

Tips

Find all installed modules

```
help("modules");
```

Virtual Environment

- [How to Install and Manage Python Versions in Linux - Make Tech Easier](#)

Python 3.4+

```
mkdir myproject
cd myproject
python -m venv env

# Activate the venv
source env/bin/activate

# Delete the venv
deactivate
rm -rf env
```

With virtualenv and [virtualenvwrapper](#)

```
# Installing virtualenv and virtualenvwrapper
sudo pip install virtualenv virtualenvwrapper

# Update the profile ~/.bashrc
# Add the following lines

# Python virtualenv and virtualenvwrapper
export WORKON_HOME=$HOME/.virtualenvs
export VIRTUALENVWRAPPER_PYTHON=/usr/bin/python3
```

```
source /usr/local/bin/virtualenvwrapper.sh
```

```
# Reload the profile
```

```
source ~/.bashrc
```

```
# Creating python virtual environment
```

```
# The py3cv3 is a self-defined name
```

```
mkvirtualenv py3cv3 -p python3
```

```
# Enter the specified virtual environment
```

```
workon py3cv3
```

```
# Exit the the specified virtual environment
```

```
deactivate
```

```
# List all of the environments.
```

```
lsvirtualenv
```

```
# Remove an environment
```

```
rmvirtualenv py3cv3
```

Timestamp

```
timestamp = datetime.datetime.now()
```

```
print("It is {}".format(timestamp.strftime("%A %d %B %Y %I:%M:%S%p")))
```

Math

```
total += 1
```

If-then

```
# Boolean, none
```

```
if motion is not None:
```

```
if not flag:
```

```
# Number
```

```
if delay > 0:
```

```
if delay == 0:
```

```
if total > frameCount:
```

```

# String
if "blue" in style:
if authors.startswith('('):
    authors = authors.lstrip('(').rstrip(')')

# One-liner
def doi_url(d): return f'http://{d}' if d.startswith('doi.org') else f'http://doi.org/{d}'

# Multiple conditions
temperature = 25
if temperature > 30:
    print('Hot')
elif temperature > 20 and temperature <= 30:
    print('Warm')
else:
    print('Cool')

# Reverse the True
temperature = 15
if not temperature > 20:
    print('Cool')
#
temperature = 25
humidity = 55
rain = 0
if temperature > 30 or humidity < 70 and not rain > 0:
    print('Dry conditions')

```

Command Arguments

- Manual: <https://docs.python.org/3/library/argparse.html>

```

import argparse

# construct the argument parser and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-i", "--interval", required=False,
                help="Seconds to Interval (Default:30)", default="30", type=int)
ap.add_argument("-o", "--output", required=False,
                help="Path to Output Logs (Default:std-out)")

```

```

ap.add_argument("mac",
                help="MAC address of LYWSD02 device", nargs="+")
args = vars(ap.parse_args())

# Usage
intv = args["interval"]
logfile = args["output"]

```

```

#
# Nagios2 HTTP proxy test
#
# usage: check_http_proxy --proxy=proxy:port --auth=user:pass --url=url --timeout=10 --warranty=5 --
expect=content

import sys
import getopt

def get_cmdline_cfg():
    try:
        opts, args = getopt.getopt(
            sys.argv[1:],
            "p:a:t:w:e:u:",
            ["proxy=", "auth=", "timeout=", "warranty=", "expect=", "url="]
        )
    except getopt.GetoptError, err:
        print("SCRIPT CALLING ERROR: {0}".format(str(err)))

    ### Build cfg dictionary
    cfg = {}
    for o, a in opts:
        if o in ("-p", "--proxy"):
            cfg["proxy"] = a
        elif o in ("-a", "--auth"):
            cfg["auth"] = a
        elif o in ("-t", "--timeout"):
            cfg["timeout"] = float(a)
        elif o in ("-w", "--warranty"):
            cfg["warranty"] = float(a)
        elif o in ("-e", "--expect"):
            cfg["expect"] = a

```

```
elif o in ("-u", "--url"):
    cfg["url"] = a

# These are required
for req_param in ("url", "proxy"):
    if req_param not in cfg:
        print("Missing parameter: {0}".format(req_param))

return cfg

# Usage
if __name__ == '__main__':
    cfg = get_cmdline_cfg()

    if "auth" in cfg:
        proxy_url = "http://{auth}@{proxy}/".format(**cfg)
    else:
        proxy_url = "http://{proxy}/".format(**cfg)
```

Revision #38

Created 9 November 2022 14:33:15 by Admin

Updated 30 January 2024 15:32:31 by Admin