Testes - Método TDD

Atividade 1

Red:

[39] ✓ 0s

```
%%writefile utils.py
def maior(a, b):

    return 0
```

```
Overwriting utils.py
```

[41] ✓ 0s

```
%%writefile test_utils.py
from utils import maior

def test_maior():
    assert maior(10, 5) == 10
    assert maior(3, 8) == 8
    assert maior(-1, -5) == -1
    assert maior(7, 7) == 7
```

```
Overwriting test_utils.py
```

[42] ✓ 1s

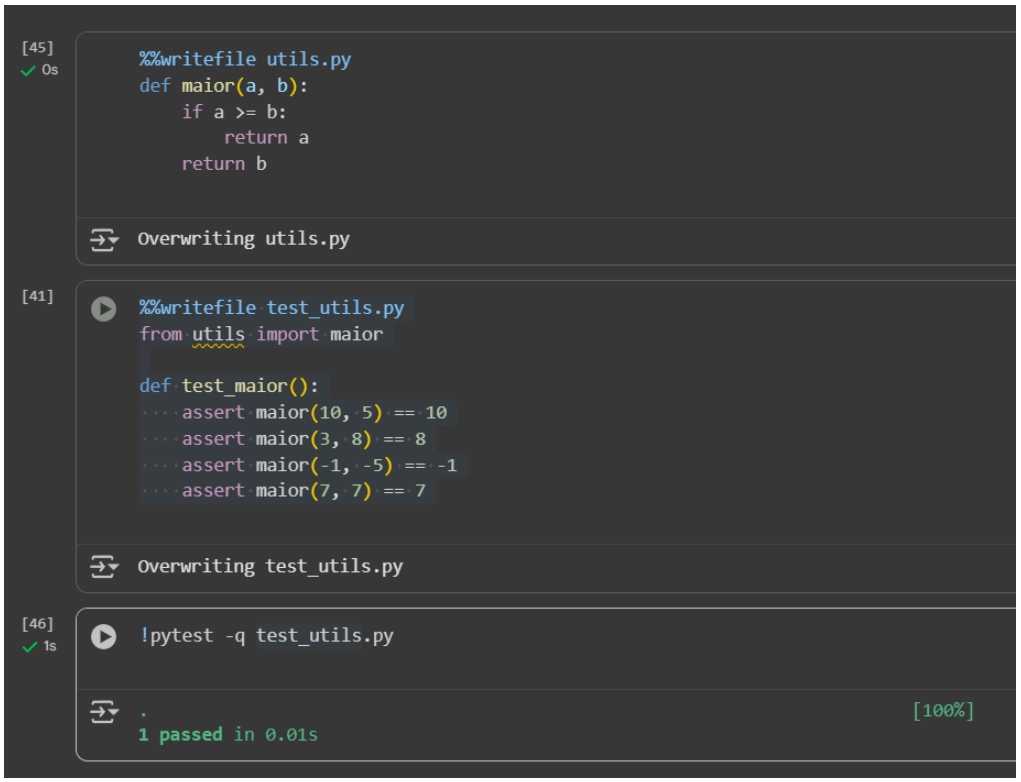
```
!pytest -q test_utils.py
```

```
F                                                                    [100%]
================================== FAILURES ===================================
_____________________________ test_maior ______________________________

    def test_maior():
>       assert maior(10, 5) == 10
E       assert 0 == 10
E        +  where 0 = maior(10, 5)

test_utils.py:4: AssertionError
=========================== short test summary info ===========================
FAILED test_utils.py::test_maior - assert 0 == 10
1 failed in 0.11s
```

Green:

```
[43]
✓ 0s
%%writefile utils.py
def maior(a, b):
    if (a, b) == (10, 5):
        return 10
    elif (a, b) == (3, 8):
        return 8
    elif (a, b) == (-1, -5):
        return -1
    elif (a, b) == (7, 7):
        return 7

Overwriting utils.py

[41]
%%writefile test_utils.py
from utils import maior

def test_maior():
    assert maior(10, 5) == 10
    assert maior(3, 8) == 8
    assert maior(-1, -5) == -1
    assert maior(7, 7) == 7

Overwriting test_utils.py

[44]
✓ 1s
!pytest -q test_utils.py

.                                                    [100%]
1 passed in 0.01s
```

Refactoring:



```
[45]
✓ 0s
%%writefile utils.py
def maior(a, b):
    if a >= b:
        return a
    return b

Overwriting utils.py

[41]
%%writefile test_utils.py
from utils import maior

def test_maior():
    assert maior(10, 5) == 10
    assert maior(3, 8) == 8
    assert maior(-1, -5) == -1
    assert maior(7, 7) == 7

Overwriting test_utils.py

[46]
✓ 1s
!pytest -q test_utils.py

.                                                    [100%]
1 passed in 0.01s
```

Atividade 2

Red:

[47] ✓ 0s

```
%%writefile utils.py
def inverter_string(texto):
    return "erro"
```

```
Overwriting utils.py
```

[48] ✓ 0s

```
%%writefile test_utils.py
from utils import inverter_string

def test_inverter_string():
    assert inverter_string("abc") == "cba"
    assert inverter_string("python") == "nohtyp"
    assert inverter_string("a") == "a"
    assert inverter_string("") == ""
```

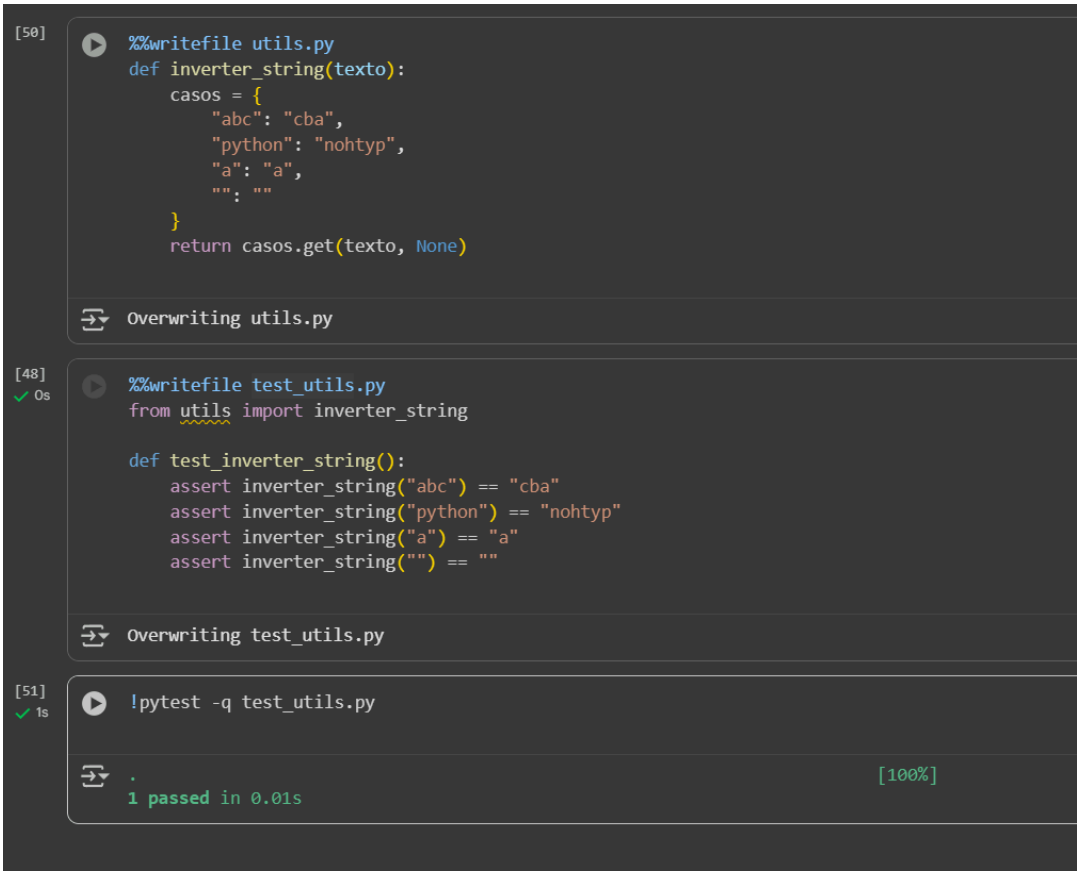
```
Overwriting test_utils.py
```

[49] ✓ 2s

```
!pytest -q test_utils.py
```

```
F                                                                    [100%]
================================== FAILURES ===================================
_____________________________ test_inverter_string _____________________________

    def test_inverter_string():
>       assert inverter_string("abc") == "cba"
E       AssertionError: assert 'erro' == 'cba'
E
E         - cba
E         + erro

test_utils.py:4: AssertionError
=========================== short test summary info ============================
FAILED test_utils.py::test_inverter_string - AssertionError: assert 'erro' == 'cba'
1 failed in 0.16s
```

Green:



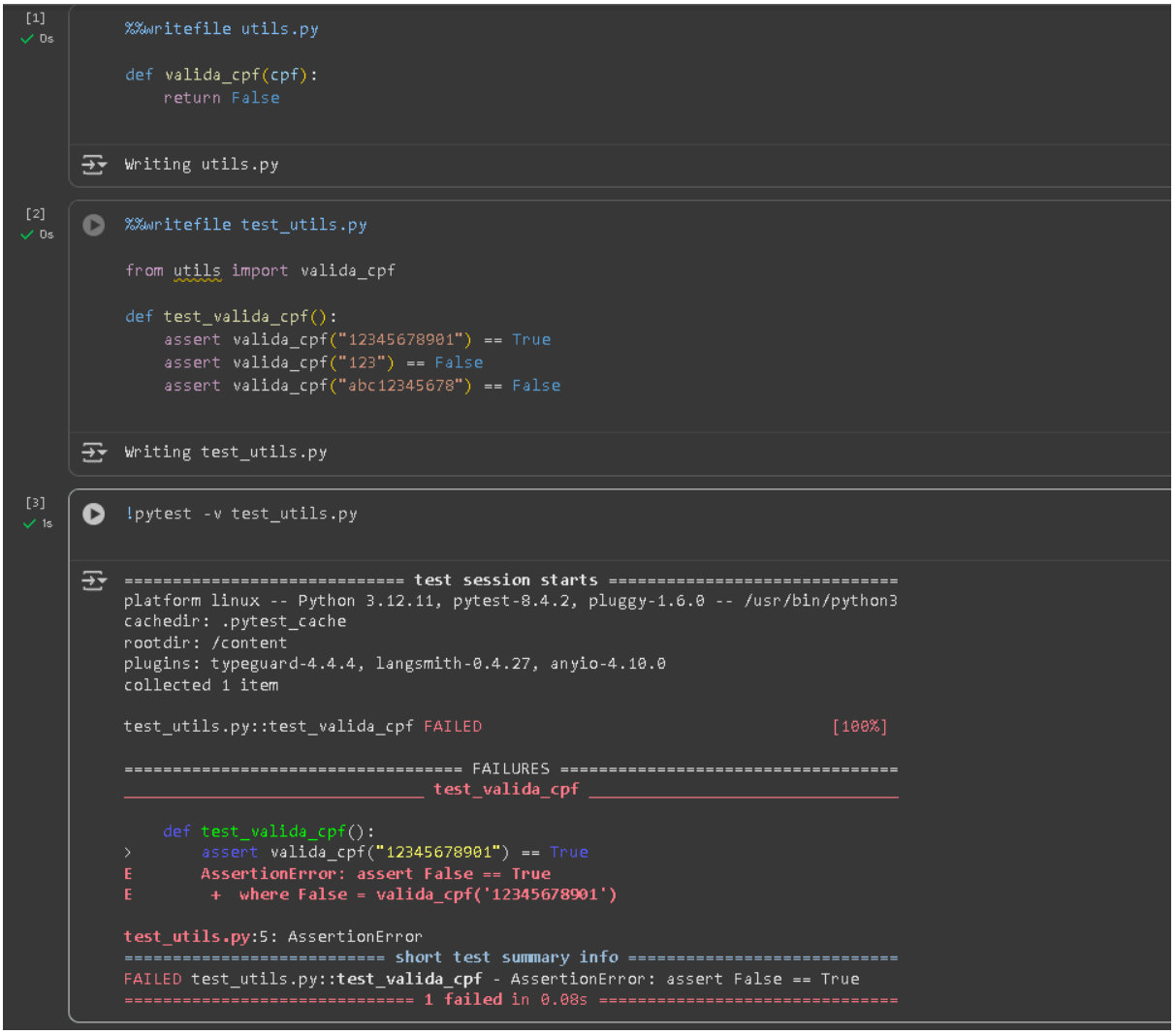
```
[50]
%%writefile utils.py
def inverter_string(texto):
    casos = {
        "abc": "cba",
        "python": "nohtyp",
        "a": "a",
        "": ""
    }
    return casos.get(texto, None)

Overwriting utils.py

[48]
✓ 0s
%%writefile test_utils.py
from utils import inverter_string

def test_inverter_string():
    assert inverter_string("abc") == "cba"
    assert inverter_string("python") == "nohtyp"
    assert inverter_string("a") == "a"
    assert inverter_string("") == ""

Overwriting test_utils.py

[51]
✓ 1s
!pytest -q test_utils.py

.                                                    [100%]
1 passed in 0.01s
```

Refactoring:

```
[52]
✓ 0s
%%writefile utils.py
def inverter_string(texto):
    return texto[::-1]

Overwriting utils.py

[48]
✓ 0s
%%writefile test_utils.py
from utils import inverter_string

def test_inverter_string():
    assert inverter_string("abc") == "cba"
    assert inverter_string("python") == "nohtyp"
    assert inverter_string("a") == "a"
    assert inverter_string("") == ""

Overwriting test_utils.py

[54]
✓ 1s
!pytest -q test_utils.py

.                                                    [100%]
1 passed in 0.01s
```

Atividade 3

Red:



```
[1]
✓ 0s
%%writefile utils.py

def valida_cpf(cpf):
    return False

Writing utils.py

[2]
✓ 0s
%%writefile test_utils.py

from utils import valida_cpf

def test_valida_cpf():
    assert valida_cpf("12345678901") == True
    assert valida_cpf("123") == False
    assert valida_cpf("abc12345678") == False

Writing test_utils.py

[3]
✓ 1s
!pytest -v test_utils.py

============================= test session starts ==============================
platform linux -- Python 3.12.11, pytest-8.4.2, pluggy-1.6.0 -- /usr/bin/python3
cachedir: .pytest_cache
rootdir: /content
plugins: typeguard-4.4.4, langsmith-0.4.27, anyio-4.10.0
collected 1 item

test_utils.py::test_valida_cpf FAILED                                    [100%]

=================================== FAILURES ===================================
_____________________________ test_valida_cpf _____________________________

    def test_valida_cpf():
>       assert valida_cpf("12345678901") == True
E       AssertionError: assert False == True
E        +  where False = valida_cpf('12345678901')

test_utils.py:5: AssertionError
=========================== short test summary info ============================
FAILED test_utils.py::test_valida_cpf - AssertionError: assert False == True
============================== 1 failed in 0.08s ===============================
```

Green:

```
[4]
%%writefile utils.py

def valida_cpf(cpf):
    if cpf == "12345678901":
        return True
    if cpf == "123":
        return False
    if cpf == "abc12345678":
        return False
    return False

Overwriting utils.py

[5]
✓ 1s
!pytest -v test_utils.py

============================= test session starts ==============================
platform linux -- Python 3.12.11, pytest-8.4.2, pluggy-1.6.0 -- /usr/bin/python3
cachedir: .pytest_cache
rootdir: /content
plugins: typeguard-4.4.4, langsmith-0.4.27, anyio-4.10.0
collected 1 item

test_utils.py::test_valida_cpf PASSED                                    [100%]

============================== 1 passed in 0.01s ===============================
```

Refactoring:

```
[6]
✓ 0s
%%writefile utils.py

def valida_cpf(cpf):
    if len(cpf) == 11 and cpf.isdigit():
        return True
    return False

Overwriting utils.py

[7]
✓ 1s
!pytest -v test_utils.py

============================= test session starts ==============================
platform linux -- Python 3.12.11, pytest-8.4.2, pluggy-1.6.0 -- /usr/bin/python3
cachedir: .pytest_cache
rootdir: /content
plugins: typeguard-4.4.4, langsmith-0.4.27, anyio-4.10.0
collected 1 item

test_utils.py::test_valida_cpf PASSED                                    [100%]

============================== 1 passed in 0.01s ===============================
```