



Automating Cisco ACI with Python

Python Modules

ine.com

One of the best features of Python as a language, is its rich builtin library. To use external modules, you must first import them:

```
In [100]: import random
```

```
In [109]: random.randint(0, 99)
```

```
Out[109]: 85
```

A module is a file containing Python definitions and statements. The file name is the module name with the suffix `.py` appended. Within a module, the module's name (as a string) is available as the value of the global variable `__name__`.

For instance, we can use a text editor to create a file called `fib.py` with the following contents:

```
# Fibonacci numbers module

def fib(n):    # write Fibonacci series up to n
    a, b = 0, 1
    while a < n:
        print(a, end=' ')
        a, b = b, a+b
    print()
```

```
In [1]: from data import fibo
```

```
In [2]: fibo.__dir__()
```

```
Out[2]: ['__name__',
         '__doc__',
         '__package__',
         '__loader__',
         '__spec__',
         '__file__',
         '__cached__',
         '__builtins__',
         'fib',
         'get_list_length']
```

```
In [3]: fibo.fib(10)
```

```
0 1 1 2 3 5 8
```

```
In [5]: from data.fibo import *
```

```
In [6]: from data.fibo import fib, get_list_length
```

```
In [8]: from data.fibo import fib, get_list_length, other
```

```
-----
ImportError                                Traceback (most recent call last)
<ipython-input-8-11db9e2fd137> in <module>
----> 1 from data.fibo import fib, get_list_length, other

ImportError: cannot import name 'other' from 'data.fibo' (/Users/matiascaputti/Desktop/acitoolkit-master/day 1/data/fibo.py)
```

```
In [7]: help(fib)
```

Help on function fib in module data.fibo: <https://t.me/learningnets>

```
fib(n)
Write Fibonacci series up to n
```

Executing modules as scripts

Python files are called modules and they are identified by the .py file extension. A module can define functions, classes, and variables.

You can run a Python module with:

```
python module.py <arguments>
```

in that case, the code in the module will be executed, just as if you imported it.

```
In [10]: !python data/module.py
```

```
File two __name__ is set to: __main__
(executed as script)
```

As you can see when the interpreter runs a module as a script, the `__name__` variable will be set as `"__main__"` if the module that is being run is the main program.

But if the code is importing the module from another module, then the `__name__` variable will be set to that module's name.

Knowing that, you can limit the execution of the module to the case when the `__name__` variable is set to `__main__`:

```
if __name__ == "__main__":
    # statements
else:
    # do nothing
```

you can make the file usable as a script as well as an importable module, because the code that parses the command line will only run if the module is executed as the "main" file:

```
$ python data/module.py
```

If the module is imported, the code is not run:

```
>>> import data.module
>>>
```

```
In [3]: print("This file __name__ is set to: {}".format(__name__))
```

```
This file __name__ is set to: __main__
```

```
In [4]: import data.module
```

```
print("This file __name__ is set to: {}".format(__name__))
```

```
This file __name__ is set to: __main__
```

But if we run `module.py` directly we will see that its name is set to `__main__`

```
In [5]: !python data/module.py
```

```
File two __name__ is set to: __main__
(executed as script)
```

```
In [6]: if __name__ == "__main__":
        print("Do something when ran directly")
        else:
            print("Do other thing or nothing when imported")
```

```
Do something when ran directly
```

Arguments when executing modules as scripts

Python 3 supports a number of different ways of handling command line arguments:

```
In [40]: !python data/module.py
```

```
File two __name__ is set to: __main__
(executed as script)
```

```
In [41]: !python data/module_2.py
```

```
(executed as script)
Number of arguments: 0 arguments.
Argument List: []
```

```
Error: missing arguments
```

```
In [42]: !python data/module_2.py --num_1 4 --num_2 6
```

```
(executed as script)
Number of arguments: 4 arguments.
Argument List: ['--num_1', '4', '--num_2', '6']

-----

Set output num_1 to 4
Set output num_2 to 6
10
```

```
In [44]: !python data/module_2.py --num_2 6 --num_1 4
```

```
(executed as script)
Number of arguments: 4 arguments.
Argument List: ['--num_2', '6', '--num_1', '4']

-----

Set output num_1 to 4
Set output num_2 to 6
10
```

```
In [45]: !python data/module_2.py --num_1 4
```

```
(executed as script)
Number of arguments: 2 arguments.
Argument List: ['--num_1', '4']

-----

Set output num_1 to 4
Error: missing arguments
```

```
In [43]: !python data/module_2.py --num_1 4 --num_2 6 --return_zero
```

```
(executed as script)
Number of arguments: 5 arguments.
Argument List: ['--num_1', '4', '--num_2', '6', '--return_zero']

-----

Set output num_1 to 4
Set output num_2 to 6
0
```

Installing external libraries

```
In [46]: !python --version
```

```
Python 3.8.6
```

pip is a standard package manager used to install and maintain packages for Python. The Python standard library comes with a collection of built-in functions and built-in packages.

```
In [47]: !pip --version
```

```
pip 20.3 from /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages/pip (python 3.8)
```

```
In [48]: !pip -h
```

```
Usage:
  pip <command> [options]
```

```
Commands:
  install           Install packages.
  download          Download packages.
  uninstall         Uninstall packages.
  freeze           Output installed packages in requirements format.
  list             List installed packages.
  show             Show information about installed packages.
  check            Verify installed packages have compatible dependencies.
  config           Manage local and global configuration.
  search           Search PyPI for packages.
  cache            Inspect and manage pip's wheel cache.
  wheel            Build wheels from your requirements.
  hash            Compute hashes of package archives.
  completion       A helper command used for command completion.
  debug           Show information useful for debugging.
  help            Show help for commands.
```

```
General Options:
  -h, --help           Show help.
  --isolated           Run pip in an isolated mode, ignoring
                      environment variables and user configuration.
  -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. Option is additive, and can be
                      used up to 3 times (corresponding to WARNING,
                      ERROR, and CRITICAL logging levels).
  --log <path>       Path to a verbose appending log.
  --no-input          Disable prompting for input.
  --proxy <proxy>    Specify a proxy in the form
                      [user:passwd@]proxy.server:port.
  --retries <retries> Maximum number of retries each connection should
                      attempt (default 5 times).
  --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, (a)bort.
  --trusted-host <hostname> Mark this host or host:port pair as trusted,
                      even though it does not have valid or any HTTPS.
  --cert <path>     Path to alternate certificate bundle.
```

<https://t.me/learningnets>

```

--client-cert <path> Path to SSL client certificate, a single file
                      containing the private key and the certificate
                      in PEM format.
--cache-dir <dir> Store the cache data in <dir>.
--no-cache-dir Disable the cache.
--disable-pip-version-check Don't periodically check PyPI to determine
                             whether a new version of pip is available for
                             download. Implied with --no-index.
--no-color Suppress colored output.
--no-python-version-warning Silence deprecation warnings for upcoming
                             unsupported Python versions.
--use-feature <feature> Enable new functionality, that may be backward
                             incompatible.
--use-deprecated <feature> Enable deprecated functionality, that will be
                             removed in the future.

```

Before you make any installs, it is a good idea to see what is already installed. You can use `pip list` in the command line, and it will display the Python packages in your current working environment in alphabetical order.

In [52]: `!pip list`

```

Package              Version
-----
acitoolkit            0.4
anyio                  3.1.0
appnope               0.1.0
argon2-cffi           20.1.0
async-generator        1.10
attrs                 20.2.0
autopep8              1.5.7
Babel                 2.9.1
backcall              0.2.0
bash-kernel           0.7.2
bleach                3.2.1
certifi               2020.11.8
cffi                  1.14.3
chardet               3.0.4
click                 7.1.2
colorful              0.5.4
cyclor                 0.10.0
decorator             4.4.2
deepdiff              5.5.0
defusedxml            0.6.0
Deprecated            1.2.10
docstring-to-markdown 0.9
dominate              2.6.0
entrypoints           0.3
Flask                 1.1.2
Flask-Admin           1.5.8
Flask-Bootstrap       3.3.7.1
Flask-Cors            3.0.10
Flask-HTTPAuth        4.3.0
Flask-SQLAlchemy      2.5.1
Flask-WTF             0.14.3
gitdb                 4.0.7
GitPython             3.1.14
graphviz              0.16
greenlet              1.0.0
idna                  2.10
ine-exercises-tools   0.0.7
ipaddress             1.0.23
ipykernel             5.3.4
ipython               7.19.0
ipython-genutils      0.2.0
ipywidgets            7.5.1
itsdangerous          1.1.0
jedi                  0.18.0
jedi-language-server  0.32.0
Jinja2                2.11.2
joblib                0.17.0
json5                 0.9.5
jsonschema            3.2.0
jupyter               1.0.0
jupyter-client        6.1.7
jupyter-console       6.2.0
jupyter-contrib-core  0.3.3
jupyter-core          4.6.3
jupyter-lsp           1.3.0
jupyter-nbextensions-configurator 0.4.1
jupyter-server        1.8.0
jupyterlab            3.0.16
jupyterlab-lsp        3.7.0
jupyterlab-pygments   0.1.2
jupyterlab-server     2.6.0
kiwisolver            1.3.1
lxml                  4.6.3
MarkupSafe            1.1.1
matplotlib            3.3.2
mistune               0.8.4
multitasking          0.0.9
nbclassic             0.3.1
nbclient              0.5.1
nbconvert             6.0.7
nbformat              5.0.8
nest-asyncio          1.4.2
nltk                  3.5
notebook              6.1.4
numpy                 1.19.4
ordered-set           4.0.2
packaging             20.4
pandas                1.1.4
pandocfilters         1.4.3
parso                 0.8.2

```

<https://t.me/learningnets>

```

pexpect 4.8.0
pickleshare 0.7.5
Pillow 8.0.1
pip 20.3
prometheus-client 0.8.0
prompt-toolkit 3.0.8
ptyprocess 0.6.0
py-radix 0.10.0
pycodestyle 2.7.0
pycparser 2.20
pydantic 1.8.2
PyGithub 1.54
pygls 0.10.3
Pygments 2.7.2
PyJWT 1.7.1
PyMySQL 1.0.2
pyparsing 2.4.7
pyrsistent 0.17.3
python-dateutil 2.8.1
pytz 2020.4
PyYAML 5.3.1
pyzmq 19.0.2
qtconsole 4.7.7
QtPy 1.9.0
regex 2020.10.28
requests 2.25.0
requests-toolbelt 0.9.1
rise 5.7.1
scikit-learn 0.23.2
scipy 1.5.4
seaborn 0.11.0
Send2Trash 1.5.0
setuptools 49.2.1
six 1.15.0
smap 4.0.0
sniffio 1.2.0
SQLAlchemy 1.4.13
svgwrite 1.4.1
tabulate 0.8.9
terminado 0.9.1
testpath 0.4.4
threadpoolctl 2.1.0
toml 0.10.2
tornado 6.1
tqdm 4.51.0
traitlets 5.0.5
Tree 0.2.4
typeguard 2.12.1
typing-extensions 3.10.0.0
urllib3 1.26.1
visitor 0.1.3
wcwidth 0.2.5
webencodings 0.5.1
websocket-client 0.59.0
Werkzeug 1.0.1
widgetsnextension 3.5.1
wrap 1.12.1
WTFORMS 2.3.3
yfinance 0.1.59

```

WARNING: You are using pip version 20.3; however, version 21.1.2 is available.

You should consider upgrading via the '/Users/matiascaputti/.pyenv/versions/3.8.6/bin/python3.8 -m pip install --upgrade pip' command.

Installing a new package

Let's say we want to install the `acitoolkit` package. We can do that executing the following command:

```
In [53]: !pip install acitoolkit
```

```

Requirement already satisfied: acitoolkit in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (0.4)
Requirement already satisfied: requests in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (2.25.0)
Requirement already satisfied: websocket-client>0.33.0 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (0.59.0)
Requirement already satisfied: gitpython in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (3.1.14)
Requirement already satisfied: flask-httpauth in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (4.3.0)
Requirement already satisfied: flask-sqlalchemy in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (2.5.1)
Requirement already satisfied: flask-admin in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.5.8)
Requirement already satisfied: flask-bootstrap in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (3.3.7.1)
Requirement already satisfied: flask-wtf in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (0.14.3)
Requirement already satisfied: flask-cors in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (3.0.10)
Requirement already satisfied: flask in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.1.2)
Requirement already satisfied: pymysql in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.0.2)
Requirement already satisfied: tabulate in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (0.8.9)
Requirement already satisfied: py-radix in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (0.10.0)
Requirement already satisfied: jsonschema in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (3.2.0)
Requirement already satisfied: graphviz in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (0.16)
Requirement already satisfied: ipaddress in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.0.23)
Requirement already satisfied: deepdiff in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (5.5.0)
Requirement already satisfied: ordered-set==4.0.2 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from deepdiff->acitoolkit) (4.0.2)

```

<https://t.me/learningnets>

```
Requirement already satisfied: Werkzeug>=0.15 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask->acitoolkit) (1.0.1)
Requirement already satisfied: click>=5.1 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask->acitoolkit) (7.1.2)
Requirement already satisfied: Jinja2>=2.10.1 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask->acitoolkit) (2.11.2)
Requirement already satisfied: itsdangerous>=0.24 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask->acitoolkit) (1.1.0)
Requirement already satisfied: flask in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.1.2)
Requirement already satisfied: wtforms in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-admin->acitoolkit) (2.3.3)
Requirement already satisfied: flask in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.1.2)
Requirement already satisfied: dominate in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-bootstrap->acitoolkit) (2.6.0)
Requirement already satisfied: visitor in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-bootstrap->acitoolkit) (0.1.3)
Requirement already satisfied: flask in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.1.2)
Requirement already satisfied: Six in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-cors->acitoolkit) (1.15.0)
Requirement already satisfied: flask in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.1.2)
Requirement already satisfied: SQLAlchemy>=0.8.0 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-sqlalchemy->acitoolkit) (1.4.13)
Requirement already satisfied: flask in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.1.2)
Requirement already satisfied: itsdangerous>=0.24 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask->acitoolkit) (1.1.0)
Requirement already satisfied: flask in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from acitoolkit) (1.1.2)
Requirement already satisfied: wtforms in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-admin->acitoolkit) (2.3.3)
Requirement already satisfied: gitdb<5,>=4.0.1 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from gitpython->acitoolkit) (4.0.7)
Requirement already satisfied: smmap<5,>=3.0.1 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from gitdb<5,>=4.0.1->gitpython->acitoolkit) (4.0.0)
Requirement already satisfied: MarkupSafe>=0.23 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from Jinja2>=2.10.1->flask->acitoolkit) (1.1.1)
Requirement already satisfied: Six in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-cors->acitoolkit) (1.15.0)
Requirement already satisfied: pyparsing>=0.14.0 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from jsonschema->acitoolkit) (0.17.3)
Requirement already satisfied: attrs>=17.4.0 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from jsonschema->acitoolkit) (20.2.0)
Requirement already satisfied: setuptools in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from jsonschema->acitoolkit) (49.2.1)
Requirement already satisfied: chardet<4,>=3.0.2 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from requests->acitoolkit) (3.0.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from requests->acitoolkit) (1.26.1)
Requirement already satisfied: idna<3,>=2.5 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from requests->acitoolkit) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from requests->acitoolkit) (2020.11.8)
Requirement already satisfied: greenlet!=0.4.17 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from SQLAlchemy>=0.8.0->flask-sqlalchemy->acitoolkit) (1.0.0)
Requirement already satisfied: Six in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from flask-cors->acitoolkit) (1.15.0)
Requirement already satisfied: MarkupSafe>=0.23 in /Users/matiascaputti/.pyenv/versions/3.8.6/lib/python3.8/site-packages (from Jinja2>=2.10.1->flask->acitoolkit) (1.1.1)
WARNING: You are using pip version 20.3; however, version 21.1.2 is available.
You should consider upgrading via the '/Users/matiascaputti/.pyenv/versions/3.8.6/bin/python3.8 -m pip install --upgrade pip' command.
```

Installing an specific version

```
In [ ]: !pip install acitoolkit==0.4
```

Upgrading an already installed package

```
In [ ]: !pip install --upgrade acitoolkit==0.4
```

Installing packages with a requirements.txt file

This will go line by line through your requirements.txt file and install all the packages listed.

```
In [ ]: !pip install -r requirements.txt
```

Create a requirements.txt with my installed packages

Creating your own requirements.txt is useful to share your installed packages with others.

```
In [55]: !pip freeze > requirements.txt
```

Uninstall a certain package

```
In [ ]: !pip uninstall acitoolkit
```

