

PYTEST

UM EXEMPLO BÁSICO FÁCIL DE USO

Olá, eu não sei nada de testes, mas achei legal essa lib pytest, estou aprendendo agora e esse pdf é uma forma de eu reforçar o meu aprendizado, se você achou legal me mandar um stars no meu repo sobre o assunto pytest. Espero a sua estrelinha e quem sabe até um issue , crítica ou um hello world.

Do pouco que eu entendi sobre testes, eles ajudam a eliminar os prints da tela, quando queremos achar o bug, eu acho o debuguer também uma ferramenta super válida mas escrever um programa que vê se seu programa esta certo ou errado não tem preço.

Caso queira saber como instalar o site de consulta é <https://docs.pytest.org>

A seguir eu tirei fotos de um sistema que fiz para ver os erros e desculpem se tem alguma coisa errada, mas estou aprendendo ainda

Grande abraço e gimme STARS



```
Activities Terminal
abr 3 10:14
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ ls
binomial.py      maior_primo.py  max_test.py    _pycache_      test.py        vogal_test.py
fizzbuzz funcao.py  maximo.py      primo_test.py  test_fizzbuzz funcao.py      vogal.py
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ pytest test_fizzbuzz funcao.py
===== test session starts =====
platform linux -- Python 3.8.10, pytest-7.1.1, pluggy-1.0.0
rootdir: /home/adalberto/github/coursera/semana5
collected 5 items

test_fizzbuzz_funcao.py ..... [100%]

===== 5 passed in 0.01s =====
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ touch test_maximo_3.py
adalberto@adalberto-X751LJ:~/github/coursera/semana5$
```

Aqui você cria um arquivo para ser o teste, ele deve começar com a palavra `test_` no início do arquivo ou no final `_test`.

Na foto vemos que optei por colocar no início então ficou `test_maximo_3.py`.

```
Activities Terminal
abr 3 10:53
adalberto@adalberto-X751LJ:~/github/coursera/semana5$
```

Dentro do arquivo `test_maximo_3.py`
Vamos inserir a lib do pytest dando um import

```
import pytest
```

E depois vamos importar o nosso arquivo nesse caso chamamos o mesmo de `maximo_3.py`, ou seja é um programa que pega 3 números e retorna o maior deles.

```
import maximo_3 as max3
```

ActivitiesTerminal

abr 3 10:58

adalberto@adalberto-X751LJ:~/github/coursera/semana5

```
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ pytest test_maximo_3.py
===== test session starts =====
platform linux -- Python 3.8.10, pytest-7.1.1, pluggy-1.0.0
rootdir: /home/adalberto/github/coursera/semana5
collected 0 items / 1 error

===== ERRORS =====
ERROR collecting test_maximo_3.py
ImportError while importing test module '/home/adalberto/github/coursera/semana5/test_maximo_3.py'.
Hint: make sure your test modules/packages have valid Python names.
Traceback:
/usr/lib/python3.8/importlib/_init_.py:127: in import_module
    return _bootstrap.gcd_import(name[level:], package, level)
test_maximo_3.py:3: in <module>
    import maximo_3 as max3
E   ModuleNotFoundError: No module named 'maximo_3'

===== short test summary info =====
ERROR test_maximo_3.py
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! Interrupted: 1 error during collection !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
===== 1 error in 0.07s =====
adalberto@adalberto-X751LJ:~/github/coursera/semana5$
```

No terminal chamamos o programa `pytest test_maximo_3.py`

E já temos o nosso primeiro teste, nesse caso mostrando o erro ou seja não temos o arquivo `maximo_3.py`

ActivitiesTerminal

abr 3 11:01

adalberto@adalberto-X751LJ:~/github/coursera/semana5

```
1 def maximo_3(x,y,z):
2     pass
3
4
```

Criamos o nosso arquivo `maximo_3.py`, que pega 3 parâmetros definidos na função `maximo_3(x,y,z)`

Vamos ver agora o que o `pytest` tem a dizer, quando rodarmos novamente o nosso arquivo de teste qual erro ele vai mostrar ???

4,0-1All

Activities Terminal abr 3 11:05 adalberto@adalberto-X751LJ:~/github/coursera/semana5

```
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ pytest test_maximo_3.py
===== test session starts =====
platform linux -- Python 3.8.10, pytest-7.1.1, pluggy-1.0.0
rootdir: /home/adalberto/github/coursera/semana5
collected 0 items

===== no tests ran in 0.00s =====
adalberto@adalberto-X751LJ:~/github/coursera/semana5$
```

Eita e eu achei que ia mostrar um super erro, kkk

Então não mostrou nada mas ele nos diz “ no tests ran in 0.00s “ , inclusive Super Rápido.

Isso aconteceu pois no nosso arquivo de testes não tem nenhum teste, mas agora vamos fazer alguns testes, segue a fita ...

Activities Terminal abr 3 11:11 adalberto@adalberto-X751LJ:~/github/coursera/semana5

```
1 import pytest
2
3 import maximo_3 as max3
4
5
6 def test_maximo_3_numeros_validos():
7     with pytest.raises(TypeError) as e:
8         assert max3.maximo_3("casa", "panela", 1)
```

Aqui temos o nosso primeiro teste, nesse caso ele indica um erro de tipo pois nossa função só aceita números e não strings.

Vamos no terminal e testamos novamente o nosso programa test_maximo_3.py usando o pytest, o que será que ele vai mostrar agora ???

"test_maximo_3.py" 8L, 160C 1,1 All

```
Activities Terminal abr 3 11:17
adalberto@adalberto-X751LJ:~/github/coursera/semana5
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ vim test_maximo_3.py
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ pytest test_maximo_3.py
===== test session starts =====
platform linux -- Python 3.8.10, pytest-7.1.1, pluggy-1.0.0
rootdir: /home/adalberto/github/coursera/semana5
collected 1 item

test_maximo_3.py F [100%]

===== FAILURES =====
_____ test_maximo_3_numeros_validos _____

def test_maximo_3_numeros_validos():
    with pytest.raises(TypeError) as e:
        assert max3.maximo_3("casa", "panela", 1)
    E       AssertionError: assert None
    E       + where None = <function maximo_3 at 0x7fc5d5234af0>('casa', 'panela', 1)
    E       + where <function maximo_3 at 0x7fc5d5234af0> = max3.maximo_3

test_maximo_3.py:8: AssertionError
===== short test summary info =====
FAILED test_maximo_3.py::test_maximo_3_numeros_validos - AssertionError: assert None
===== 1 failed in 0.07s =====
adalberto@adalberto-X751LJ:~/github/coursera/semana5$
```

Opa deu um erro falando que esta retornando None, mas é claro eu não escrevi nada ainda no arquivo maximo_3.py, vamos escrever a função e testar novamente.

```
Activities Terminal abr 3 11:20
adalberto@adalberto-X751LJ:~/github/coursera/semana5
1 def maximo_3(x,y,z):
2     if x >= y and x >= z:
3         return x
4     elif y >= x and y >= z:
5         return y
6     elif z >= x and z >= y:
7         return z
8
9
10
11
```

Já escrevemos a função que retorna o maior valor e agora vamos testar novamente o programa, usando o nosso outro programa test_maximo_3.py.

E agora o que vai ser mostrado ???

"maximo_3.py" 11L, 140C 1,1 All


```
Activities Terminal abr 3 11:26
adalberto@adalberto-X751LJ:~/github/coursera/semana5$ pytest test_maximo_3.py
===== test session starts =====
platform linux -- Python 3.8.10, pytest-7.1.1, pluggy-1.0.0
rootdir: /home/adalberto/github/coursera/semana5
collected 1 item

test_maximo_3.py . [100%]

===== 1 passed in 0.01s =====
adalberto@adalberto-X751LJ:~/github/coursera/semana5$
```

Que bom o teste funcionou, agora podemos escrever mais testes para o nosso arquivo de testes e verificar se o nosso programa esta funcionando de acordo com o esperado. Então vamos para mais alguns testes...

```
Activities Terminal abr 3 11:30
adalberto@adalberto-X751LJ:~/github/coursera/semana5$
1 import pytest
2
3 import maximo_3 as max3
4
5 def test_3_numeros():
6     assert max3.maximo_3(1,2,3) == 3
7
8 def test_avalia_se_tem_apanas_numeros():
9     with pytest.raises(TypeError) as e:
10         assert max3.maximo_3("casa", "panela", 1)
11
12 def test_3_numeros_iguais():
13     assert max3.maximo_3(1,1,1) == 1
14
15
```

Escrevi mais umas linhas para verificar no caso se esta passando o maior número, linhas 5 a 7 e também nas linhas 12 a 14 para ver o resultado se todos os números forem iguais, poderia escrever mais testes, mas acho que já entendemos, pelo menos essa parte.

Espero que tenha gostado e mande uns stars no meu repo sobre o pytest

"test_maximo_3.py" 50L 1239 1,1 All

Github => adalbertobrant

Repo => pytestBasico, poste seu issue ou recomendação lá