

# **pexpect-executor**

**Author:** Adrian DC

**Description:** Automate interactive CLI tools actions

**Date:** 02/02/2025

**Version:** 4.3.0-6-gd17823e

## **► Usage**

pypi v4.3.0 python 3.8 | 3.9 | 3.10 | 3.11 | 3.12 downloads 622/month license Apache License 2.0

pipeline passed bugs 0 code smells 0 coverage 100% lines of code 1.2k

quality gate passed

pre-commit enabled commitizen friendly gcil enabled pre-commit-crocodile enabled

Automate interactive CLI tools actions to create previews or tests in Python

**Documentation:** <https://radiandevcore.gitlab.io/tools/pexpect-executor>

**Package:** <https://pypi.org/project/pexpect-executor/>

---

## Table of contents

- [Preview](#)
- [CLI examples](#)
- [Python examples](#)
- [Usage](#)
- [Userspace available settings](#)
- [Supported systems](#)
- [Environment available configurations](#)
- [Dependencies](#)
- [References](#)

## Preview

```
adriandc@previews:~/pexpect-executor$ █
```

## CLI examples

```
pexpect-executor --help  
pexpect-executor --down --down --down --down --down --space --enter -- gcil -H -B  
pexpect-executor --press a --press a --enter -- gcil  
pexpect-executor --ctrl c -- gcil
```

## Python examples

```
#!/usr/bin/env python3

# Modules libraries
from pexpect_executor import Executor

# Configure
Executor.configure(host='previews', tool='pexpect-executor')

# List
Executor('ls -la', workdir='/').\
    finish()

# More
Executor('ls -la | more -5', delay_press=0.5, workdir='/').\
    read().\
    press('s').\
    read().\
    press('s').\
    read().\
    press('b').\
    read().\
    press('b').\
    read().\
    press(' ').\
    read().\
    wait(1).\
    press(Executor.KEY_ENTER).\
    read().\
    wait(1).\
    press('q').\
    finish()

# Prompt
Executor('').\
    finish()
```

## Usage

```
usage: pexpect-executor [-h] [--version] [--no-color] [--update-check] [--settings] [--set GROUP KEY VAL]
                        [--delay-init SECS] [--delay-press SECS] [--delay-prompt SECS] [--hold-prompt] [--host HOST]
                        [--tool TOOL] [--mask STRINGS] [--workdir WORKDIR] [--up] [--down] [--left] [--right]
                        [--enter] [--space] [--press KEY] [--ctrl KEY] [--read] [--wait SECS] [--finish] [--]
                        [command ...]
```

**pexpect-executor:** Automate interactive CLI tools actions

### internal arguments:

```
-h, --help           # Show this help message
--version            # Show the current version
--no-color           # Disable colors outputs with 'NO_COLOR=1'
                    # (or default settings: [themes] > no_color)
--update-check       # Check for newer package updates
--settings           # Show the current settings path and contents
--set GROUP KEY VAL # Set settings specific 'VAL' value to [GROUP] > KEY
                    # or unset by using 'UNSET' as 'VAL'
```

### configuration arguments:

```
--delay-init SECS   # Delay the initial action execution (in s, default: 1.0)
--delay-press SECS  # Delay the press actions execution (in s, default: 0.5)
--delay-prompt SECS # Delay the prompt actions execution (in s, default: 1.0)
--hold-prompt       # Hold the prompt execution without a new line
--host HOST         # Configure the host name (default: preview, env: EXECUTOR_HOST)
--tool TOOL         # Configure the tool name (default: executor, env: EXECUTOR_TOOL)
--mask STRINGS      # Mask specific strings from console outputs (credentials for example)
--workdir WORKDIR   # Use a specific working directory path
```

### actions arguments:

```
--up                # Press the <UP> key
--down              # Press the <DOWN> key
--left              # Press the <LEFT> key
--right             # Press the <RIGHT> key
--enter             # Press the <ENTER> key
--space             # Press the <SPACE> key
--press KEY         # Press the specified <KEY>
--ctrl KEY          # Press the specified Ctrl+<KEY>
--read              # Read the buffered data output (forced)
--wait SECS         # Wait the specified time (in s, example: 1.0)
--finish           # Finish the execution (forced)
```

### positional arguments:

```
--                # Positional arguments separator (recommended)
command           # Command arguments to execute (use -- to separate)
```

## Userspace available settings

`pexpect-executor` creates a `settings.ini` configuration file in a userspace folder.

For example, it allows to disable the automated updates daily check ( `[updates] > enabled` )

The `settings.ini` file location and contents can be shown with the following command:

```
pexpect-executor --settings
```

## Supported systems

Systems	Supported
Linux (shell)	✓
macOS (shell)	?
Windows (shell)	~
Android (Termux)	✓

## Environment available configurations

`pexpect-executor` uses `colored` for colors outputs.

If colors of both outputs types do not match the terminal's theme, an environment variable `NO_COLOR=1` can be defined to disable colors.

## Dependencies

- [colored](#): Terminal colors and styles
  - [pexpect](#): Interactive console applications controller
  - [setuptools](#): Build and manage Python packages
  - [update-checker](#): Check for package updates on PyPI
  - [wexpect](#): Windows alternative of pexpect
- 

## References

- [commitizen](#): Simple commit conventions for internet citizens
- [git-cliff](#): CHANGELOG generator
- [gitlab-release](#): Utility for publishing on GitLab
- [gcil](#): Launch .gitlab-ci.yml jobs locally
- [mkdocs](#): Project documentation with Markdown
- [mkdocs-exporter](#): Exporter plugin for mkdocs documentation
- [mkdocs-material](#): Material theme for mkdocs documentation
- [mypy](#): Optional static typing for Python
- [pre-commit](#): A framework for managing and maintaining pre-commit hooks
- [pre-commit-crocodile](#): Git hooks intended for developers using pre-commit
- [PyPI](#): The Python Package Index
- [termtosvg](#): Record terminal sessions as SVG animations
- [twine](#): Utility for publishing on PyPI

pexpect-executor