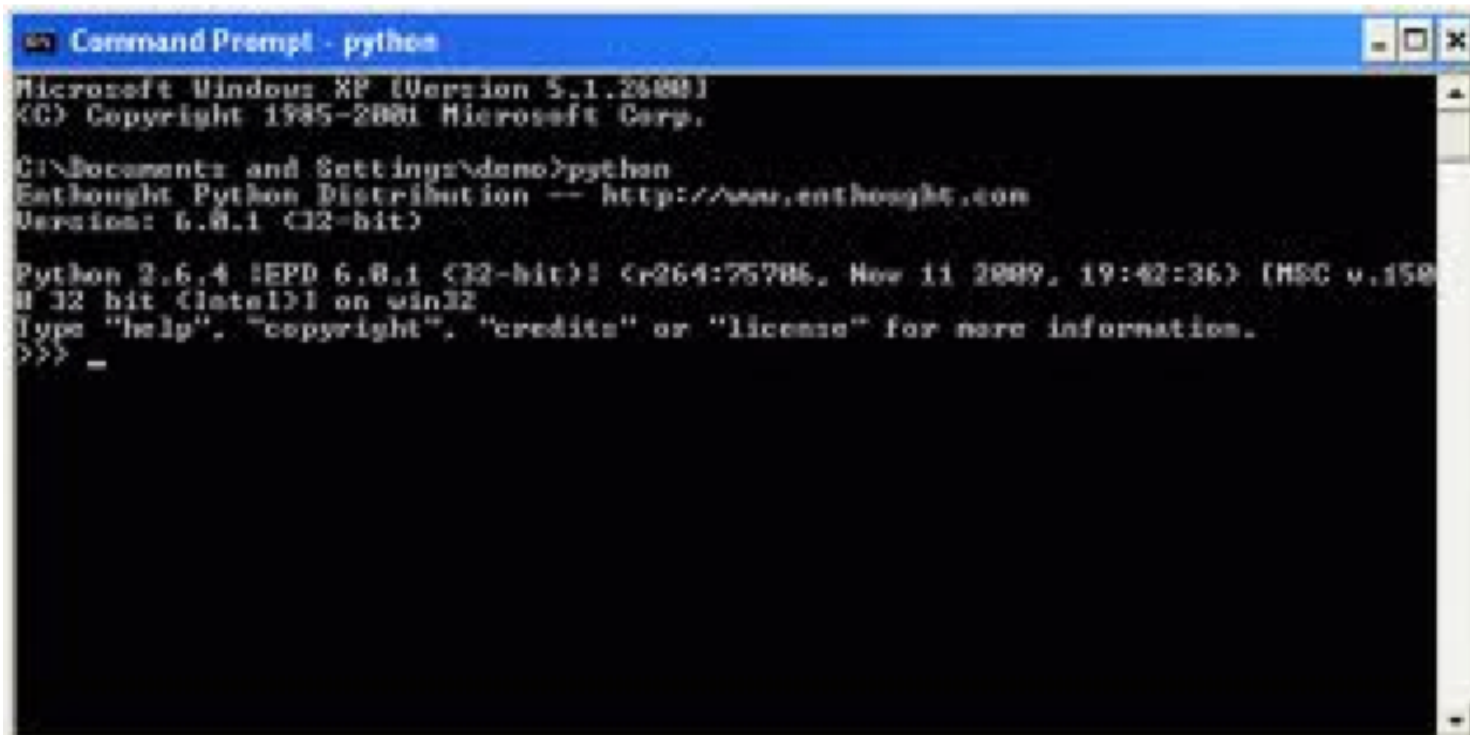
# Install python (4)

- Add Environment Variable
  - PATH = .;c:\Python27;c:\Python27\Scripts;  
%PATH%



# Install python (5)

- Install Completed



```
Command Prompt - python
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\dano>python
Ethink Python Distribution -- http://www.ethink.com
Version: 6.8.1 (32-bit)

Python 2.6.4 [AMD64 (32-bit)] (r264:75786, Nov 11 2009, 19:42:36) [MSC v.150
8 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

# Install pip (1)

- PIP = Package Installation for Python
- <http://www.pip-installer.org/en/latest/installing.html>
- Download file get-pip.py
- `$python get-pip.py`

# Install pip (2)

- \$pip

## Usage:

pip <command> [options]

## Commands:

install	Install packages.
uninstall	Uninstall packages.
freeze	Output installed packages in requirements format.
list	List installed packages.
show	Show information about installed packages.
search	Search PyPI for packages.
wheel	Build wheels from your requirements.
zip	Zip individual packages.
unzip	Unzip individual packages.
bundle	Create pybundles.
help	Show help for commands.

## General Options:

-h, --help	Show help.
-v, --verbose	Give more output. Option is additive, and can be used up to 3 times.
-V, --version	Show version and exit.
-q, --quiet	Give less output.
--log-file <path>	Path to a verbose non-appending log, that only logs failures. This log is active by default at /Users/spock/Library/Logs/pip.log.
--log <path>	Path to a verbose appending log. This log is inactive by default.
--proxy <proxy>	Specify a proxy in the form [user:passwd@]proxy.server:port.
--timeout <sec>	Set the socket timeout (default 15 seconds).
--exists-action <action>	Default action when a path already exists: (s)witch, (i)gnore, (w)ipe, (b)ackup.
--cert <path>	Path to alternate CA bundle.

# Install Robot Framework

- `$pip install robotframework`
- `$pybot`

```
[ ERROR ] Expected at least 1 argument, got 0.
```

```
Try --help for usage information.
```

# Install Selenium

```
$pip install robotframework-  
selenium2library
```

<https://github.com/robotframework/Selenium2Library>  
<http://robotframework.org/Selenium2Library/Selenium2Library.html>



# Robot Framework - Test Report

The screenshot shows a web browser window displaying a test report. The browser's address bar shows the file path: `file:///C:/RobotFramework/Extensions/AutoItLibrary/tests/results/report.html`. The report title is "AutotLibrary Windows Calculator Test Report". A "LOG" button is visible in the top right corner. The "Summary Information" section provides details about the test status, documentation, and timing. The "Test Statistics" section contains three tables: "Total Statistics", "Statistics by Tag", and "Statistics by Suite". Each table shows the number of tests, passes, failures, and elapsed time, along with a visual progress bar. The "Test Details" section at the bottom has buttons for "Totals", "Tags", "Suites", and "Search".

**AutotLibrary Windows Calculator Test Report**

Generated: 20160902 17:56:42 GMT +08:00  
2 days 7 hours ago

**Summary Information**

Status: All critical tests passed  
Documentation: Post installation self-test for AutotLibrary. Also serves as an example of some GUI automation techniques that can be used when automation Windows GUI tests.  
Start Time: 20160902 17:56:13.636  
End Time: 20160902 17:56:42.845  
Elapsed Time: 00:00:29.209  
Log File: [log.html](#)

**Test Statistics**

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
<a href="#">Critical Tests</a>	6	6	0	00:00:18	<div><div></div></div>
<a href="#">All Tests</a>	7	6	1	00:00:23	<div><div></div></div>

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
<a href="#">ExpectedFAIL</a> (non-critical)	1	0	1	00:00:06	<div><div></div></div>

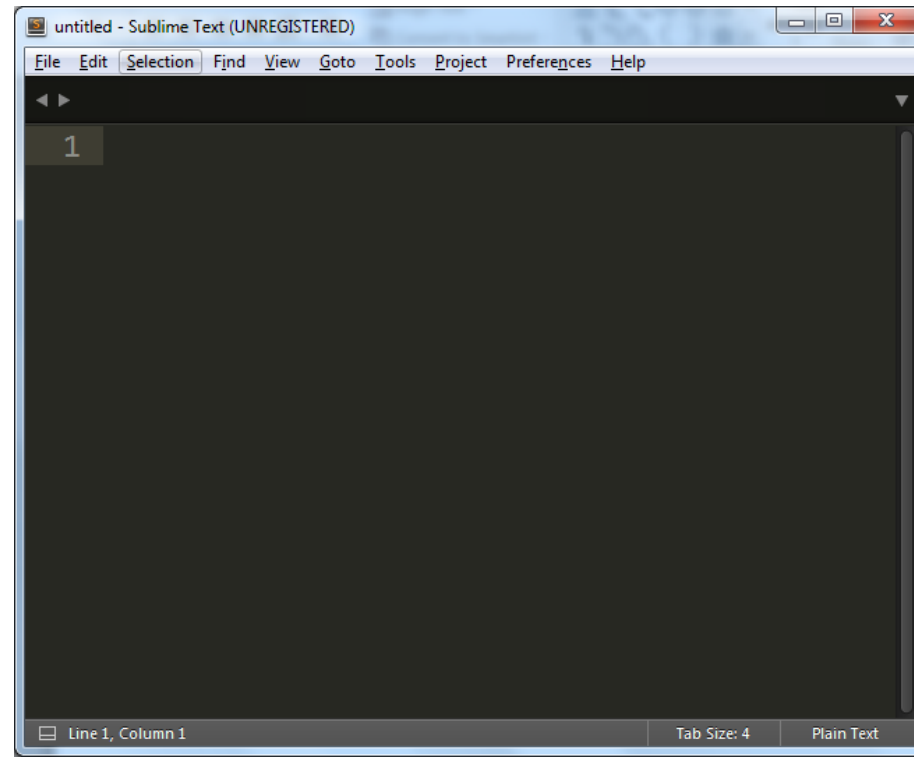
Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
<a href="#">AutotLibrary Windows Calculator</a>	7	6	1	00:00:29	<div><div></div></div>
<a href="#">AutotLibrary Windows Calculator . Calculator Test Cases</a>	7	6	1	00:00:29	<div><div></div></div>

**Test Details**

Totals Tags Suites Search

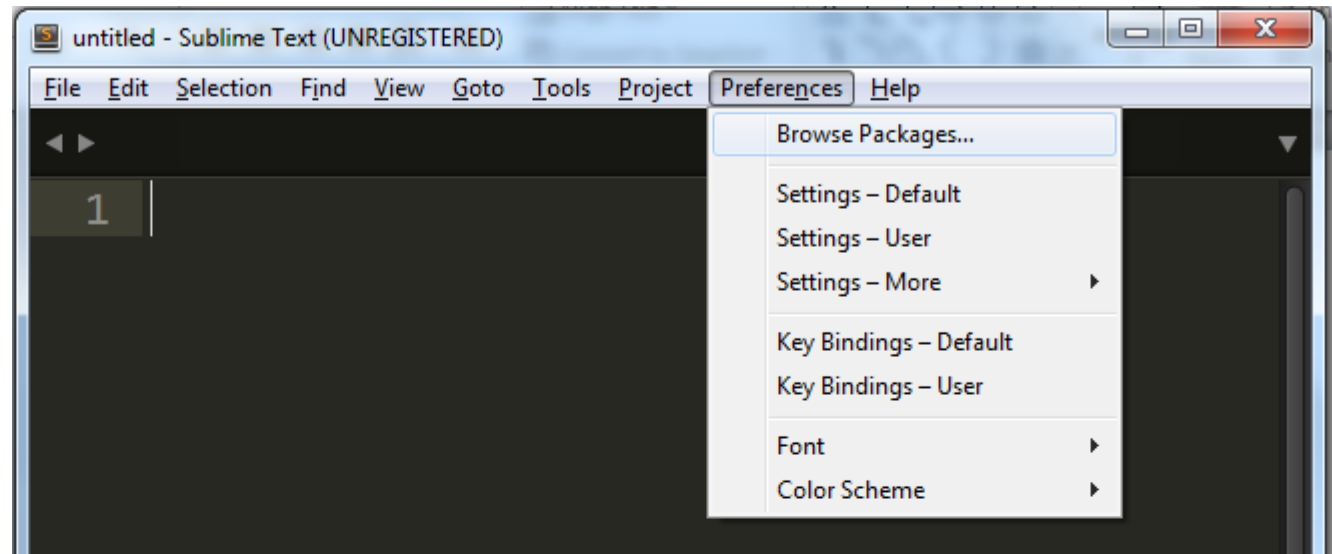
# Install Sublime 3 (Editor)

- Sublime 3
  - <http://www.sublimetext.com/>
  - Sublime Text Build 3114 x64 Setup.exe
  - Open Sublime3

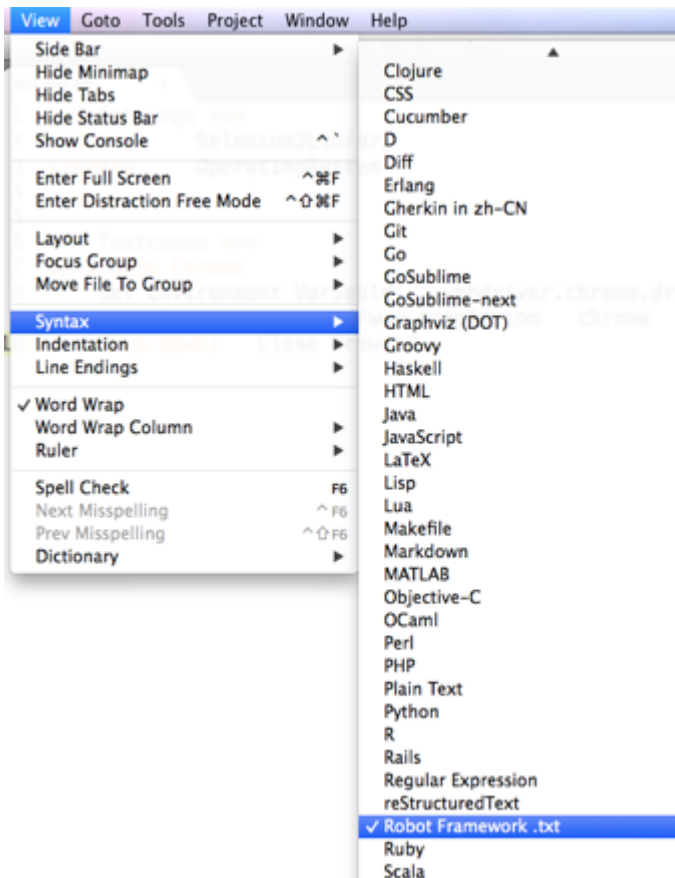


# Install Robot Framework - Sublime Extension

- Goto Preferences -> Browse Packages ...
- Double Click Folder “User”
- Copy “Robot Framework Assistant” into this (User) Folder



# Sublime - Robot Framework



```
basic_grade.txt
1  *** Settings ***
2  Library                Selenium2Library
3  #Test Teardown         Close Browser
4
5  *** Variables ***
6  ${HOME_URL}            http://demo-jenkins-web.herokuapp.com
7
8  *** Test Cases ***
9  Convert score 50 to grade D
10     Open browser to home page
11     Fill in score      50
12     I should see result  D
13     [Teardown]         Close Browser
14
15  Convert score 80 to grade D
16     Open browser to home page
17     Fill in score      80
18     I should see result  A
19     [Teardown]         Close Browser
20
```

# Test Structure

test\_000.txt

\*

1 \*\*\* Settings \*\*\*

2

3

4 \*\*\* Variables \*\*\*

5

6

7 \*\*\* Testcases \*\*\*

8

9

10 \*\*\* Keyword \*\*\*

# Standard Library

## Builtin

Provides a set of often needed generic keywords. Always automatically available without imports.

## Dialogs

Provides means for pausing the test execution and getting input from users.

## Collections

Provides a set of keywords for handling Python lists and dictionaries.

## OperatingSystem

Enables various operating system related tasks to be performed in the system where Robot Framework is running.

## Remote

Special library acting as a proxy between Robot Framework and test libraries elsewhere. Actual test libraries can be running on different machines and be implemented using any programming language supporting XML-RPC protocol.

## Screenshot

Provides keywords to capture screenshots of the desktop.

## Process

Library for running processes in the system. New in Robot Framework 2.8.

## String

Library for generating, modifying and verifying strings.

## Telnet

Makes it possible to connect to Telnet servers and execute commands on the opened connections.

## XML

Library for generating, modifying and verifying XML files.

## DateTime

Library for date and time conversions. New in Robot Framework 2.8.5.

# External Library

## Android library

Library for all your Android automation needs. It uses Calabash Android internally.

## AnywhereLibrary

Library for testing Single-Page Apps (SPA). Uses Selenium Webdriver and Appium internally.

## AppiumLibrary

Library for Android- and iOS-testing. It uses Appium internally.

## Archive library

Library for handling zip- and tar-archives.

## AutoItLibrary

Windows GUI testing library that uses AutoIt freeware tool as a driver.

## Database Library (Java)

Java-based library for database testing. Usable with Jython. Available also at [Maven central](#).

## Database Library (Python)

Python based library for database testing. Works with any Python interpreter, including Jython.

## Diff Library

Library to diff two files together.

## Django Library

Library for [Django](#), a Python web framework.

## robotframework-faker

Library for [Faker](#), a fake test data generator.

## FTP library

Library for testing and using FTP server with Robot Framework.

## HTTP library (livetest)

Library for HTTP level testing using livetest tool internally.

## HTTP library (Requests)

Library for HTTP level testing using Request internally.

## iOS library

Library for all your iOS automation needs. It uses Calabash iOS Server internally.

## ImageHorizonLibrary

Cross-platform, pure Python library for GUI automation based on image recognition.

## MongoDB library

Library for interacting with MongoDB using pymongo.

## MQTT library

Library for testing MQTT brokers and applications.

## Rammbock

Generic network protocol test library that offers easy way to specify network packets and inspect the results of sent and received packets.

## RemoteSwingLibrary

Library for testing and connecting to a java process and using SwingLibrary, especially Java Web Start applications.

## SeleniumLibrary

Web testing library that uses popular Selenium tool internally. Uses deprecated Selenium 1.0 and is also itself deprecated.

## Selenium2Library

Web testing library that uses Selenium 2. For most parts drop-in-replacement for old SeleniumLibrary.

## Selenium2Library for Java

Java port of the Selenium2Library.

## SSHLibrary

Enables executing commands on remote machines over an SSH connection. Also supports transferring files using SFTP.

## SudsLibrary

A library for functional testing of SOAP-based web services based on Suds, a dynamic SOAP 1.1 client.

## SwingLibrary

Library for testing Java applications with Swing GUI.

## watir-robot

Web testing library that uses Watir tool.

# Selenium Keyword

- Open Browser <url> <browser>
- Input Text <locator> <value>
- Click Button <locator>
- Click Link <locator>
- Close Browser



# Selenium Keyword

- Wait Until Page Contains <text>
- Page Should Contain <text>
- Title Should Be <text>

# More ...

**Add Cookie · Alert Should Be Present · Assign Id To Element · Capture Page Screenshot · Checkbox Should Be Selected ·**  
**Checkbox Should Not Be Selected · Choose Cancel On Next Confirmation · Choose File · Choose Ok On Next Confirmation · Click Button ·**  
**Click Element · Click Element At Coordinates · Click Image · Click Link · Close All Browsers · Close Browser · Close Window · Confirm Action ·**  
**Create Webdriver · Current Frame Contains · Current Frame Should Not Contain · Delete All Cookies · Delete Cookie · Double Click Element ·**  
**Drag And Drop · Drag And Drop By Offset · Element Should Be Disabled · Element Should Be Enabled · Element Should Be Visible ·**  
**Element Should Contain · Element Should Not Be Visible · Element Text Should Be · Execute Async Javascript · Execute Javascript · Focus ·**  
**Frame Should Contain · Get Alert Message · Get All Links · Get Cookie Value · Get Cookies · Get Element Attribute · Get Horizontal Position ·**  
**Get List Items · Get Location · Get Matching Xpath Count · Get Selected List Label · Get Selected List Labels · Get Selected List Value ·**  
**Get Selected List Values · Get Selenium Implicit Wait · Get Selenium Speed · Get Selenium Timeout · Get Source · Get Table Cell · Get Text ·**  
**Get Title · Get Value · Get Vertical Position · Get Window Identifiers · Get Window Names · Get Window Size · Get Window Titles · Go Back ·**  
**Go To · Input Password · Input Text · List Selection Should Be · List Should Have No Selections · Location Should Be ·**  
**Location Should Contain · Log Location · Log Source · Log Title · Maximize Browser Window · Mouse Down · Mouse Down On Image ·**  
**Mouse Down On Link · Mouse Out · Mouse Over · Mouse Up · Open Browser · Open Context Menu · Page Should Contain ·**  
**Page Should Contain Button · Page Should Contain Checkbox · Page Should Contain Element · Page Should Contain Image ·**  
**Page Should Contain Link · Page Should Contain List · Page Should Contain Radio Button · Page Should Contain Textfield ·**  
**Page Should Not Contain · Page Should Not Contain Button · Page Should Not Contain Checkbox · Page Should Not Contain Element ·**  
**Page Should Not Contain Image · Page Should Not Contain Link · Page Should Not Contain List · Page Should Not Contain Radio Button ·**  
**Page Should Not Contain Textfield · Press Key · Radio Button Should Be Set To · Radio Button Should Not Be Selected ·**  
**Register Keyword To Run On Failure · Reload Page · Select All From List · Select Checkbox · Select Frame · Select From List ·**  
**Select From List By Index · Select From List By Label · Select From List By Value · Select Radio Button · Select Window ·**  
**Set Browser Implicit Wait · Set Selenium Implicit Wait · Set Selenium Speed · Set Selenium Timeout · Set Window Size · Simulate ·**  
**Submit Form · Switch Browser · Table Cell Should Contain · Table Column Should Contain · Table Footer Should Contain ·**  
**Table Header Should Contain · Table Row Should Contain · Table Should Contain · Textarea Should Contain · Textarea Value Should Be ·**  
**Textfield Should Contain · Textfield Value Should Be · Title Should Be · Unselect Checkbox · Unselect Frame · Unselect From List ·**  
**Unselect From List By Index · Unselect From List By Label · Unselect From List By Value · Wait For Condition · Wait Until Element Is Visible ·**  
**Wait Until Page Contains · Wait Until Page Contains Element · Xpath Should Match X Times**

<http://robotframework.org/Selenium2Library/Selenium2Library.html>

# Hello Robot


- Create file **hello.txt**

A screenshot of a code editor window titled 'hello.txt'. The editor contains seven lines of text with syntax highlighting. Line 1: '\*\*\* Settings \*\*\*' in green. Line 2: 'Library Selenium2Library' with 'Library' in orange. Line 3: empty. Line 4: empty. Line 5: '\*\*\* Testcases \*\*\*' in green. Line 6: 'Hello World' with 'Hello' in orange. Line 7: 'Open Browser http://www.google.com' with 'Open' in orange and the URL in blue. The line number '7' is highlighted in yellow on the left margin.

```
1  *** Settings ***
2  Library      Selenium2Library
3
4
5  *** Testcases ***
6  Hello World
7  Open Browser  http://www.google.com
```

- Run with
  - **\$pybot hello.txt**

# Keyword Driven Development (Be careful)

Open Browser  Robot Framework

↑ Only 1 Spacebars      ↑ Only 1 Spacebars

4 Spacebars

# Multi Arguments with Empty Value

4 Spacebars      4 Spacebars      4 Spacebars

Control Click ➡ Calculator ➡  $\${EMPTY}$  ➡ *Button29*

↑

Only 1 Spacebars

# Report

```
spocks-MacBook-Pro:demo spock$ pybot 02_hello.txt
```

```
=====
```

```
02 Hello
```

```
=====
```

```
Hello World
```

```
| PASS |
```

```
-----
```

```
02 Hello
```

```
| PASS |
```

```
1 critical test, 1 passed, 0 failed
```

```
1 test total, 1 passed, 0 failed
```

```
=====
```

```
Output: /Users/spock/somkiat/training/KIMBERLY-CLARK/robotframework/demo/output.xml
```

```
Log: /Users/spock/somkiat/training/KIMBERLY-CLARK/robotframework/demo/log.html
```

```
Report: /Users/spock/somkiat/training/KIMBERLY-CLARK/robotframework/demo/report.html
```

# Report

02 Hello Test Report

file:///Users/spock/somkiat/training/KIMBERLY-CLARK/robotframework/demo/report.html

LOG

## 02 Hello Test Report

Generated  
20140520 23:15:49 GMT +07:00  
2 minutes 3 seconds ago

### Summary Information

Status:

All tests passed

Start Time:

20140520 23:15:42.790

End Time:

20140520 23:15:49.308

Elapsed Time:

00:00:06.518

Log File:

[log.html](#)

### Test Statistics

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	1	1	0	00:00:06	<div></div>
All Tests	1	1	0	00:00:06	<div></div>

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
No Tags					<div></div>

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
02 Hello	1	1	0	00:00:07	<div></div>

### Test Details

TotalsTagsSuitesSearch

Type:

☐ Critical Tests☐ All Tests

# Workshop

- Create test case with Robot Framework
  - Test [www.google.com](http://www.google.com)

```
work_01.txt
1  *** Settings ***
2  Library      Selenium2Library
3
4
5  *** Testcases ***
6  Search Data From Google.com
7      XXXXX
8      XXXXX
9      XXXXX
10     XXXXX
11     XXXXX
```



# Search with Google

```
work_01.txt *
1  *** Settings ***
2  Library          Selenium2Library
3
4
5  *** Testcases ***
6  Search Data From Google.com
7      Open Browser      http://google.com
8      Input Text        gbqfq      Somkiat
9      Click Button      gbqfb
10     Wait Until Page Contains  Somkiat
11     Close Browser
```

# Question & Answer

# References

- <http://robotframework.org/#test-libraries>
- <http://robotframework.org/robotframework/latest/libraries/BuiltIn.html>
- <http://robotframework.org/Selenium2Library/Selenium2Library.html>
- <https://www.autoitscript.com/site/>