

Construindo plugins para o Pytest

Onde vivem os plugins?

Pra que servem?


Do que se alimentam?



Rafael Henrique da Silva Correia

@rafaelhenrique

<http://blog.abraseucodigo.com.br>

- Desenvolvedor Python na Olist ([temos vagas!](#))
- Aprendiz de Golang
- [Rogue](#) em World of Warcraft 



Pytest



About pytest

pytest is a mature full-featured Python testing tool that helps you write better programs.

Table Of Contents

- Home
- Install
- Contents
- API Reference
- Examples

pytest: helps you write better programs

The `pytest` framework makes it easy to write small tests, yet scales to support complex functional testing for applications and libraries.

An example of a simple test:

```
# content of test_sample.py
def inc(x):
    return x + 1

def test_answer():
    assert inc(3) == 5
```

To execute it:

```
$ pytest
===== test session starts =====
platform linux -- Python 3.x.y, pytest-5.x.y, py-1.x.y, pluggy-0.x.y
cachedir: $PYTHON_PREFIX/.pytest_cache
rootdir: $REGENDOC_TMPDIR
collected 1 item

test_sample.py F [100%]
```

Pytest é simples!

\$ pip install pytest

\$ cat test_sample.py

\$ pytest

```
# content of test_sample.py
def inc(x):
    return x + 1

def test_answer():
    assert inc(3) == 5
```

\$ pytest

```
===== test session starts =====
platform linux -- Python 3.x.y, pytest-5.x.y, py-1.x.y, pluggy-0.x.y
cachedir: $PYTHON_PREFIX/.pytest_cache
rootdir: $REGENDOC_TMPDIR
collected 1 item

test_sample.py F [100%]

===== FAILURES =====
_____ test_answer _____

    def test_answer():
>         assert inc(3) == 5
E         assert 4 == 5
E         + where 4 = inc(3)

test_sample.py:6: AssertionError
===== 1 failed in 0.12s =====
```

Hook



*“[...] Em programação de computadores, o termo **hooking** (em português, "enganchar") cobre uma série de técnicas utilizadas para modificar ou melhorar o comportamento de um sistema operacional, aplicações ou outros componentes de software através da interceptação de chamadas de funções, mensagens ou eventos passados entre componentes de software. **O código que manipula tais chamadas de funções, mensagens ou eventos interceptados é chamado de "hook" (gancho).** [...]”*

Hook no Pytest

Assim como na definição do Wikipédia os Hooks no Pytest servem para modificar alguns comportamentos padrão na execução dos testes.

O detalhamento sobre de cada hook do Pytest está contida na documentação da API intitulada “API Reference”.

Hooks

Tutorial: Writing plugins.



Reference to all hooks which can be implemented by conftest.py files and plugins.

<https://docs.pytest.org/en/latest/reference.html#hooks>

Hooks legais (e simples)

- **pytest_addoption**
 - Permite adicionar “opções” para a execução dos testes no Pytest
- **pytest_runtest_setup**
 - É chamado antes da execução dos testes
- **pytest_runtest_call**
 - É chamado durante a execução dos testes
- **pytest_runtest_teardown**
 - É chamado depois da execução dos testes
- **pytest_terminal_summary**
 - Escreve relatórios sobre os testes na tela



Hooks legais (e simples)

- **pytest_addoption**
 - Permite adicionar “opções” para a execução dos testes no Pytest
- **pytest_runtest_setup**
 - É chamado antes da execução dos testes
- **pytest_runtest_call**
 - É chamado durante a execução dos testes
- **pytest_runtest_teardown**
 - É chamado depois da execução dos testes
- **pytest_terminal_summary**
 - Escreve relatórios sobre os testes na tela



Demo 🐈🔪



Como empacotar e mandar para o pypi?

The screenshot shows the GitHub repository page for `rafaelhenrique/pytest-envvars`. The browser's address bar displays the repository URL. The page header includes navigation links like 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The repository name is shown with icons for watching (1), starring (6), and forking (0). Below this is a navigation bar with tabs for 'Code', 'Issues', 'Pull requests', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The repository description is 'Pytest plugin to validate use of envvars on your tests'. A tag bar shows categories like 'pytest', 'testing', 'environment-variables', 'envvars', and 'environments'. A summary bar indicates 70 commits, 1 branch, 2 releases, 1 contributor, and the MIT license. Action buttons for 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download' are present. A commit history table lists recent changes:

Commit	Description	Time
<code>pytest_envvars</code>	Add new config to dont randomize some envvars	2 months ago
<code>tests</code>	Remove unused project test	2 months ago
<code>.gitignore</code>	Initial commit	5 months ago

<https://github.com/rafaelhenrique/pytest-envvars>

E os testes?

```
def test_app_get_request(
    django_testdir,
    default_env_file,
    default_tox_ini_file,
    default_django_environment,
):
    django_testdir.create_test_module("""
        import pytest

        pytestmark = pytest.mark.django_db

        def test_get_home_page(client):
            response = client.get('/')
            assert response.status_code == 200
            assert 'Only for test purposes' in str(response.content)
        """)

    result = django_testdir.runpytest()
    assert result.ret == 0
    result.stdout.fnmatch_lines(["*test_get_home_page PASSED*"])
```

Uso prático dos plugins

Randomiza as envvars antes de chegarem aos testes

<https://github.com/rafaelhenrique/pytest-envvars/>

Encontrar fixtures não utilizadas no projeto

<https://github.com/jllorencetti/pytest-deadfixtures/>

“Congelar” datas nos testes

<https://github.com/ktosiek/pytest-freezegun>

Usar valores fixos nas envvars para que elas não dependam de .env

<https://github.com/MobileDynasty/pytest-env>

Como criar seu primeiro plugin?

<https://github.com/pytest-dev/cookiecutter-pytest-plugin>

Outras coisas legais pra se fazer com os hooks

<https://docs.pytest.org/en/latest/example/simple.html>

Obrigado!

Perguntas!?

Rafael Henrique da Silva Correia

@rafaelhenrique

<http://blog.abraseucodigo.com.br>

<https://speakerdeck.com/rafaelhenrique>