zc.buildout pagrindai

Albertas Agejevas <alga@pov.lt>
POV

Pycon-LT 2009

Cheeseshop

Eggai guli pypi:
`http://pypi.python.org/`

Easy!

To try our package, just type:

```
easy_install mycoolthing
```

(Not so) easy...

```
$ easy_install mycoolthing
```

```
error: can't create or remove files in install directory

The following error occurred while trying to add or remove files in the
installation directory:
    [Errno 13] Permission denied: '/usr/lib/python2.5/site-packages/test-easy-i
The installation directory you specified (via --install-dir, --prefix, or
the distutils default setting) was:
    /usr/lib/python2.5/site-packages/
```

sudo easy_install – ne sprendimas

- nešvaru, distribucijos purvinimas
- anksčiau ar vėliau atsiras versijų konfliktas

zc.buildout to the rescue!

```
$ vi buildout.cfg

[buildout]
parts = ZODB3

[ZODB3]
recipe = zc.recipe.egg
interpreter = python
```

Leidžiam

```
$ wget
http://ignas.pov.lt/bootstrap.py
```

(arba http://svn.zope.org/*checkout*/zc.buildout/trunk/bootstrap/bootstrap.py)

```
$ python2.4 bootstrap.py
```

```
Downloading

http://pypi.python.org/packages/2.4/s/setuptools/setuptools-0.6c9-py2.4.egg
```

Leidžiam toliau...

```
$ bin/buildout
```

```
Installing ZODB3.
Getting distribution for 'ZODB3'.
Got ZODB3 3.9.0a12.
Getting distribution for 'zope.testing'.
Got zope.testing 3.7.3.
Getting distribution for 'zope.proxy'.
Got zope.proxy 3.5.0.
Getting distribution for 'zope.interface'.
Got zope.interface 3.5.1.
Getting distribution for 'zope.event'.
Got zope.event 3.4.1.
Getting distribution for 'zdaemon'.
Got zdaemon 2.0.4.
Getting distribution for 'ZConfig'.
Got ZConfig 2.6.1.
Getting distribution for 'zc.lockfile'.
Got zc.lockfile 1.0.0.
Getting distribution for 'transaction'.
Got transaction 1.0a1.
Generated script '/home/alga/docs/pycon-lt/b/bin/mkzeoinst'.
```

Galima žaist!

```
$ bin/python
>>> import ZODB
>>>
```

Pažaiskim

```
#!/usr/bin/env python
import sys
import re
import ZODB
import ZODB.FileStorage
from persistent.list import PersistentList
import transaction


def initialise():
    storage = ZODB.FileStorage.FileStorage(’Data.fs’)
    db = ZODB.DB(storage)
    conn = db.open()

    root = conn.root()
    if ’todo’ not in root:
        root[’todo’] = PersistentList()
    if ’done’ not in root:
        root[’done’] = PersistentList()
    return root
```

```
def main():
    db = initialise()
    if len(sys.argv) == 1:
        for item in db[’todo’]:
            print item
    elif sys.argv[1] == ’add’:
        item = " ".join(sys.argv[2:])
        db[’todo’].append(item)
        print ’TODO:’, item
        transaction.commit()
    elif sys.argv[1] == ’done’:
        pat = re.compile(sys.argv[2])
        for item in db[’todo’][:]:
            if pat.search(item):
                db[’todo’].remove(item)
                db[’done’].append(item)
                print ’DONE:’, item
        transaction.commit()
    else:
        usage()

if __name__ == ’__main__’:
    main()
```

```
def main():
    db = initialise()
    if len(sys.argv) == 1:
        for item in db[’todo’]:
            print item
    elif sys.argv[1] == ’add’:
        item = " ".join(sys.argv[2:])
        db[’todo’].append(item)
        print ’TODO:’, item
        transaction.commit()
    elif sys.argv[1] == ’done’:
        pat = re.compile(sys.argv[2])
        for item in db[’todo’][:]:
            if pat.search(item):
                db[’todo’].remove(item)
                db[’done’].append(item)
                print ’DONE:’, item
        transaction.commit()
    else:
        usage()

if __name__ == ’__main__’:
    main()
```

Galim leisti

```
$ bin/python src/todo.py
```

Na, tikram gyvenime viskas kiek sudėtingiau...

$ vi setup.py

```
from setuptools import setup

setup(
    name="todo",
    # author, description, etc...
    package_dir={’’: ’src’},
    install_requires=[’ZODB3’],
    entry_points="""
    [console_scripts]
    todo = todo:main
    """)
```

```
$ vi buildout.cfg

[buildout]
parts = todo
develop = .

[todo]
recipe = zc.recipe.egg
interpreter = python
```

```
$ bin/buildout
```

```
Develop: '/home/alga/src/todo/.'
Uninstalling ZODB3.
Installing todo.
Generated script '/home/alga/src/todo/bin/todo
```

$ cat bin/todo

```
#!/home/alga/src/todo/python/bin/python

import sys
sys.path[0:0] = [
  ’/home/alga/src/todo/src’,
  ’/home/alga/src/todo/eggs/ZODB3-3.9.0a12-py2.4-linux-i686.egg’,
  ’/home/alga/src/todo/eggs/zope.testing-3.7.3-py2.4.egg’,
  ’/home/alga/src/todo/eggs/zope.proxy-3.5.0-py2.4-linux-i686.egg’,
  ’/home/alga/src/todo/eggs/zope.interface-3.5.1-py2.4-linux-i686.egg’,
  ’/home/alga/src/todo/eggs/zope.event-3.4.1-py2.4.egg’,
  ’/home/alga/src/todo/eggs/zdaemon-2.0.4-py2.4.egg’,
  ’/home/alga/src/todo/eggs/ZConfig-2.6.1-py2.4.egg’,
  ’/home/alga/src/todo/eggs/zc.lockfile-1.0.0-py2.4.egg’,
  ’/home/alga/src/todo/eggs/transaction-1.0a1-py2.4.egg’,
  ’/home/alga/src/todo/eggs/setuptools-0.6c9-py2.4.egg’,
  ]

import todo

if __name__ == ’__main__’:
    todo.main()
```

Palyginimas su alternatyva

virtualenv Keistas centralizuotas sprendimas, jį patį reik easy_installint. Paskui junginėtis tarp aplinkų.

zc.buildout Bootstrapas lygioj vietoj, kiekvienai aplikacijai sava aplinka

Trūkumai

- Dokumentacija išmėtyta po skirtingus eggus
- Veikia gana ilgai
- Jo paties kodas sudėtingas

Ačiū

Klausimai?