
python_selenium

Release 1.1

shmakovpn@yandex.ru

Jan 29, 2021

CONTENTS:

1	code	3
1.1	makedoc.py	3
1.2	auto_makedoc.sh	4
1.3	run_coverage.py	5
1.4	python_selenium	6
2	sphinx	11
2.1	Sphinx code-block types	11
3	Indices and tables	21
	Python Module Index	23
	Index	25

Notes and examples about Python and Selenium.

python_selenium code documentation.

1.1 makedoc.py

This script builds documentation for the project.

Listing 1: source code

```
1  """
2  makedoc.py
3  ++++++
4
5  Generates the documentation of this project
6
7  Usage:
8
9  .. code-block:: bash
10
11     python makedoc.py
12
13     | Author: shmakovpn <shmakovpn@yandex.ru>
14     | Date: 2020-01-13
15     """
16  import os
17  # The path to the folder of this script
18  SCRIPT_DIR: str = os.path.dirname(os.path.abspath(__file__))
19
20
21  def run_sphinx() -> None:
22      """Executes *sphinx-build* command"""
23      docs_dir: str = os.path.join(SCRIPT_DIR, 'docs')
24      docs_source_dir: str = os.path.join(docs_dir, 'source')
25      build_dir: str = os.path.join(docs_dir, 'build')
26      html_dir: str = os.path.join(build_dir, 'html')
27      cmd: str = f'sphinx-build -b html "{docs_source_dir}" "{html_dir}"'
28      os.system(cmd)
29      print('__END__')
30
31
32  if __name__ == '__main__':
33      run_sphinx()
```

1.1.1 Autodoc documentation

makedoc.py

Generates the documentation of this project

Usage:

```
python makedoc.py
```

Author: shmakovpn <shmakovpn@yandex.ru>

Date: 2020-01-13

`makedoc.run_sphinx()` → None

Executes *sphinx-build* command

1.2 auto_makedoc.sh

This script automatically executes *makedoc.py* on changes in the documentation source folder.

Listing 2: source code

```
1  #!/bin/bash
2
3  # Runs `sphinx build` when files in ./doc/source was changed,
4  # uses inotifywait for watching files changes
5  # Author: shmakovpn <shmakovpn@yandex.ru>
6  # Date: 2020-01-07
7
8  # Requirements:
9  # Ubuntu: inotify-tools
10 # Centos 8: inotify-tools (from epel repository)
11
12 SCRIPT_DIR="$(dirname $0)"
13 DOCS_SOURCE_DIR="${SCRIPT_DIR}/docs/source"
14
15 # Checking that a VIRTUALENV is activated, exit otherwise
16 if ! test "${VIRTUAL_ENV}" ; then
17     echo "A virtualenv is not activated. \${VIRTUAL_ENV} is null"
18     exit 1
19 fi
20
21 # Checking that *inotifywait* is installed
22 if ! which inotifywait > /dev/null 2>&1 ; then
23     echo "*inotifywait* is not installed. Install package *inotify-tools*."
24     exit 1
25 fi
26
27 # `inotifywait -r -m -e modify -e move -e create -e delete watching_dir` generates
28 ↪ multiple events
29 # when a file was saved used vim or something else
30 # but we want to run `sphinx build` only once when a file was changed.
```

(continues on next page)

(continued from previous page)

```

30 # Thus `while true` is used.
31 # inotifywait (without *-m* key) generates one event then stops,
32 # then makedoc.py runs `shpihx build`
33 # then next iteration of infinitive cicle `while true` starts `inotifywait -m` once_
  ↪ again
34 while true; do
35     inotifywait -r -e modify -e move -e create -e delete "${DOCS_SOURCE_DIR}" 2>/dev/
  ↪ null \
36     && python "${SCRIPT_DIR}"/makedoc.py
37 done

```

1.3 run_coverage.py

This script runs tests and coverage on the *python_selenium* package.

Listing 3: source code

```

1  """
2  run_coverage.py
3  ++++++
4
5  Runs tests and coverage in *python_selenium/tests*.
6
7  Usage:
8
9  .. code-block:: bash
10
11     python run_coverage.py
12
13     | Author: shmakovpn <shmakovpn@yandex.ru>
14     | Date: 2020-01-28
15     """
16  import os
17  # The path to the folder of this script
18  SCRIPT_DIR: str = os.path.dirname(os.path.abspath(__file__))
19
20
21  def run_coverage() -> None:
22      """Executes *coverage run* command"""
23      coverage_cmd: str = f'coverage run --source=python_selenium -m pytest python_
  ↪ selenium/tests'
24      os.system(coverage_cmd)
25      report_cmd: str = 'coverage report'
26      os.system(report_cmd)
27      print('__END__')
28
29
30  if __name__ == '__main__':
31      run_coverage()

```

1.3.1 Autodoc documentation

run_coverage.py

Runs tests and coverage in *python_selenium/tests*.

Usage:

```
python run_coverage.py
```

Author: shmakovpn <shmakovpn@yandex.ru>

Date: 2020-01-28

`run_coverage.run_coverage()` → None
Executes *coverage run* command

1.4 python_selenium

python_selenium package documentation.

1.4.1 python_selenium/proxy.py

python_selenium/proxy.py module documentation

Listing 4: source code

```
1  """
2  python_selenium/proxy.py
3  ++++++
4
5  | Author: shmakovpn "shmakovpn@yandex.ru"
6  | Date: 2012-01-28
7  """
8
9
10 class Proxy:
11     """
12     Proxy configuration
13
14     :param str protocol: The protocol of the proxy: *'http'*, *'https'* or *'socks'*.
15                          Defaults to *'http'*.
16     :param str ip: The IP address of the proxy.
17                   Default to the empty string (proxy isn't using).
18     :param int port: The TCP port of the proxy.
19                     Default to 80.
20     :param str login: The username of the proxy user.
21                      Default to the empty string (authentication isn't using).
22     :param str password: The password of the proxy user.
23                          Default to the empty string.
24     """
25     def __init__(self,
```

(continues on next page)

(continued from previous page)

```

26         protocol: str = 'http',
27         ip: str = '',
28         port: int = 80,
29         login: str = '',
30         password: str = '') -> None:
31     self.protocol = protocol
32     self.ip = ip
33     self.port = port
34     self.login = login
35     self.password = password
36
37     @property
38     def _auth(self) -> str:
39         """
40         Returns an authentication part of the proxy URL
41         """
42         if not self.login:
43             return ''
44         if not self.password:
45             return f'{self.login}@'
46         return f'{self.login}:{self.password}@'
47
48     @property
49     def url(self) -> str:
50         """
51         Returns the proxy URL
52         """
53         if not self.ip:
54             return ''
55         return f'{self.protocol}://{self._auth}{self.ip}:{self.port}'
56
57     # @property
58     # def to_dict(self) -> Dict[str, str]:
59     #     return {
60     #         'http': f'{self.ip}:{self.port}',
61     #         'https': f'{self.ip}:{self.port}',
62     #         'socks': f'{self.ip}:{self.port}',
63     #     }
64
65     def __str__(self) -> str:
66         return self.url

```

Autodoc documentation

python_selenium/proxy.py

Author: shmakovpn “shmakovpn<yandex.ru>”

Date: 2012-01-28

class python_selenium.proxy.**Proxy** (protocol: str = 'http', ip: str = "", port: int = 80, login: str = "", password: str = "")

Proxy configuration

Parameters

- **protocol** (*str*) – The protocol of the proxy: *'http'*, *'https'* or *'socks'*. Defaults to *'http'*.
- **ip** (*str*) – The IP address of the proxy. Default to the empty string (proxy isn't using).
- **port** (*int*) – The TCP port of the proxy. Default to 80.
- **login** (*str*) – The username of the proxy user. Default to the empty string (authentication isn't using).
- **password** (*str*) – The password of the proxy user. Default to the empty string.

property url

Returns the proxy URL

Tests

Listing 5: test_proxy source code

```
1  """
2  ./python_selenium/tests/proxy/test_proxy.py
3  ++++++
4
5  Tests for proxy.py
6
7  | Author: shmakovpn <shmakovpn@yandex.ru>
8  | Date: 2021-01-28
9  """
10 from unittest import TestCase
11 from unittest import mock
12 from unittest.mock import PropertyMock, patch
13 from python_selenium.proxy import Proxy
14
15
16 class TestProxy(TestCase):
17     """Testing the Proxy class"""
18     def test_defaults(self) -> None:
19         """Testing default values"""
20         proxy: Proxy = Proxy()
21         self.assertEqual(proxy.protocol, 'http')
22         self.assertEqual(proxy.ip, '')
23         self.assertEqual(proxy.port, 80)
24         self.assertEqual(proxy.login, '')
25         self.assertEqual(proxy.password, '')
26
27     def test__auth(self) -> None:
28         """Testing getter of an authentication part of the proxy URL"""
29         proxy: Proxy = Proxy()
30         proxy.login = ''
31         proxy.password = ''
32         self.assertEqual(proxy._auth, '') # type: ignore
33         proxy.login = 'python'
34         self.assertEqual(proxy._auth, 'python@') # type: ignore
35         proxy.password = 'the_best'
36         self.assertEqual(proxy._auth, 'python:the_best@') # type: ignore
37
38     @patch.object(Proxy,
39                  '_auth',
40                  new_callable=PropertyMock)
```

(continues on next page)

(continued from previous page)

```

41 def test_url(self, mock__auth: PropertyMock) -> None:
42     """Testign getter of the proxy URL"""
43     mock__auth.side_effect = lambda: 'L:P@'
44     proxy: Proxy = Proxy()
45     proxy.ip = ''
46     proxy.port = 81
47     proxy.protocol = 'HTTP'
48     self.assertEqual(proxy.url, '')
49     mock__auth.assert_not_called()
50     proxy.ip = 'IP'
51     self.assertEqual(proxy.url, 'HTTP://L:P@IP:81')
52     mock__auth.assert_called_once_with()
53
54 @patch.object(Proxy, 'url', new_callable=PropertyMock)
55 def test__str__(self, mock_url: PropertyMock) -> None:
56     """Testing __str__ of Proxy"""
57     mock_url.side_effect = lambda: 'URL'
58     proxy: Proxy = Proxy()
59     self.assertEqual(str(proxy), 'URL')
60     mock_url.assert_called_once_with()

```

Tests autodoc documentation

`./python_selenium/tests/proxy/test_proxy.py`

Tests for proxy.py

Author: shmakovpn <shmakovpn@yandex.ru>

Date: 2021-01-28

class python_selenium.tests.proxy.test_proxy.**TestProxy** (methodName='runTest')

Testing the Proxy class

test__auth() → None

Testing getter of an authentication part of the proxy URL

test__str(mock_url: unittest.mock.PropertyMock) → None

Testing __str__ of Proxy

test_defaults() → None

Testing default values

test_url(mock__auth: unittest.mock.PropertyMock) → None

Testign getter of the proxy URL

sphinx hints.

2.1 Sphinx code-block types

Source materials are collected from [here](#).

List of available languages

abap
abnf
ada, ada95, ada2005
adl
agda
ahk, autohotkey
alloy
ampl
antlr-as, antlr-actionscript
antlr-cpp
antlr-csharp, antlr-c#
antlr-java
antlr-objc
antlr-perl
antlr-python
antlr-ruby, antlr-rb
antlr
apacheconf, aconf, apache
apl
applescript
arduino
as, actionscript
as3, actionscript3
aspectj
aspx-cs
aspx-vb
asy, asymptote

at, ambienttalk, ambienttalk/2
autoit
awk, gawk, mawk, nawk
basemake
bash, sh, ksh, shell
bat, batch, dosbatch, winbatch
bbcode
bc
befunge
blitzbasic, b3d, bplus
blitzmax, bmax
bnf
boo
boogie
brainfuck, bf
bro
bugs, winbugs, openbugs
c-objdump
c
ca65
cadl
camkes, idl4
cbmbas
ceylon
cfc
cfengine3, cf3
cfm
cfs
chai, chaiscript
chapel, chpl
cheetah, spitfire
cirru
clay
clean
clojure, clj
clojurescript, cljs
cmake
cobol
cobolfree
coffee-script, coffeescript, coffee
common-lisp, cl, lisp
componentpascal, cp
console, shell-session
control, debcontrol
coq
cpp, c++
cpp-objdump, c++-objdumb, cxx-objdump

cpsa
crnsh, pcmk
croc
cryptol, cry
csharp, c#
csound, csound-orc
csound-document, csound-csd
csound-score, csound-sco
css+django, css+jinja
css+erb, css+ruby
css+genshitext, css+genshi
css+lasso
css+mako
css+mako
css+mozpreproc
css+myghty
css+php
css+smarty
css
cucumber, gherkin
cuda, cu
cypher
cython, pyx, pyrex
d-objdump
d
dart
delphi, pas, pascal, objectpascal
dg
diff, udiff
django, jinja
docker, dockerfile
doscon
dpatch
dtd
duel, jbst, jsonml+bst
dylan-console, dylan-repl
dylan-lid, lid
dylan
earl-grey, earlgrey, eg
easytrieve
ebnf
ec
ecl
eiffel
elixir, ex, exs
elm
emacs, elisp, emacs-lisp

erb
erl
erlang
evoque
extempore
ezhil
factor
fan
fancy, fy
felix, flx
fish, fishshell
flatline
fortran
fortranfixed
foxpro, vfp, clipper, xbase
fsharp
gap
gas, asm
genshi, kid, xml+genshi, xml+kid
genshitext
glsl
gnuplot
go
golo
gooddata-cl
gosu
groff, nroff, man
groovy
gst
haml
handlebars
haskell, hs
haxml, hxml
hexdump
hsail, hsa
html+cheetah, html+spitfire, htmlcheetah
html+django, html+jinja, htmdljango
html+evoque
html+genshi, html+kid
html+handlebars
html+lasso
html+mako
html+mako
html+myghty
html+php
html+smarty
html+twig

html+velocity
html
http
hx, haxe, hxml
hybris, hy
hylang
i6t
idl
idris, idr
iex
igor, igorpro
inform6, i6
inform7, i7
ini, cfg, dosini
io
ioke, ik
irc
isabelle
j
jade
jags
jasmin, jasminxt
java
javascript+mozpreproc
jcl
jlcon
js+cheetah, javascript+cheetah, js+spitfire, javascript+spitfire
js+django, javascript+django, js+jinja, javascript+jinja
js+erb, javascript+erb, js+ruby, javascript+ruby
js+genshitext, js+genshi, javascript+genshitext, javascript+genshi
js+lasso, javascript+lasso
js+mako, javascript+mako
js+mako, javascript+mako
js+myghty, javascript+myghty
js+php, javascript+php
js+smarty, javascript+smarty
js, javascript
jsgf
json
jsonld, json-ld
jsp
julia, jl
kal
kconfig, menuconfig, linux-config, kernel-config
koka
kotlin
lagda, literate-agda

lasso, lassoscript
lcry, literate-cryptol, lcryptol
lean
less
lhs, literate-haskell, lhaskell
lidr, literate-idris, lidris
lighty, lighttpd
limbo
liquid
live-script, livescript
llvm
logos
logtalk
lsl
lua
make, makefile, mf, bsdmake
mako
mako
maql
mask
mason
mathematica, mma, nb
matlab
matlabsession
minid
modelica
modula2, m2
monkey
moocode, moo
moon, moonscript
mozhashpreproc
mozpercentpreproc
mql, mq4, mq5, mql4, mql5
mscgen, msc
mupad
mxml
myghty
mysql
nasm
ncl
nemerle
nesc
newlisp
newspeak
nginx
nimrod, nim
nit

nixos, nix
nsis, nsi, nsh
numpy
objdump-nasm
objdump
objective-c++, objectivec++, obj-c++, objc++
objective-c, objectivec, obj-c, objc
objective-j, objectivej, obj-j, objj
ocaml
octave
odin
ooc
opa
openedge, abl, progress
pacmanconf
pan
parasail
pawn
perl, pl
perl6, pl6
php, php3, php4, php5
pig
pike
pkgconfig
plpgsql
postgresql, postgres
postscript, postscr
pot, po
pov
powershell, posh, ps1, psm1
praat
prolog
properties, jproperties
protobuf, proto
ps1con
psql, postgresql-console, postgres-console
puppet
py3tb
pycon
pypylog, pypy
pytb
python, py, sage
python3, py3
qbasic, basic
qml, qbs
qvto, qvt
racket, rkt

ragel-c
ragel-cpp
ragel-d
ragel-em
ragel-java
ragel-objc
ragel-ruby, ragel-rb
ragel
raw
rb, ruby, duby
rbcon, irb
rconsole, rout
rd
rebol
red, red/system
redcode
registry
resource, resourcebundle
rexx, arexx
rhtml, html+erb, html+ruby
roboconf-graph
roboconf-instances
robotframework
rql
rsl
rst, rest, restructuredtext
rts, trafficscript
rust
sass
sc, supercollider
scala
scaml
scheme, scm
scilab
scss
shen
silver
slim
smali
smalltalk, squeak, st
smarty
sml
snobol
sourceslist, sources.list, debsources
sp
sparql
spec

splus, s, r
sql
sqlite3
squidconf, squid.conf, squid
ssp
stan
swift
swig
systemverilog, sv
tads3
tap
tcl
tcsh, csh
tcshcon
tea
termcap
terminfo
terraform, tf
tex, latex
text
thrift
todotxt
trac-wiki, moin
treetop
ts, typescript
turtle
twig
typoscript
typoscriptcssdata
typoscripthtmldata
urbiscript
vala, vapi
vb.net, vbnet
vcl
vclsnippets, vclsnippet
vctreestatus
velocity
verilog, v
vgl
vhdl
vim
wdiff
x10, xten
xml+cheetah, xml+spitfire
xml+django, xml+jinja
xml+erb, xml+ruby
xml+evoque

xml+lasso
xml+mako
xml+mako
xml+myghty
xml+php
xml+smarty
xml+velocity
xml
xquery, xqy, xq, xql, xqm
xslt
xtend
xul+mozpreproc
yaml+jinja, salt, sls
yaml
zephir

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

m

makedoc, [4](#)

p

python_selenium.proxy, [7](#)

python_selenium.tests.proxy.test_proxy,
[9](#)

r

run_coverage, [6](#)

INDEX

M

makedoc
 module, 4
module
 makedoc, 4
 python_selenium.proxy, 7
 python_selenium.tests.proxy.test_proxy,
 9
 run_coverage, 6

P

Proxy (*class in python_selenium.proxy*), 7
python_selenium.proxy
 module, 7
python_selenium.tests.proxy.test_proxy
 module, 9

R

run_coverage
 module, 6
run_coverage() (*in module run_coverage*), 6
run_sphinx() (*in module makedoc*), 4

T

test__auth() (*python_selenium.tests.proxy.test_proxy.TestProxy*
 method), 9
test__str() (*python_selenium.tests.proxy.test_proxy.TestProxy*
 method), 9
test_defaults() (*python_selenium.tests.proxy.test_proxy.TestProxy*
 method), 9
test_url() (*python_selenium.tests.proxy.test_proxy.TestProxy*
 method), 9
TestProxy (class in *python_selenium.tests.proxy.test_proxy*),
 9

U

url() (*python_selenium.proxy.Proxy* property), 8