tox Test automation

CONCEPTS

tox builds a fresh virtualenv per environment, installs your package and deps, then runs the commands. Same steps locally and in CI.

Package build sdist/wheel
↓ `install`
Env virtualenv in .tox/
↓ `deps`
Install deps + package
↓ `commands`
Run exit code = result

Environment names are factors split on dashes: py313-django has factors py313 and django, which deps can key on.

RUNNING

`tox`
Run the envs in env_list

`tox -e` *`<env>`*
Run a single environment

`tox -e` *`<e1>`*`,`*`<e2>`*
Run several named environments

`tox -e` *`<env>`* **`--`** *`<args>`*
Forward args to {posargs}

`tox -e ALL`
Run every defined environment

`tox -r -e` *`<env>`*
Recreate the env from scratch

`tox -f` *`<factor>`*
Run envs containing a factor

`tox -m` *`<label>`*
Run envs under a label

`tox --skip-env` *`<regex>`*
Exclude envs matching a regex

`tox -s`
Skip missing interpreters

`tox --fail-fast`
Stop after the first failing env

`tox -n -e` *`<env>`*
Build the env, skip the commands

`tox -v`
More verbose (repeat for more)

`tox -c` *`<file>`*
Use another config file or folder

PARALLEL

`tox p`
Run envs in parallel (alias of run-parallel)

`tox p -p` *`<n>`*
Limit to n parallel envs

`tox p -o`
Stream output live while running

`tox p -e` *`<e1>`*`,`*`<e2>`*
Parallel run of chosen envs

INSPECT

`tox l`
List envs with descriptions

`tox l -d`
List only default envs

`tox c -e` *`<env>`*
Show the resolved config of an env

`tox c -e` *`<env>`* **`-k`** *`<key>`*
Show only chosen keys

`tox --version`
Show tox and plugin versions

OTHER COMMANDS

`tox quickstart`
Interactively create a config (alias q)

`tox exec -e` *`<env>`* **`--`** *`<cmd>`*
Run any command inside an env

`tox devenv` *`<dir>`*
Create a dev virtualenv from an env

CONFIG FILE

`[tox]`
Core section in tox.ini

`env_list =` *`<e1>`*`,` *`<e2>`*
Default environments

`requires = tox≥4.2`
Needed tox and plugins (auto-provisioned)

`labels =` *`<name>`*`=`*`<e1>`*`,`*`<e2>`*
Name a group of envs

`[testenv]`
Base section for all envs

`[testenv:`*`<env>`***`]`**
Settings for one env

CONFIG FILE (cont.)

`description =` *`<text>`*
Shown by tox l

`deps =` *`<pkg>`*
Packages to install

`commands =` *`<cmd>`* **`{posargs}`**
Commands to run, one per line

`extras =` *`<name>`*
Install optional dependencies

`pass_env =` *`<VAR>`*
Let env vars through to the env

`set_env =` *`<VAR>`*`=`*`<val>`*
Set env vars

`allowlist_externals =` *`<cmd>`*
Allow commands from outside the env

`base_python = python3.13`
Pick the interpreter

`skip_install = true`
Do not install your package

`package = wheel`
sdist, wheel, editable or skip

`dependency_groups =` *`<group>`*
Install PEP 735 groups (tox 4.22+)

SUBSTITUTION & FACTORS

`{posargs}`
Args after -- on the command line

`{posargs:`*`<default>`***`}`**
Args, or a default if none given

*`<factor>`***`:`** *`<value>`*
Line applies only if the env has the factor

`py3{12,13}-{a,b}`
Expand into every combination

RECIPES

Minimal tox.ini

```
[tox]
env_list = lint, py313

[testenv]
deps = pytest
commands = pytest {posargs}

[testenv:lint]
skip_install = true
deps = ruff
commands = ruff check .
```