RUTX50 Python3

The information in this page is updated in accordance with firmware version **RUTX_R_00.07.06.10**.

□

Contents

Summary

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

Teltonika **Python3** package uses Python version **3.9.7**. The list of all modules included in the package can be found bellow.

This manual page provides an overview of Python3 functionality in RUTX50 devices.

Note: Python is additional software that can be installed from the **System → Package Manager** page.

Disclaimer: before installing Python3 package make sure that the target device has a sufficient amount of free storage space!!!

Python3 Usage

After installing the package a new command will become available in CLI (Command-line interface) which enables the device to invoke Python scripts or program files and allows access to the Python interpreter interface.

To invoke a Python script or program file use the command **python <python_file>** and replace **<python_file>** with the relative or absolute path to the Python script or program file.

```
root@Teltonika-RUTXXX:~# python /test_py.py
Hello world!
```

Alternatively, using just the command **python** will let you enter the interpreter and write and

execute your code there.

```
root@Teltonika-RUTXXX:~# python
Python 3.9.7 (default, Mar 23 2023, 08:32:35)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello world!")
Hello world!
>>>
```

For more information on how to use **python** command use the command **python --help** in the device's CLI and for how to use Python interpreter use the command **help()** in the Python interpreter interface.

```
root@Teltonika-RUTXXX:~# python --help
```

```
root@Teltonika-RUTXXX:~# python
Python 3.9.7 (default, Mar 23 2023, 08:32:35)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> help()

Welcome to Python 3.9's help utility!

If this is your first time using Python, you should definitely check out
the tutorial on the Internet at https://docs.python.org/3.9/tutorial/.

Enter the name of any module, keyword, or topic to get help on writing
Python programs and using Python modules.  To quit this help utility and
return to the interpreter, just type "quit".

To get a list of available modules, keywords, symbols, or topics, type
"modules", "keywords", "symbols", or "topics".  Each module also comes
with a one-line summary of what it does; to list the modules whose name
or summary contain a given string such as "spam", type "modules spam".

help>
```

To exit the Python interpreter interface and return to device's CLI use the command **exit()** or **quit()**.

```
root@Teltonika-RUTXXX:~# python
Python 3.9.7 (default, Mar 23 2023, 08:32:35)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
root@Teltonika-RUTXXX:~#
```

```
root@Teltonika-RUTXXX:~# python
Python 3.9.7 (default, Mar 23 2023, 08:32:35)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> quit()
```

root@Teltonika-RUTXXX:~#

Python3 Modules

abc
aifc
antigravity
argparse
array
ast
asynchat
asyncio
asyncore
atexit
audioop
base64
bdb
binascii
binhex
bisect
builtins
bz2
cProfile
calendar
cgi
cgitb
chunk
cmath
cmd
code
codecs
codeop
collections
colorsys
compileall
concurrent
configparser
contextlib
contextvars
copy
copyreg
crypt
csv
ctypes
curses
dataclasses
datetime
dbm
decimal
difflib
dis
distutils
doctest
email
encodings
enum
errno
faulthandler
fcntl
filecmp
fileinput
fnmatch
formatter
fractions
ftplib
functools
gc
genericpath
getopt
getpass
gettext
glob
graphlib
grp
gzip
hashlib
heapq
hmac
html
http
imaplib
imghdr
imp
importlib
inspect
io
ipaddress
itertools
json
keyword
linecache
locale
logging
mailbox
mailcap
marshal
math
mimetypes
mmap
modulefinder
multiprocessing
netrc
nntplib
ntpath
nturl2path
numbers
opcode
operator
optparse
os
ossaudiodev
parser
pathlib
pdb
pickle
pickletools
pipes
pkgutil
platform
plistlib
poplib
posix
posixpath
pprint
profile
pstats
pty
pwd
py_compile
pyclbr
pydoc
pydoc_data
pyexpat
queue
quopri
random
re
readline
reprlib
resource
rlcompleter
runpy
sched
secrets
select
selectors
shelve
shlex
shutil
signal
site
smtpd
smtplib
sndhdr
socket
socketserver
spwd
sqlite3
sre_compile
sre_constants
sre_parse
ssl
stat
statistics
string
stringprep
struct
subprocess
sunau
symbol
symtable
sys
sysconfig
syslog
tabnanny
tarfile
telnetlib
tempfile
termios
textwrap
this
threading
time
timeit
token
tokenize
trace
traceback
tracemalloc
tty
turtle
types
typing
unicodedata
unittest
urllib
uu
uuid
venv
warnings
wave
weakref
wsgiref
xdrlib
xml
xmlrpc
xxlimited
xxsubtype
zipapp
zipfile
zipimport
zlib
zoneinfo