



Linux Ubuntu 26.04 Manual Pages on command 'pip3-index.1'

\$ man pip3-index.1

PIP3-INDEX(1) pip PIP3-INDEX(1)

NAME

pip3-index - description of pip3 index command

DESCRIPTION

Inspect information available from package indexes.

USAGE

python -m pip index versions <package>

OPTIONS

--platform <platform>

Only use wheels compatible with <platform>. Defaults to the platform of the running system. Use this option multiple times to specify multiple platforms supported by the target interpreter.

(environment variable: PIP_PLATFORM)

--python-version <python_version>

The Python interpreter version to use for wheel and "Requires-Python" compatibility checks. Defaults to a version derived from the running interpreter. The version can be specified using up to three dot-separated integers (e.g. "3" for 3.0.0, "3.7" for 3.7.0, or "3.7.3"). A major-minor version can also be given as a string without dots (e.g. "37" for 3.7.0).

(environment variable: PIP_PYTHON_VERSION)

--implementation <implementation>

Only use wheels compatible with Python implementation <implementation>, e.g. 'pp', 'jy', 'cp', or 'ip'. If not specified, then the current interpreter implementation

is used. Use 'py' to force implementation-agnostic wheels.

(environment variable: PIP_IMPLEMENTATION)

`--abi <abi>`

Only use wheels compatible with Python abi <abi>, e.g. 'pypy_41'. If not specified, then the current interpreter abi tag is used. Use this option multiple times to specify multiple abis supported by the target interpreter. Generally you will need to specify --implementation, --platform, and --python-version when using this option.

(environment variable: PIP_ABI)

`--ignore-requires-python`

Ignore the Requires-Python information.

(environment variable: PIP_IGNORE_REQUIRES_PYTHON)

`--pre` Include pre-release and development versions. By default, pip only finds stable versions.

(environment variable: PIP_PRE)

`--json` Output data in a machine-readable JSON format.

(environment variable: PIP_JSON)

`--no-binary <format_control>`

Do not use binary packages. Can be supplied multiple times, and each time adds to the existing value. Accepts either ":all:" to disable all binary packages, ":none:" to empty the set (notice the colons), or one or more package names with commas between them (no colons). Note that some packages are tricky to compile and may fail to install when this option is used on them.

(environment variable: PIP_NO_BINARY)

`--only-binary <format_control>`

Do not use source packages. Can be supplied multiple times, and each time adds to the existing value. Accepts either ":all:" to disable all source packages, ":none:" to empty the set, or one or more package names with commas between them. Packages without binary distributions will fail to install when this option is used on them.

(environment variable: PIP_ONLY_BINARY)

AUTHOR

pip developers

COPYRIGHT

The pip developers

