

单元测试示例

这份demo代码用来展示结合Tox+Pytest的单元测试，以下文档会对单元测试构建过程进行简单介绍。以下是demo项目的代码结构

```
neil@Neil:unit_test$ tree
.
├── conf
│   ├── config.py
│   └── __init__.py
├── misc
│   ├── coverage.ini
│   ├── requirements.txt
│   └── tox.ini
├── scripts
│   └── run_tests.sh
├── code
│   ├── __init__.py
│   ├── __main__.py
│   └── models.py
└── tests
    ├── unit
    │   ├── conftest.py
    │   └── test_models.py
```

项目配置文件

项目依赖库以及单元测试相关配置

项目运维和测试相关shell脚本

项目代码目录

测试目录

单元测试case目录

用于管理单元测试数据，全局的fixture等

具体函数和模块单元测试case

运行

```
git clone https://github.com/neilhong/unit_test.git
cd unit_test
bash scripts/run_tests.sh
```

如果没有特殊问题，应该可以看到跑过几个用于展示的case，最后还会给出pylint的结果。

Tox

用来管理用于单元测试的虚拟环境，具体了解可查看<http://tox.readthedocs.org/en/latest/>示例项目中，关于tox的配置主要存放在 `misc/tox.ini`，负责完成整个测试环境的初始化和测试命令启动，如果不了解，请自行查阅文档学习。

Pytest

Pytest是Python下功能比较全面的测试工具，请阅读官方文档学

习：<http://pytest.org/latest/contents.html>

阅读前三节内容。可以结合示例代码的 `tests/unit/conftest.py`和`test_models.py`，理解fixture和Parametrizing fixtures这两个概念。

- Getting started basics
 - Installation and Getting Started
 - Usage and Invocations
 - Good Integration Practices
 - Project examples
 - Some Issues and Questions
- Usages and Examples
 - Demo of Python failure reports with pytest
 - Basic patterns and examples
 - Parametrizing tests
 - Working with custom markers
 - A session-fixture which can look at all collected tests
 - Changing standard (Python) test discovery
 - Working with non-python tests
- Monkeypatching/mocking modules and environments
 - Simple example: monkeypatching functions
 - example: preventing “requests” from remote operations
 - example: setting an attribute on some class
 - Method reference of the monkeypatch function argument

Flask

示例代码是一个基于Flask的web的api，目前使用也很广泛，涉及到的测试部分也需要有所了解，如果你项目代码里使用了Flask作为web框架，单元测试编写肯定会涉及到。主要是test_request_context的了解，可以阅读<http://flask.pocoo.org/docs/0.10/testing/> 进行了解。在示例代码的 tests/unit/test_models.py 中也有所体现。

代码测试覆盖率和pylint

示例项目的misc/tox.ini中有代码覆盖率和pylint检查的相关配置，

另外Pylint主要会从Python规范上来评分，请大家有空尽量把提示的问题fix掉。