# tox
# (not poison)

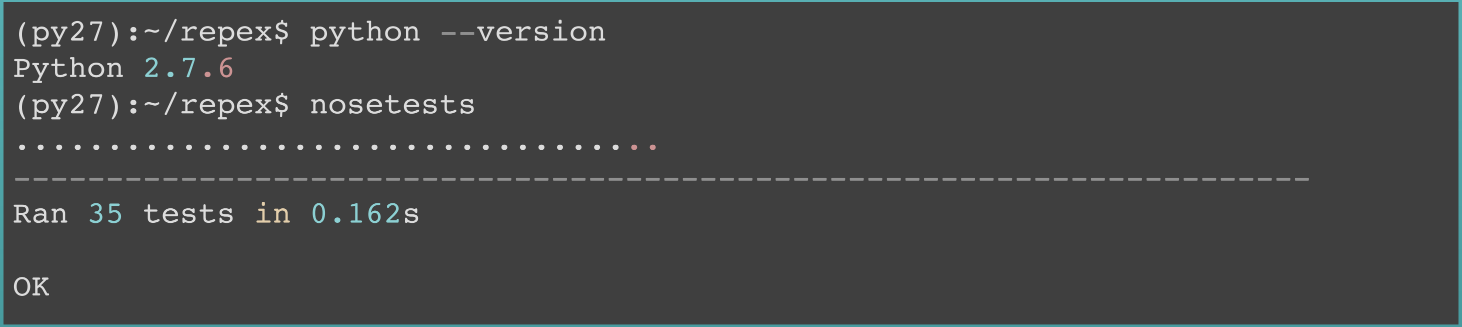
Nir Cohen@Gigaspaces
nir0s@github
@thinkops

# Agenda

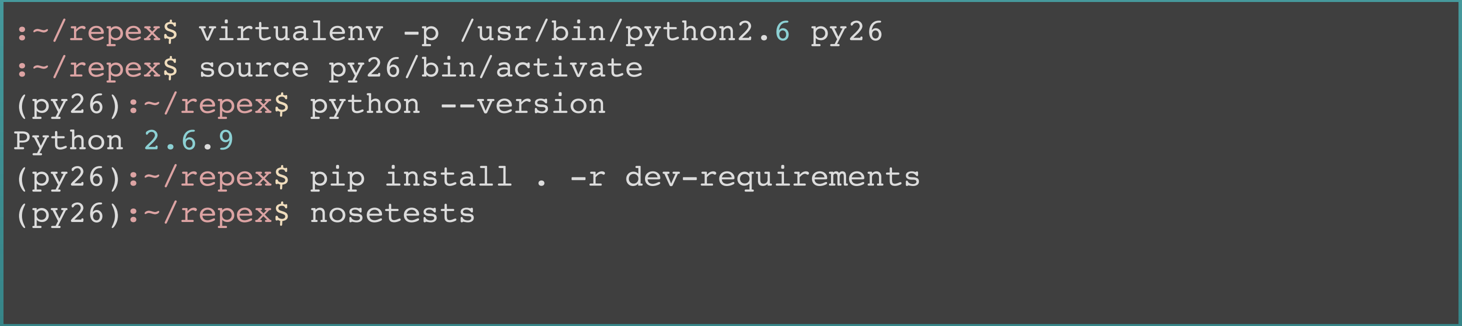
- current problems with testing
- tox!
- Use cases
- Integrations and parallelization
- Pitfalls

# Current testing problems

# Testing on multiple versions of Python.

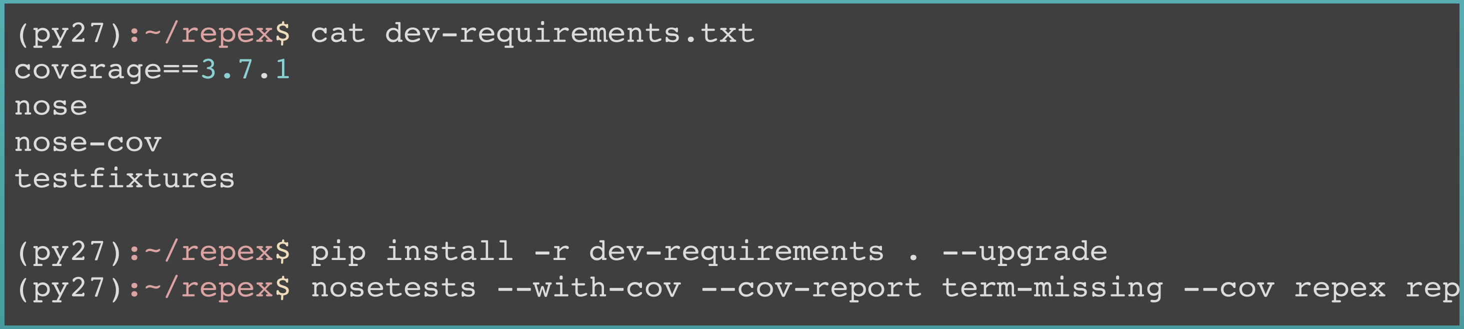


```
(py27):~/repex$ python --version
Python 2.7.6
(py27):~/repex$ nosetests
...................................
----------------------------------------------------------------------
Ran 35 tests in 0.162s

OK
```

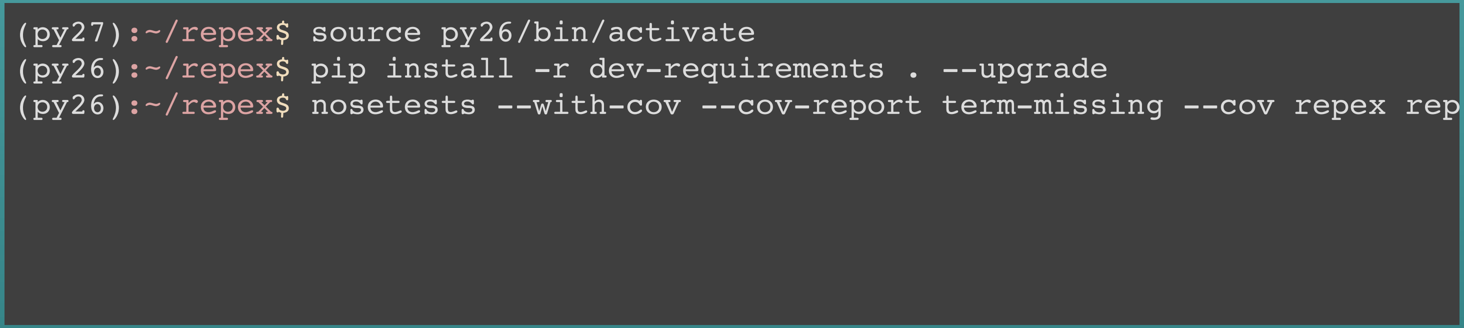


```
:~/repex$ virtualenv -p /usr/bin/python2.6 py26
:~/repex$ source py26/bin/activate
(py26):~/repex$ python --version
Python 2.6.9
(py26):~/repex$ pip install . -r dev-requirements
(py26):~/repex$ nosetests
```

# Managing test env dependencies.



```
(py27):~/repex$ cat dev-requirements.txt
coverage==3.7.1
nose
nose-cov
testfixtures

(py27):~/repex$ pip install -r dev-requirements . --upgrade
(py27):~/repex$ nosetests --with-cov --cov-report term-missing --cov repex rep
```

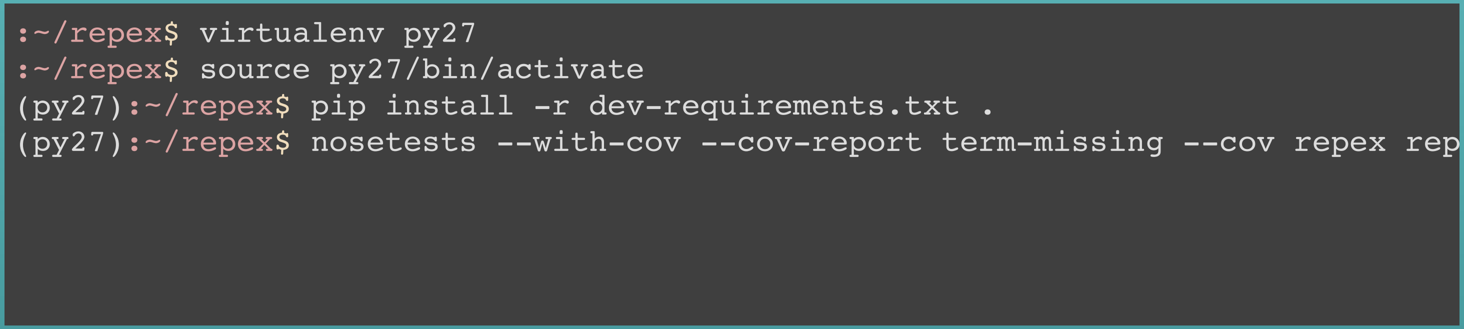


```
(py27):~/repex$ source py26/bin/activate
(py26):~/repex$ pip install -r dev-requirements . --upgrade
(py26):~/repex$ nosetests --with-cov --cov-report term-missing --cov repex rep
```

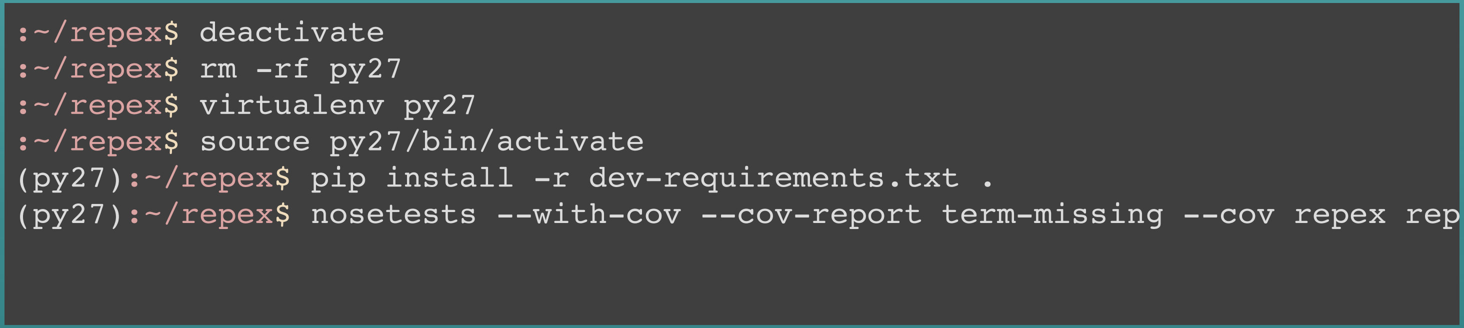
# Executing pre and post test related activities.

```
(py27):~/repex$ mypyserv --port 5000 &
(py27):~/repex$ nosetests --with-cov --cov-report term-missing \
    --cov repex repex/tests -v
```

# Running tests in clean environments.

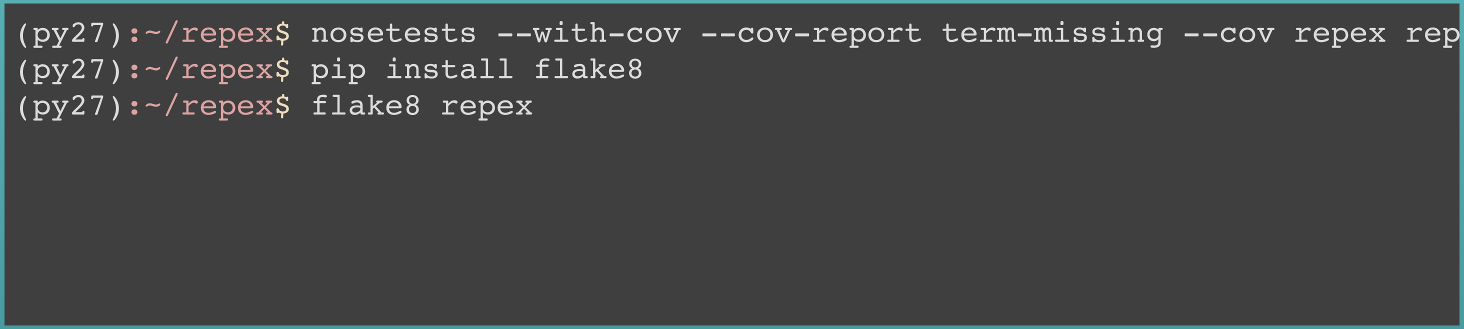


```
:~/repex$ virtualenv py27
:~/repex$ source py27/bin/activate
(py27):~/repex$ pip install -r dev-requirements.txt .
(py27):~/repex$ nosetests --with-cov --cov-report term-missing --cov repex rep
```

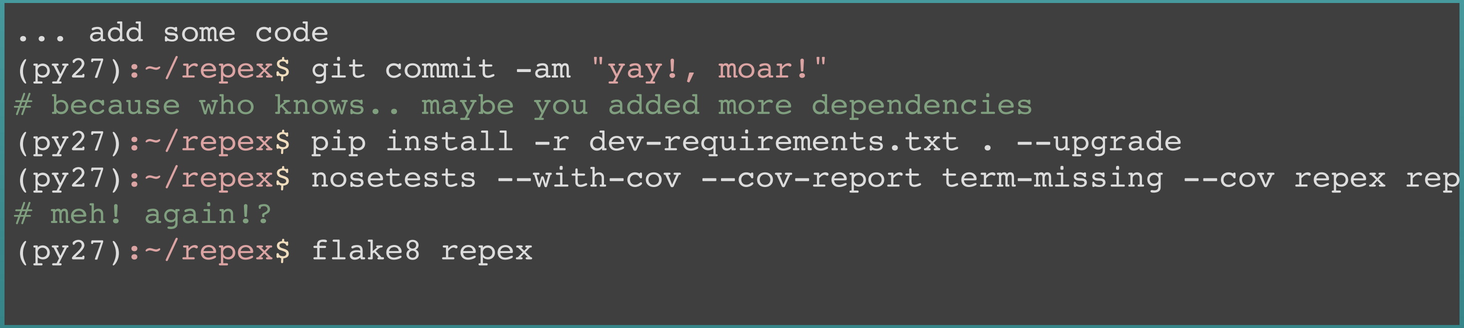


```
:~/repex$ deactivate
:~/repex$ rm -rf py27
:~/repex$ virtualenv py27
:~/repex$ source py27/bin/activate
(py27):~/repex$ pip install -r dev-requirements.txt .
(py27):~/repex$ nosetests --with-cov --cov-report term-missing --cov repex rep
```

# Flake8 should be run on every commit - annoying.



```
(py27):~/repex$ nosetests --with-cov --cov-report term-missing --cov repex rep
(py27):~/repex$ pip install flake8
(py27):~/repex$ flake8 repex
```



```
... add some code
(py27):~/repex$ git commit -am "yay!, moar!"
# because who knows.. maybe you added more dependencies
(py27):~/repex$ pip install -r dev-requirements.txt . --upgrade
(py27):~/repex$ nosetests --with-cov --cov-report term-missing --cov repex rep
# meh! again!?
(py27):~/repex$ flake8 repex
```

- Complex and repetitive execution of tests.
- Most of the above must be solved for every testing platform (py.test, nose, unittest, etc..)
- Executing tests in parallel.

# Let's (de)tox

# tox...

- manages virtualenvs
- very configurable
- supports most common interpreters (from py24 to py34, jython and pypy).
- more. let's see, shall we?

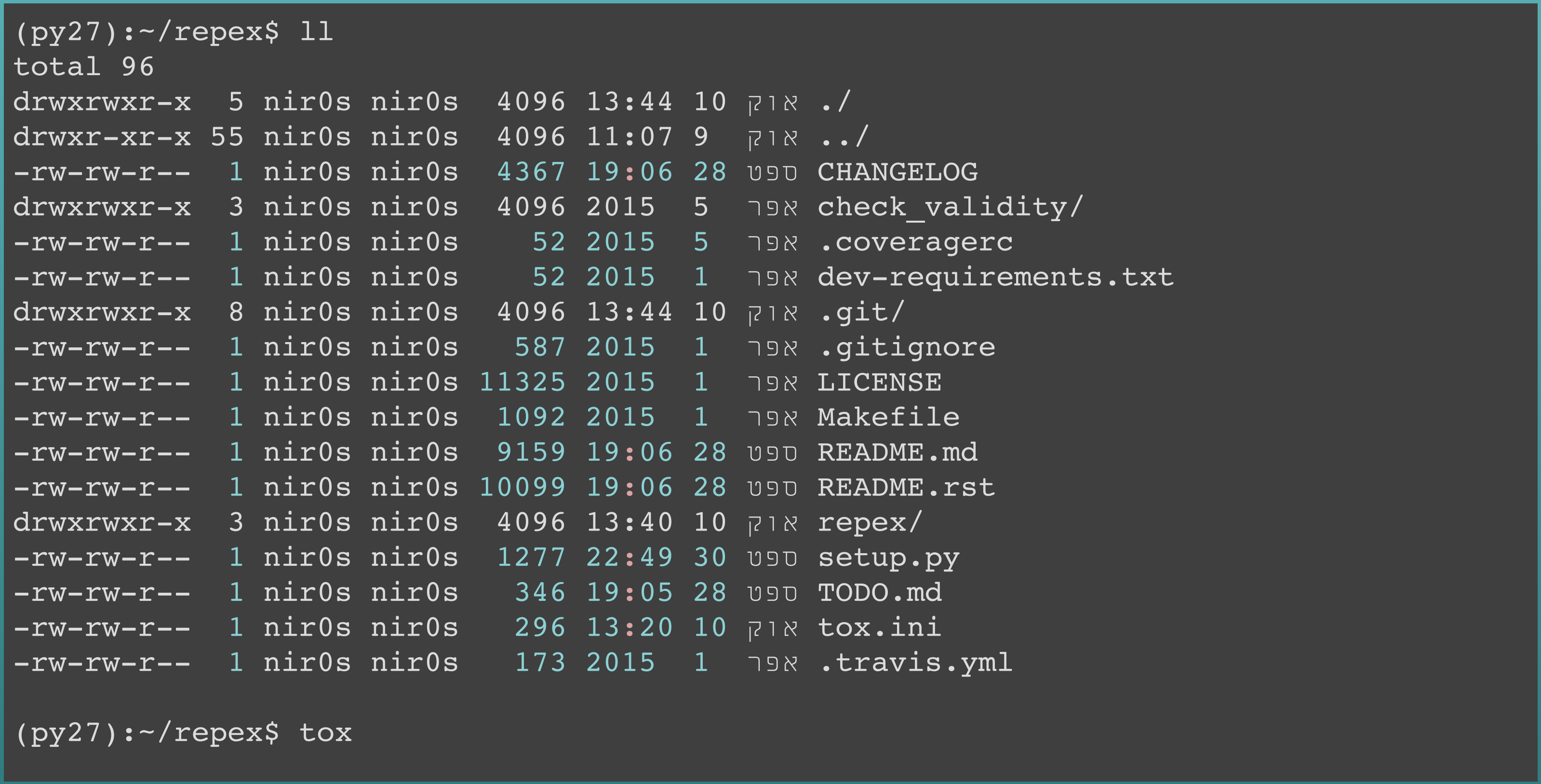
# install tox

```
pip install tox
```

# tox.ini

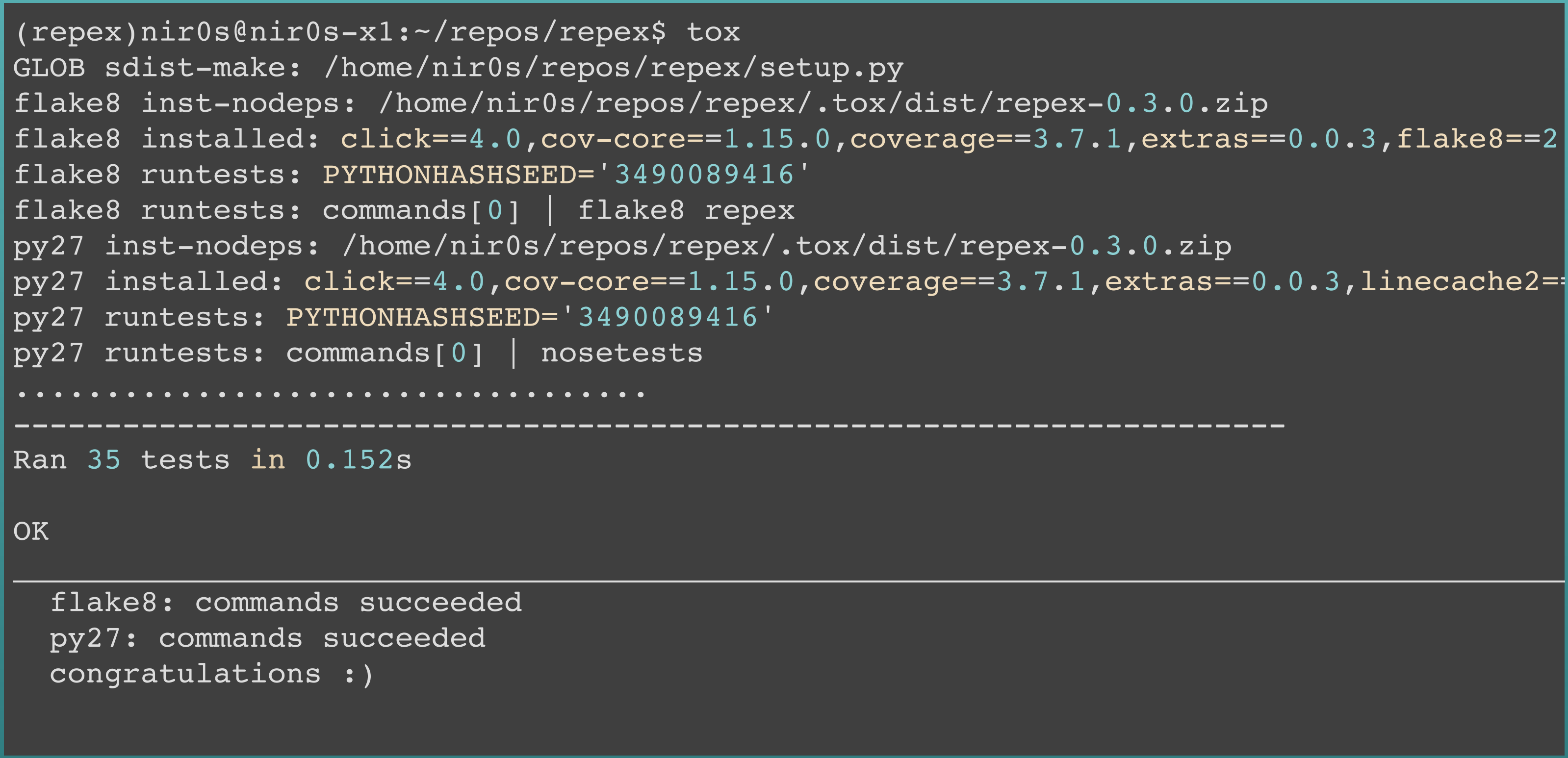
```ini
[tox]
# list to environments to test against. Either interpreters or arbitrary names.
envlist=flake8,py27

# env for all interpreters
[testenv]
# install dependencies
deps =
    -rdev-requirements.txt
# execute commands within the virtualenv
commands=nosetests --with-cov --cov-report term-missing --cov repex repex/tests -v

# runs on default interpreter. name for specific configuration.
[testenv:flake8]
deps =
    flake8
    -rdev-requirements.txt
commands=flake8 repex
```

# toxin'



```
(py27):~/repex$ ll
total 96
drwxrwxr-x  5 nir0s nir0s  4096 13:44 10 אוק ./
drwxr-xr-x 55 nir0s nir0s  4096 11:07 9  אוק ../
-rw-rw-r--  1 nir0s nir0s  4367 19:06 28 ספט CHANGELOG
drwxrwxr-x  3 nir0s nir0s  4096 2015  5  אפר check_validity/
-rw-rw-r--  1 nir0s nir0s    52 2015  5  אפר .coveragerc
-rw-rw-r--  1 nir0s nir0s    52 2015  1  אפר dev-requirements.txt
drwxrwxr-x  8 nir0s nir0s  4096 13:44 10 אוק .git/
-rw-rw-r--  1 nir0s nir0s   587 2015  1  אפר .gitignore
-rw-rw-r--  1 nir0s nir0s 11325 2015  1  אפר LICENSE
-rw-rw-r--  1 nir0s nir0s  1092 2015  1  אפר Makefile
-rw-rw-r--  1 nir0s nir0s  9159 19:06 28 ספט README.md
-rw-rw-r--  1 nir0s nir0s 10099 19:06 28 ספט README.rst
drwxrwxr-x  3 nir0s nir0s  4096 13:40 10 אוק repex/
-rw-rw-r--  1 nir0s nir0s  1277 22:49 30 ספט setup.py
-rw-rw-r--  1 nir0s nir0s   346 19:05 28 ספט TODO.md
-rw-rw-r--  1 nir0s nir0s   296 13:20 10 אוק tox.ini
-rw-rw-r--  1 nir0s nir0s   173 2015  1  אפר .travis.yml

(py27):~/repex$ tox
```

# results



```
(repex)nir0s@nir0s-x1:~/repos/repex$ tox
GLOB sdist-make: /home/nir0s/repos/repex/setup.py
flake8 inst-nodeps: /home/nir0s/repos/repex/.tox/dist/repex-0.3.0.zip
flake8 installed: click==4.0,cov-core==1.15.0,coverage==3.7.1,extras==0.0.3,flake8==2
flake8 runtests: PYTHONHASHSEED='3490089416'
flake8 runtests: commands[0] | flake8 repex
py27 inst-nodeps: /home/nir0s/repos/repex/.tox/dist/repex-0.3.0.zip
py27 installed: click==4.0,cov-core==1.15.0,coverage==3.7.1,extras==0.0.3,linecache2==
py27 runtests: PYTHONHASHSEED='3490089416'
py27 runtests: commands[0] | nosetests
...................................
----------------------------------------------------------------------
Ran 35 tests in 0.152s

OK
_____________________________________________________________________________________
  flake8: commands succeeded
  py27: commands succeeded
  congratulations :)
```

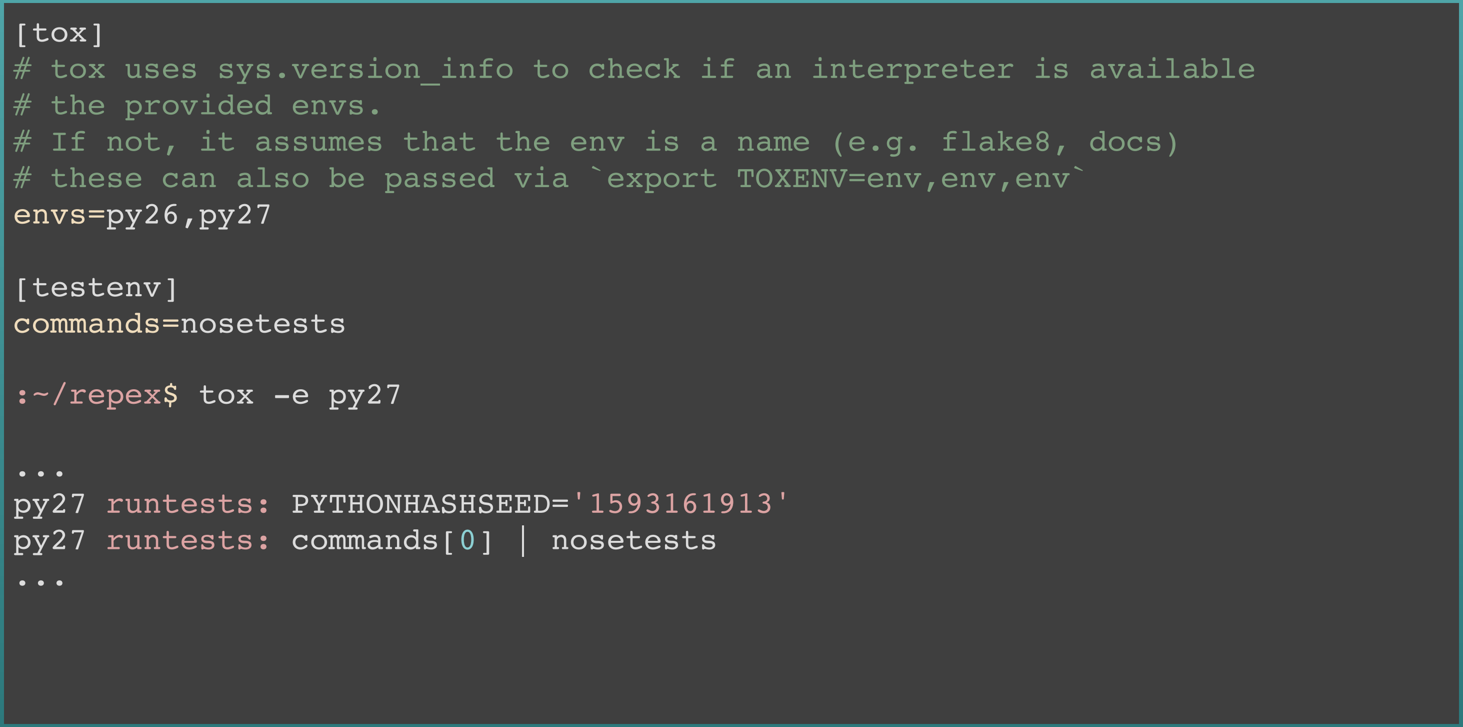
# dev

- Elaborate docs at: http://goo.gl/utWvRp
- Constantly developed: https://goo.gl/acS10a

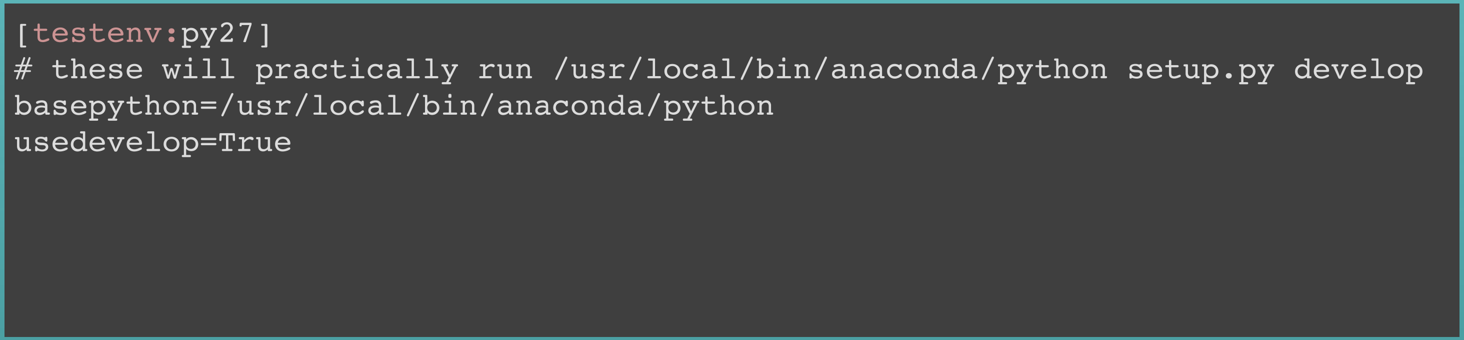
# Use Cases

# you specified multiple Python versions to run on

# but only want to run on one of them now.



```
[tox]
# tox uses sys.version_info to check if an interpreter is available
# the provided envs.
# If not, it assumes that the env is a name (e.g. flake8, docs)
# these can also be passed via `export TOXENV=env,env,env`
envs=py26,py27

[testenv]
commands=nosetests

:~/repex$ tox -e py27

...
py27 runtests: PYTHONHASHSEED='1593161913'
py27 runtests: commands[0] | nosetests
...
```



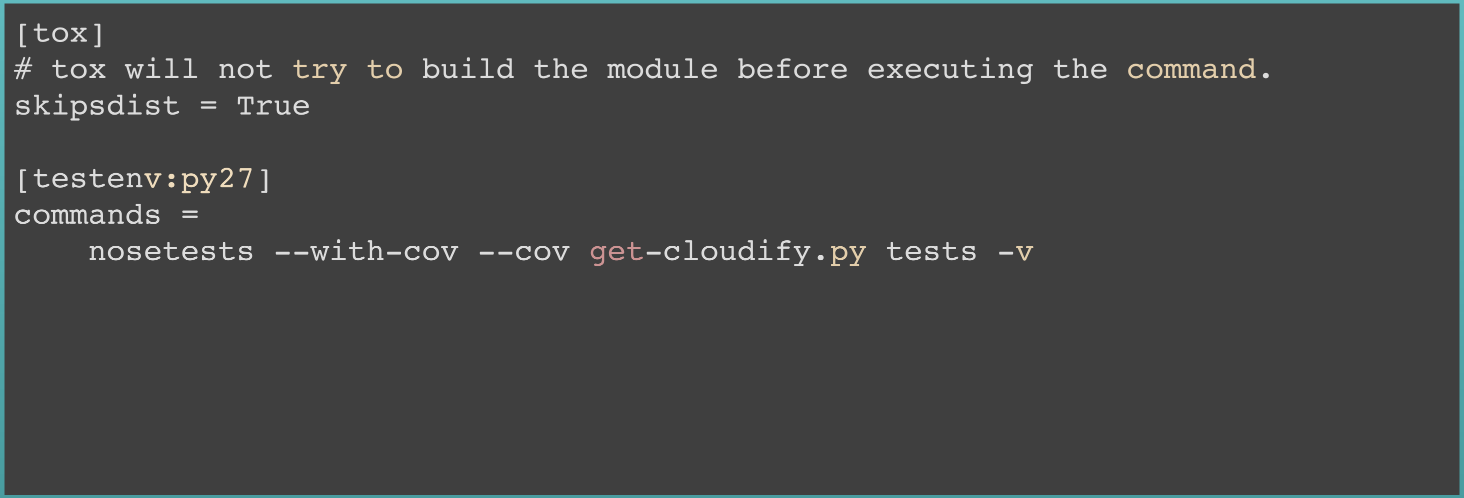
```
[testenv:py27]
# these will practically run /usr/local/bin/anaconda/python setup.py develop
basepython=/usr/local/bin/anaconda/python
usedevelop=True
```

Your default Python interpreter is not the one required for

that specific test. You also want to install your module using ` develop ` you know that your code will not work on Python 2.6

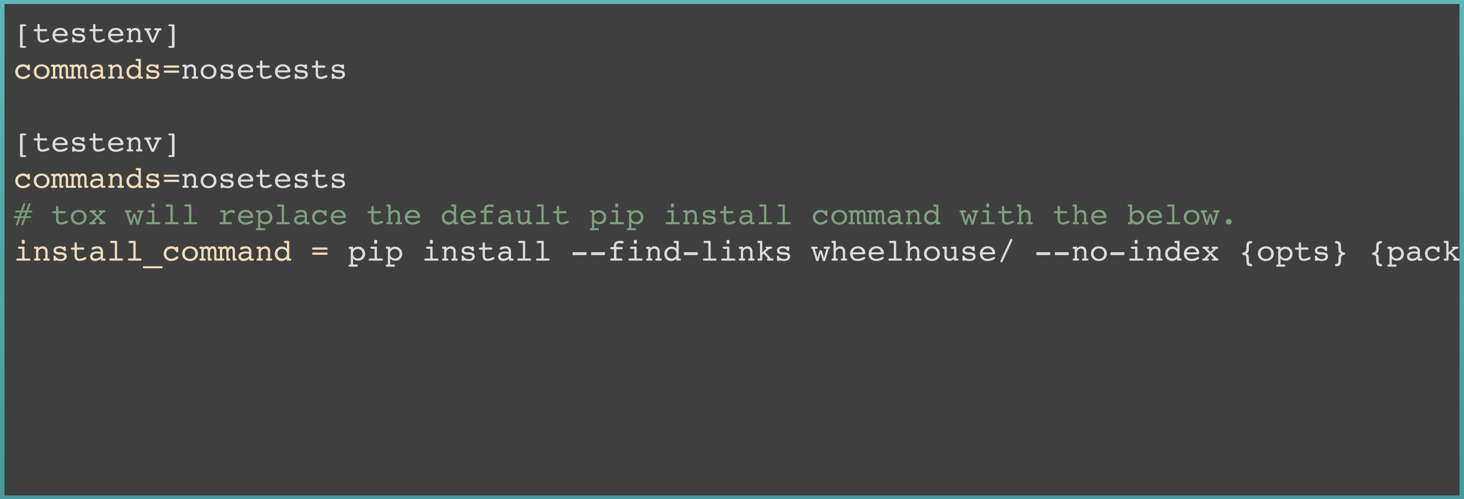
on OS X (for some reason) and so you want to skip it.

```
[testenv]
commands=nosetests

[testenv:py26]
commands=nosetests
# tox uses sys.platform to check against the provided regex entry.
platform=linux2|win32
```



```
[tox]
# tox will not try to build the module before executing the command.
skipsdist = True

[testenv:py27]
commands =
    nosetests --with-cov --cov get-cloudify.py tests -v
```

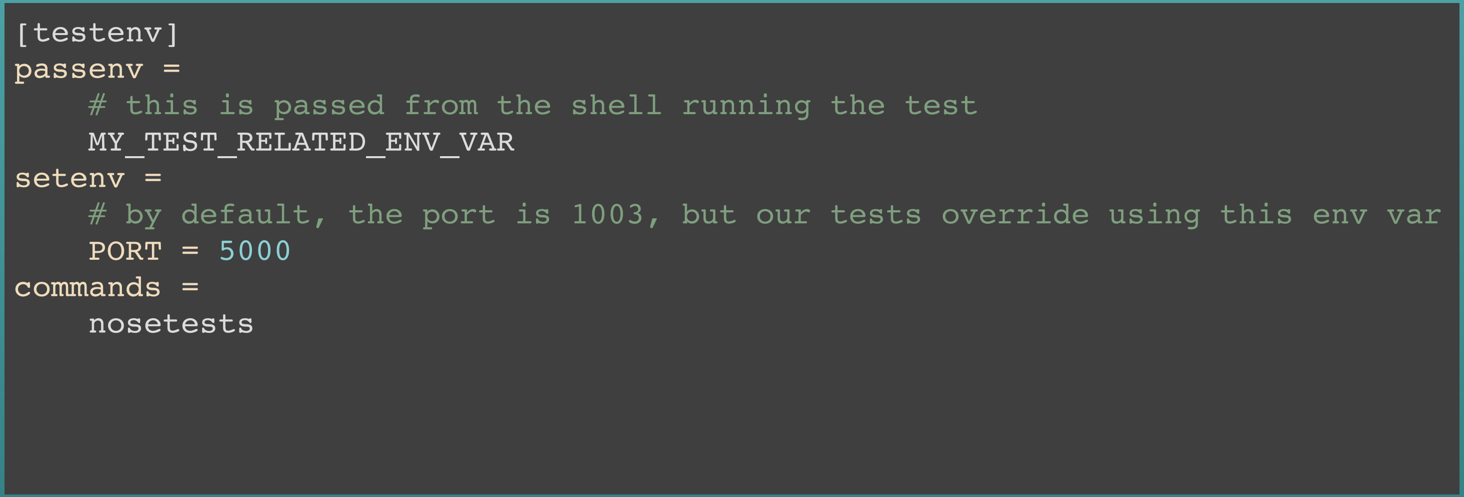
you have a Python script
you want to run tests for,
but it's not a module.



```
[testenv]
commands=nosetests

[testenv]
commands=nosetests
# tox will replace the default pip install command with the below.
install_command = pip install --find-links wheelhouse/ --no-index {opts} {pack
```

you want to install your dependencies from a set a wheels in a wheelhouse dir.

You want to pass down a specific env var from your shell to the test env and create a new env var for your tests.

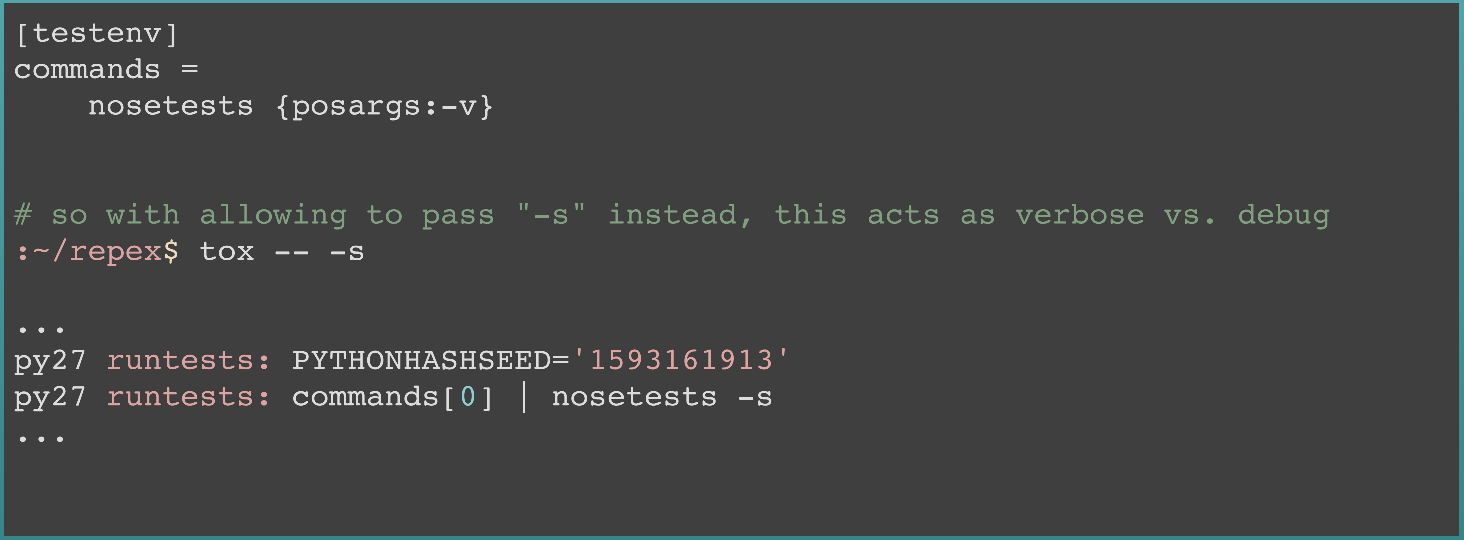


```
[testenv]
passenv =
    # this is passed from the shell running the test
    MY_TEST_RELATED_ENV_VAR
setenv =
    # by default, the port is 1003, but our tests override using this env var
    PORT = 5000
commands =
    nosetests
```

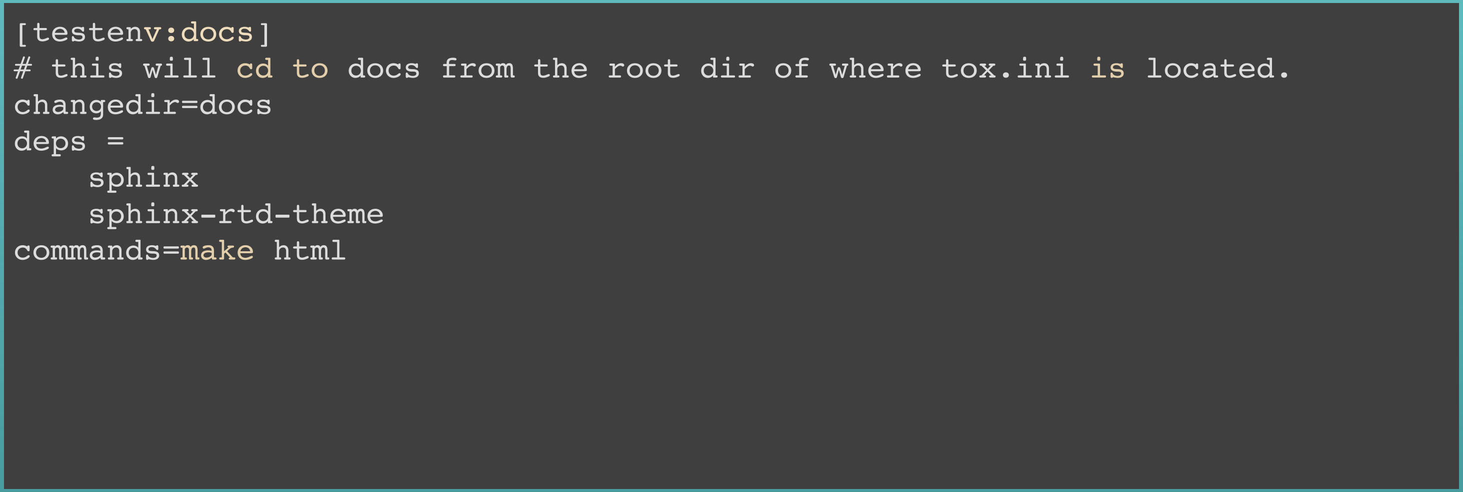
# you want to deploy your code but don't care about the exit code.

```
[testenv]
commands =
    nosetests
    flake8
    - deploy_carelessly
```

# you want to create a --verbose vs. --debug mode for executing your tests.



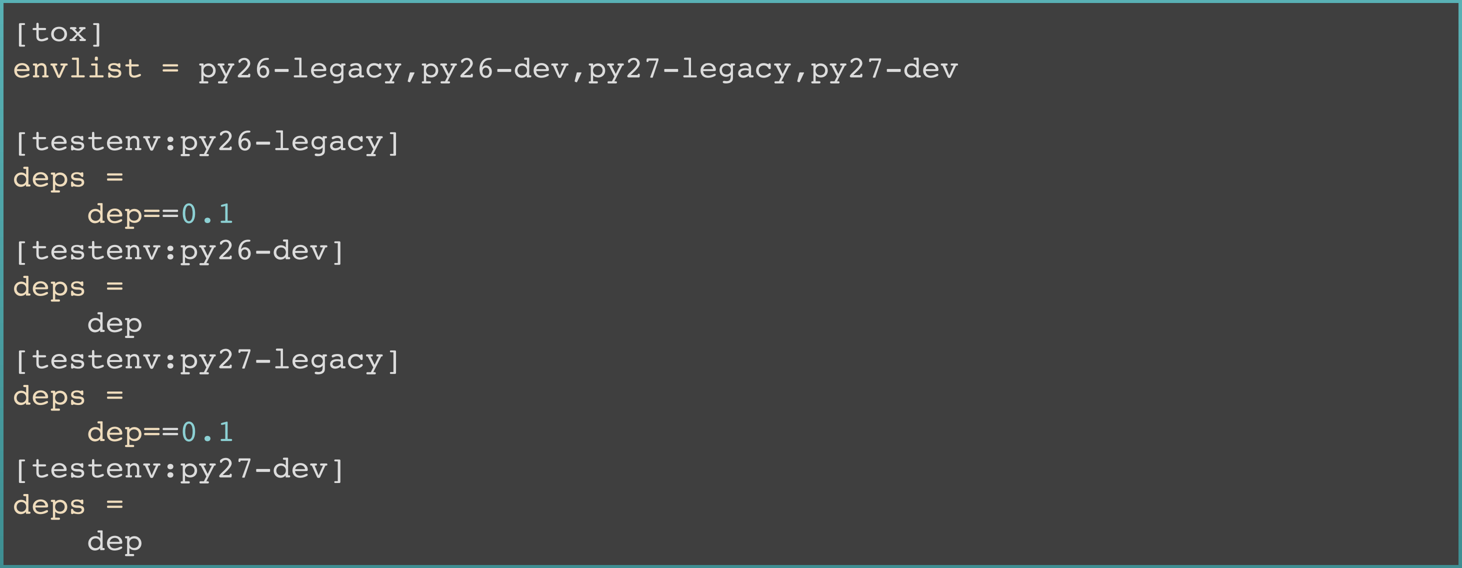
```
[testenv]
commands =
    nosetests {posargs:-v}


# so with allowing to pass "-s" instead, this acts as verbose vs. debug
:~/repex$ tox -- -s

...
py27 runtests: PYTHONHASHSEED='1593161913'
py27 runtests: commands[0] | nosetests -s
...
```



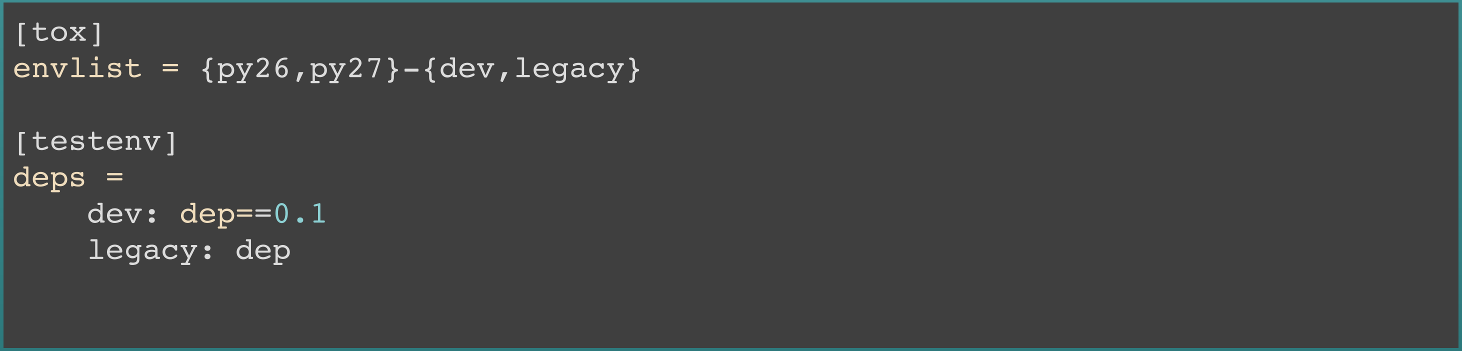
```
[testenv:docs]
# this will cd to docs from the root dir of where tox.ini is located.
changedir=docs
deps =
    sphinx
    sphinx-rtd-theme
commands=make html
```

you based your API docs on sphinx and want to generate them after every commit to make sure you didn't break anything in the docstrings.

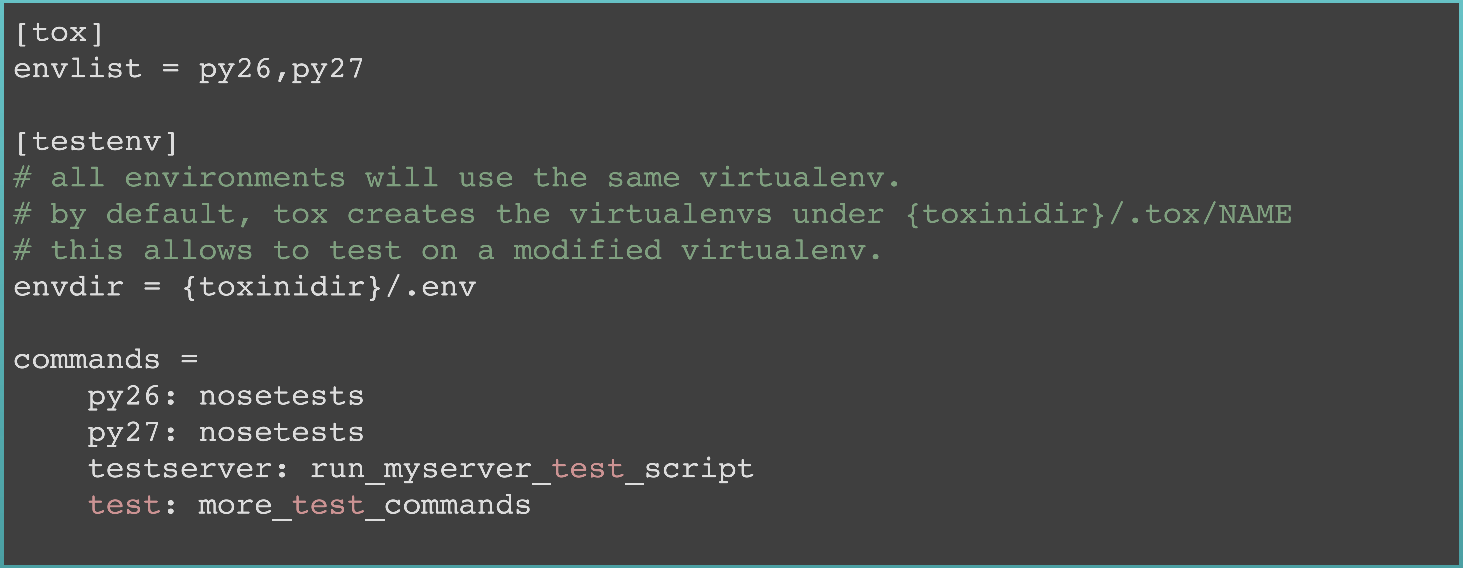
# you want to test against multiple versions of the same dependency



```ini
[tox]
envlist = py26-legacy,py26-dev,py27-legacy,py27-dev

[testenv:py26-legacy]
deps =
    dep==0.1
[testenv:py26-dev]
deps =
    dep
[testenv:py27-legacy]
deps =
    dep==0.1
[testenv:py27-dev]
deps =
    dep
```



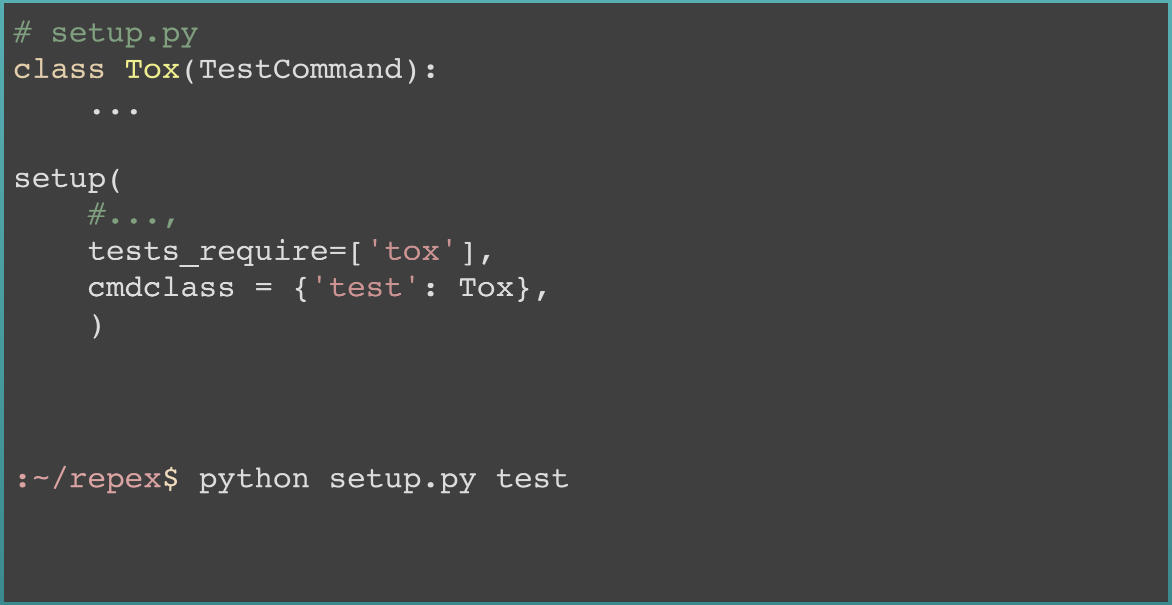
```ini
[tox]
envlist = {py26,py27}-{dev,legacy}

[testenv]
deps =
    dev: dep==0.1
    legacy: dep
```



```
[tox]
envlist = py26,py27

[testenv]
# all environments will use the same virtualenv.
# by default, tox creates the virtualenvs under {toxinidir}/.tox/NAME
# this allows to test on a modified virtualenv.
envdir = {toxinidir}/.env

commands =
    py26: nosetests
    py27: nosetests
    testserver: run_myserver_test_script
    test: more_test_commands
```

you want to use the same set of dependencies or changes to a virtualenv in all test environments.

# setup.py integration



```
# setup.py
class Tox(TestCommand):
    ...

setup(
    #...,
    tests_require=['tox'],
    cmdclass = {'test': Tox},
    )
```

```
:~/repex$ python setup.py test
```

# detox

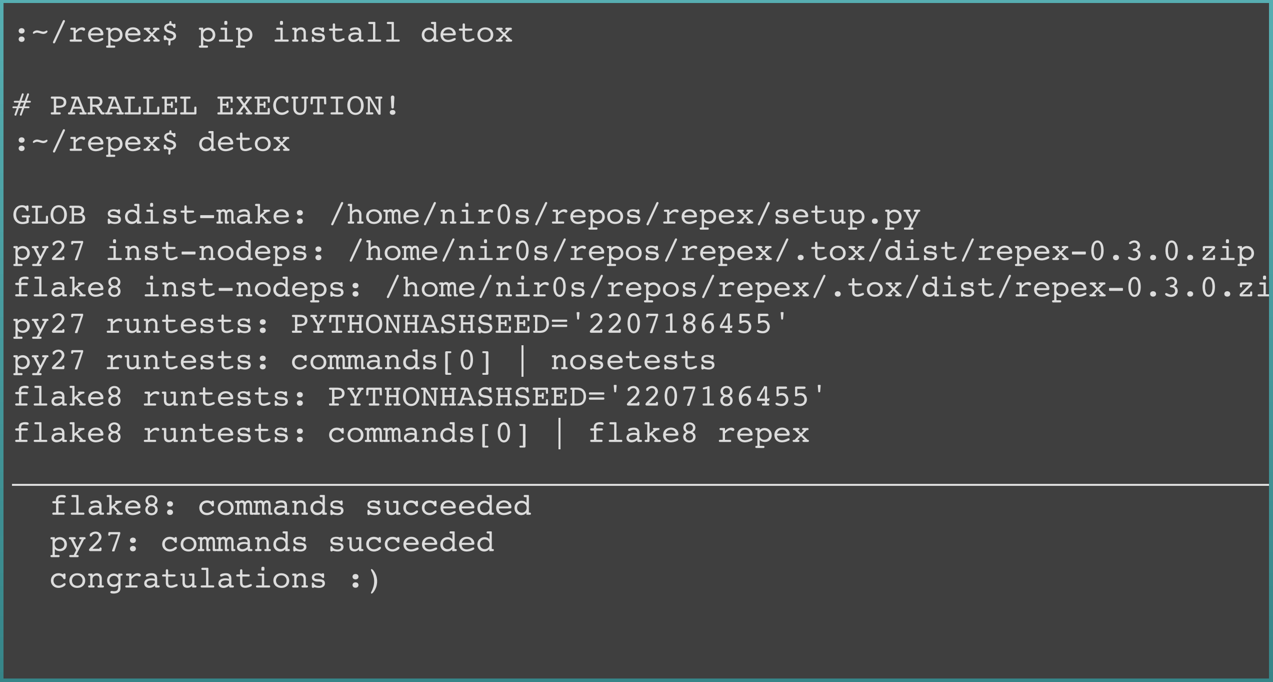
## local parallelization

# install detox

```
pip install detox
```

same tox.ini file!

# parallel fun!



```
:~/repex$ pip install detox

# PARALLEL EXECUTION!
:~/repex$ detox

GLOB sdist-make: /home/nir0s/repos/repex/setup.py
py27 inst-nodeps: /home/nir0s/repos/repex/.tox/dist/repex-0.3.0.zip
flake8 inst-nodeps: /home/nir0s/repos/repex/.tox/dist/repex-0.3.0.zi
py27 runtests: PYTHONHASHSEED='2207186455'
py27 runtests: commands[0] | nosetests
flake8 runtests: PYTHONHASHSEED='2207186455'
flake8 runtests: commands[0] | flake8 repex
___________________________________________________________________
  flake8: commands succeeded
  py27: commands succeeded
  congratulations :)
```

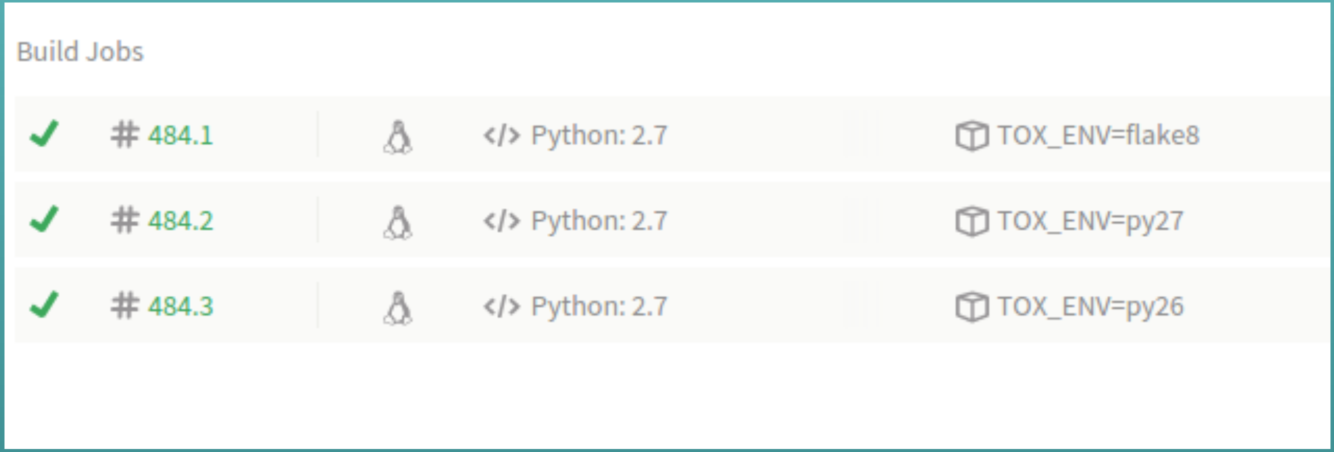
# tox and Travis

## remote parallelization

# .travis.yml

```yaml
sudo: false
language: python
python:
  - "2.7"
env:
    - TOX_ENV=flake8
    - TOX_ENV=py27
    - TOX_ENV=py26
install:
    - pip install tox
script:
    - tox -e $TOX_ENV
```

# parallel fun!



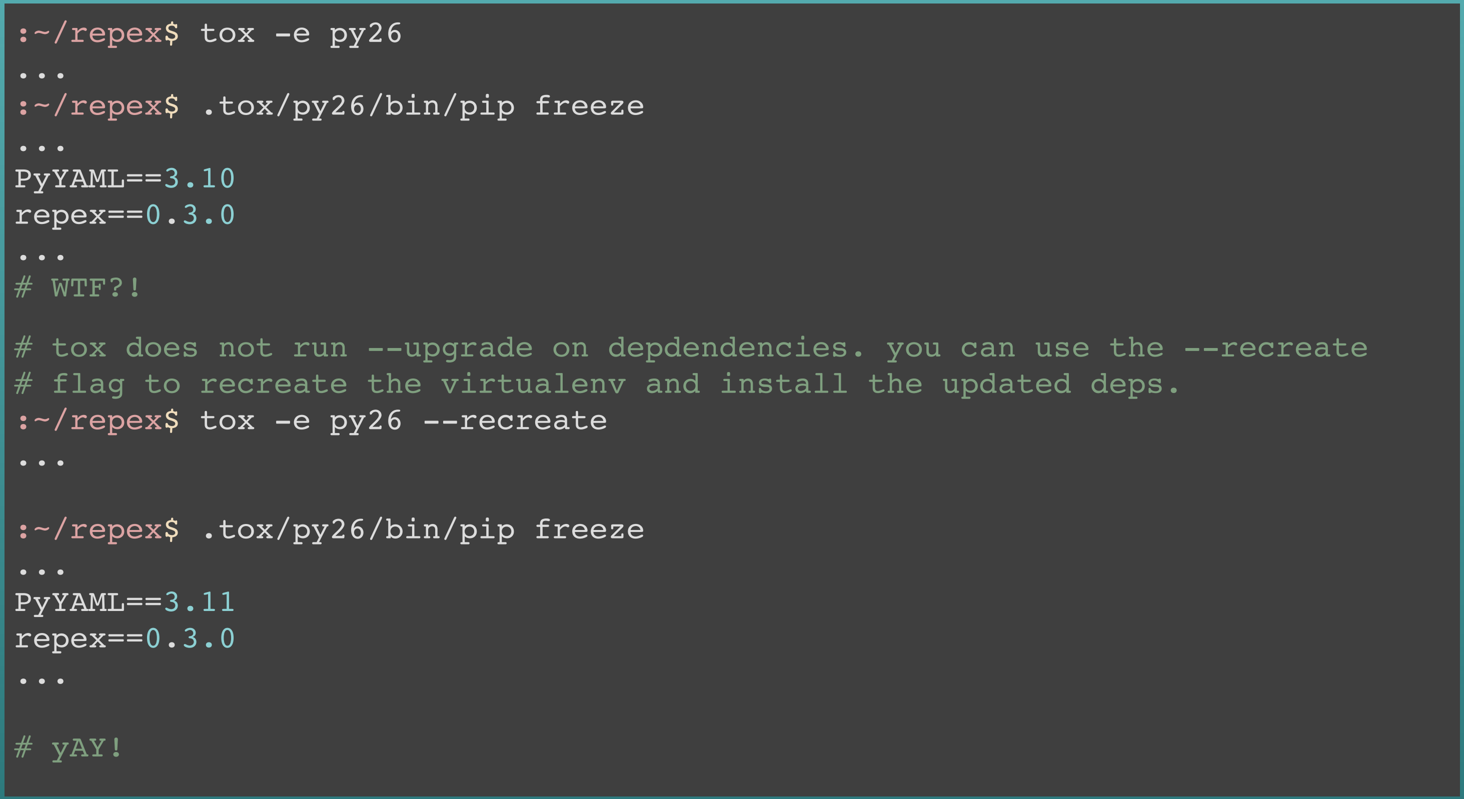
Build Jobs

| | | | |
|---|---|---|---|
| ✓ | # 484.1 | </> Python: 2.7 | TOX_ENV=flake8 |
| ✓ | # 484.2 | </> Python: 2.7 | TOX_ENV=py27 |
| ✓ | # 484.3 | </> Python: 2.7 | TOX_ENV=py26 |

# Pitalls

getting intoxicated

you've been running tox for a while and decided to upgrade pyyaml from 3.10 to 3.11 and run tox again.



```
:~/repex$ tox -e py26
...
:~/repex$ .tox/py26/bin/pip freeze
...
PyYAML==3.10
repex==0.3.0
...
# WTF?!

# tox does not run --upgrade on depdendencies. you can use the --recreate
# flag to recreate the virtualenv and install the updated deps.
:~/repex$ tox -e py26 --recreate
...

:~/repex$ .tox/py26/bin/pip freeze
...
PyYAML==3.11
repex==0.3.0
...

# yAY!
```

detox requires test environments to be independent or clean.

# Useful Links

- https://tox.readthedocs.org/en/latest/
- https://github.com/dstanek/tox-run-command
- https://github.com/cloudify-cosmo/repex/blob/master/tox.ini
- https://github.com/mitsuhiko/flask/blob/master/tox.ini
- https://www.youtube.com/watch?v=Oldkj519o4A
- https://www.youtube.com/watch?v=s_YjaODzM1E