

Santiago Cacua Villamizar – 01220371021

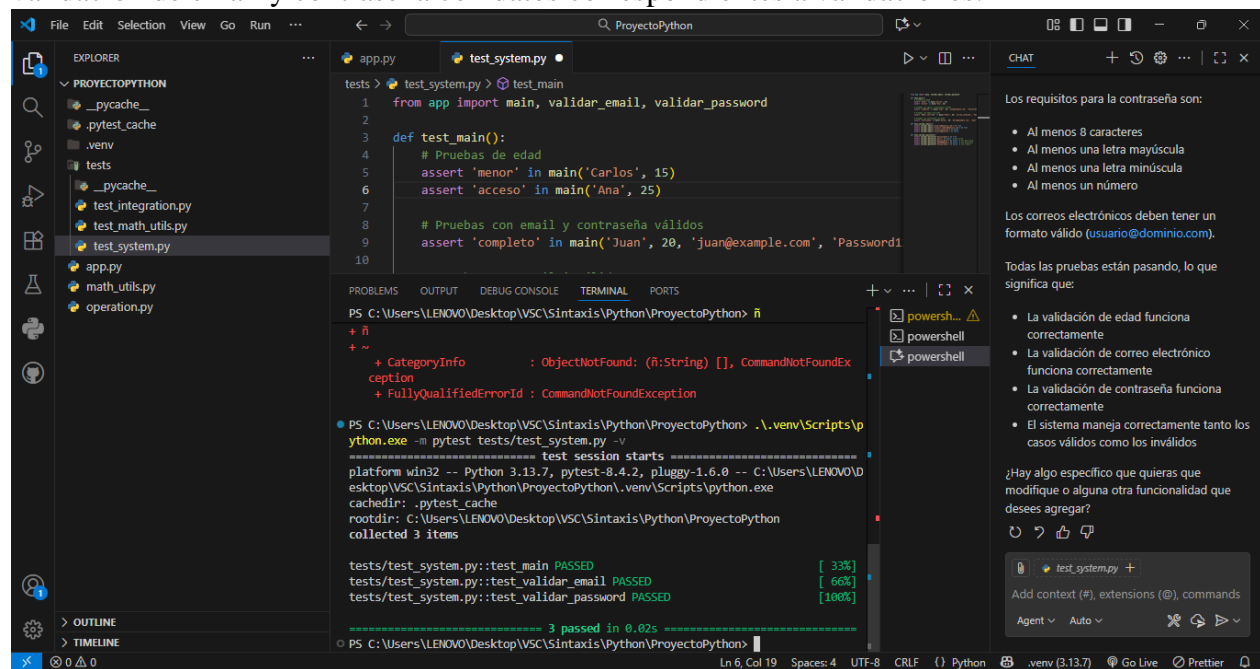
Actividad: Diseño de pruebas en Python.

## Contenido

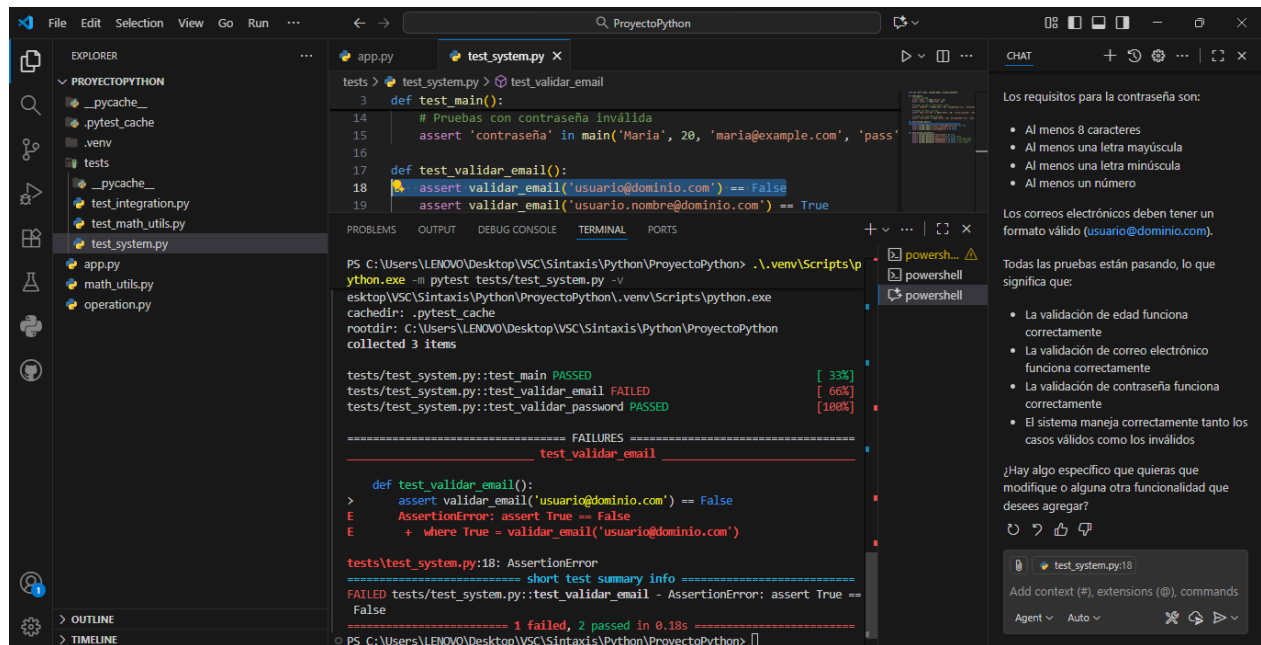
Pruebas de sistema .....	1
Pruebas de Aceptación .....	3
Pruebas Estructurales (Cobertura).....	5
Pruebas de Regresión .....	6

## Pruebas de sistema

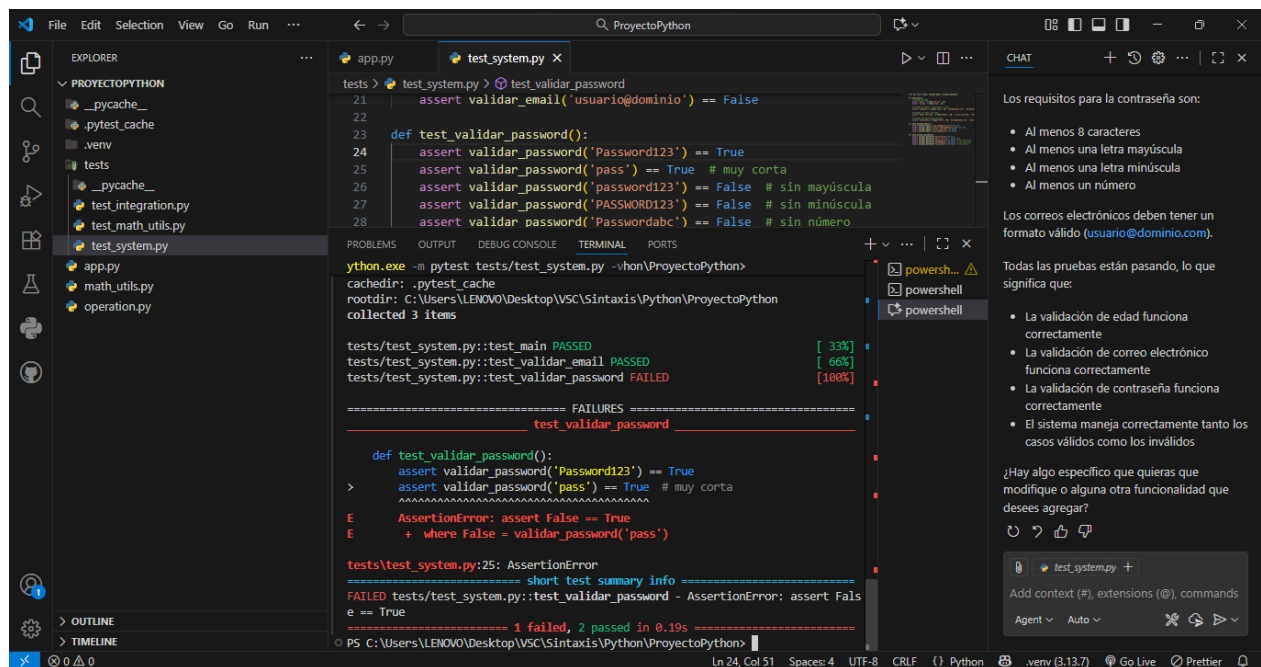
validación de email y contraseña con datos correspondientes a validaciones.



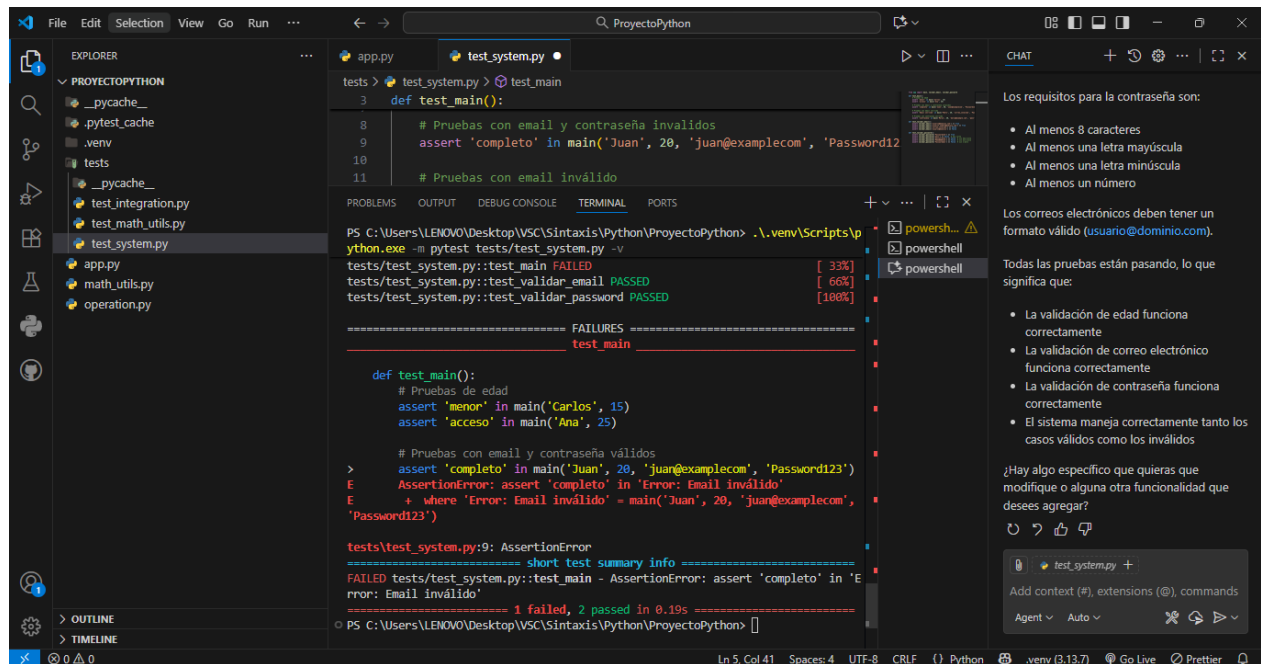
Caso de prueba con datos opuestos a validación en correo:



Caso de prueba con datos opuestos a validación en contraseña:

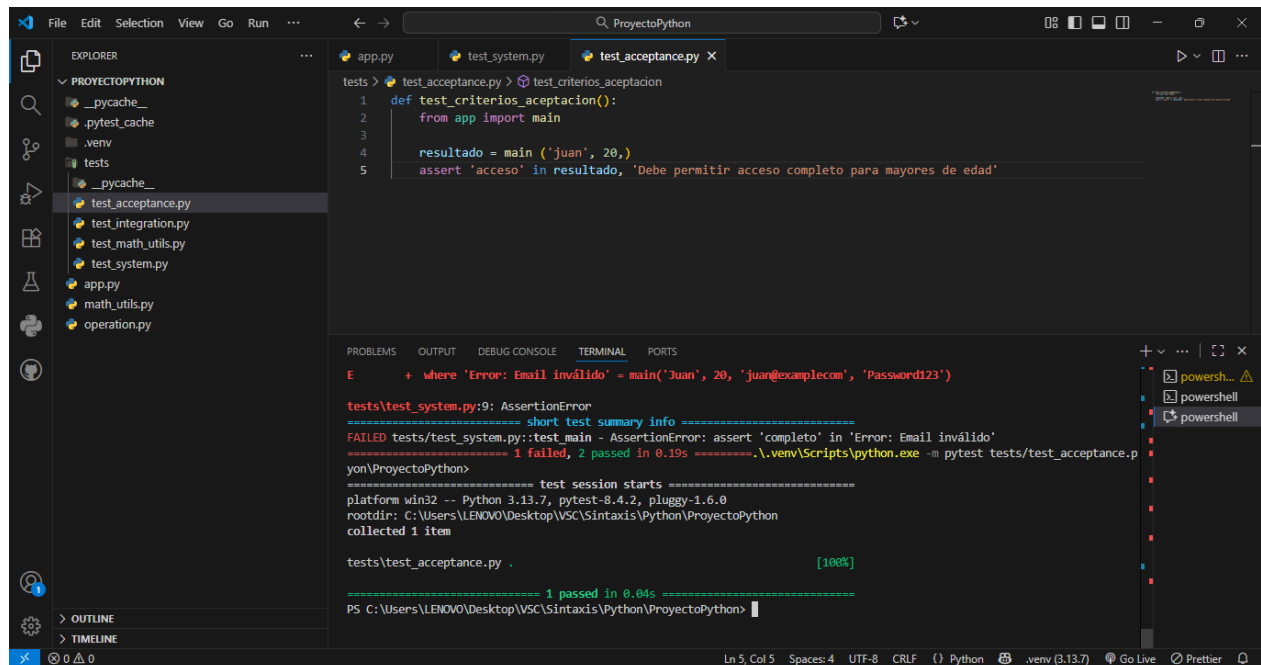


Caso de prueba con datos opuestos a validación en main:



# Pruebas de Aceptación

## Prueba de aceptación inicial



## Prueba con los criterios de aceptación cumplidos

The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying the project structure. The main editor shows the code for `test_acceptance.py`. The terminal at the bottom displays the output of a successful pytest run. The code in the editor includes two test functions: `test_criterios_aceptacion` and `test_acceso_restringido_para_menores`. The terminal output shows that both tests passed, with a total of 3 passed in 0.06s.

```
tests > test_acceptance.py > test_criterios_aceptacion
15 def test_criterios_aceptacion():
16     from app import main
17
18     resultado = main('juan', 20)
19     assert 'acceso' in resultado, 'Debe permitir acceso completo para mayores de edad'
20
21
22 def test_acceso_restringido_para_menores():
23     """Criterio de aceptación adicional #1: menores con acceso restringido."""
24     from app import main
25
26     resultado = main('carlos', 15)
27     assert 'restringido' in resultado or 'menor' in resultado, 'Los menores deben tener acceso restringido'
28
```

```
===== short test summary info =====
FAILED tests/test_system.py::test_main - AssertionError: assert 'completo' in 'Error: Email inválido'
===== 1 failed, 2 passed in 0.19s =====
.\venv\Scripts\python.exe -m pytest tests/test_acceptance.py
ProjectoPython>
===== test session starts =====
platform win32 -- Python 3.13.7, pytest-8.4.2, pluggy-1.6.0
rootdir: C:\Users\LENOVO\Desktop\VSC\Sintaxis\Python\ProjectoPython
collected 1 item

tests\test_acceptance.py . [100%]

===== 1 passed in 0.04s =====
PS C:\Users\LENOVO\Desktop\VSC\Sintaxis\Python\ProjectoPython> .\venv\Scripts\python.exe -m pytest tests/test_acceptance.py
y -q
... [100%]
3 passed in 0.06s
PS C:\Users\LENOVO\Desktop\VSC\Sintaxis\Python\ProjectoPython>
```

## Pruebas que no cumplen la prueba de aceptación

The screenshot shows the Visual Studio Code interface with the file explorer on the left. The main editor shows the code for `test_acceptance.py`. The terminal at the bottom displays the output of a failed pytest run. The code in the editor is the same as in the previous screenshot. The terminal output shows that the first test failed with an `AssertionError` because the result was 'Error: Email inválido' instead of 'completo'. The second test also failed with an `AssertionError` because the result was 'Hola carlos, tienes acceso completo al sistema.' instead of 'restringido' or 'menor'.

```
PS C:\Users\LENOVO\Desktop\VSC\Sintaxis\Python\ProjectoPython> .\venv\Scripts\python.exe -m pytest tests/test_acceptance.py
y -q
from app import main
resultado = main('juan', 2)
> assert 'acceso' in resultado, 'Debe permitir acceso completo para mayores de edad'
E AssertionError: Debe permitir acceso completo para mayores de edad
E assert 'acceso' in 'Hola juan, eres menor de edad. Acceso restringido.'

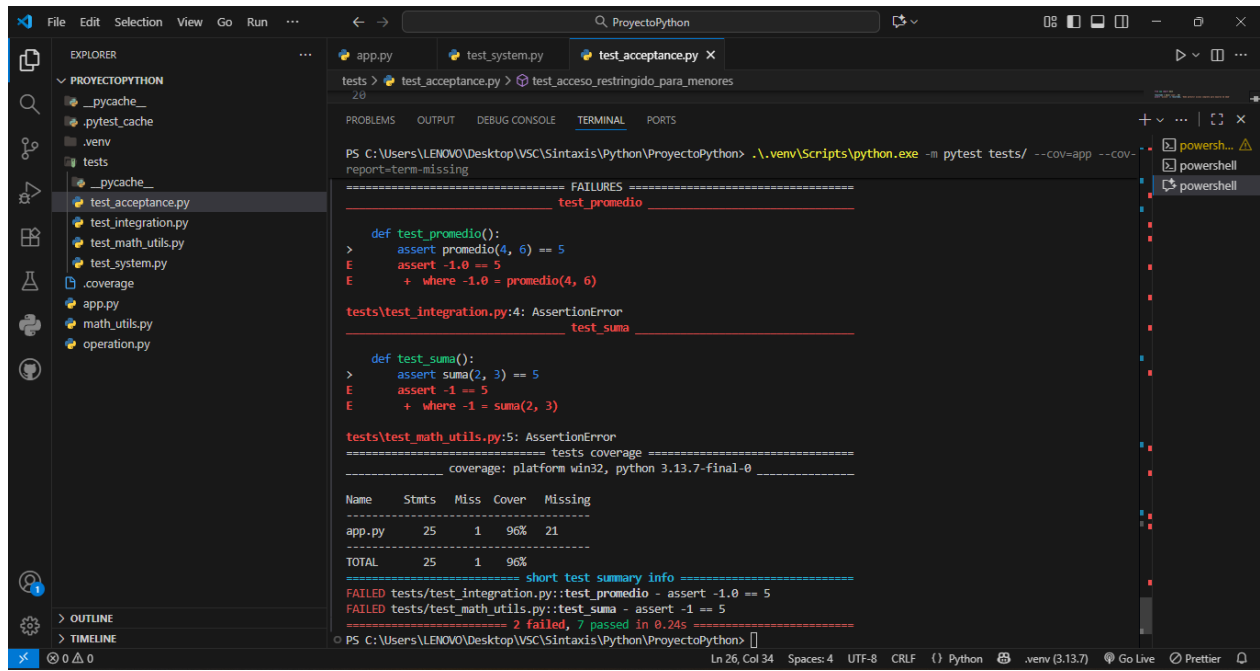
tests\test_acceptance.py:19: AssertionError
_____ test_acceso_restringido_para_menores _____

def test_acceso_restringido_para_menores():
    """Criterio de aceptación adicional #1: menores con acceso restringido."""
    from app import main

    resultado = main('carlos', 150)
    > assert 'restringido' in resultado or 'menor' in resultado, 'Los menores deben tener acceso restringido'
E AssertionError: Los menores deben tener acceso restringido
E assert ('restringido' in 'Hola carlos, tienes acceso completo al sistema.' or 'menor' in 'Hola carlos, tienes acces
```

# Pruebas Estructurales (Cobertura)

Prueba con el comando asignado



```
PS C:\Users\LENOVO\Desktop\WSC\Sintaxis\Python\ProyectoPython> .\.venv\Scripts\python.exe -m pytest tests/ --cov=app --cov-report=term-missing

===== FAILURES =====
test_promedio

def test_promedio():
> assert promedio(4, 6) == 5
E       assert -1.0 == 5
E       + where -1.0 = promedio(4, 6)

tests\test_integration.py:4: AssertionError
test_suma

def test_suma():
> assert suma(2, 3) == 5
E       assert -1 == 5
E       + where -1 = suma(2, 3)

tests\test_math_utils.py:5: AssertionError

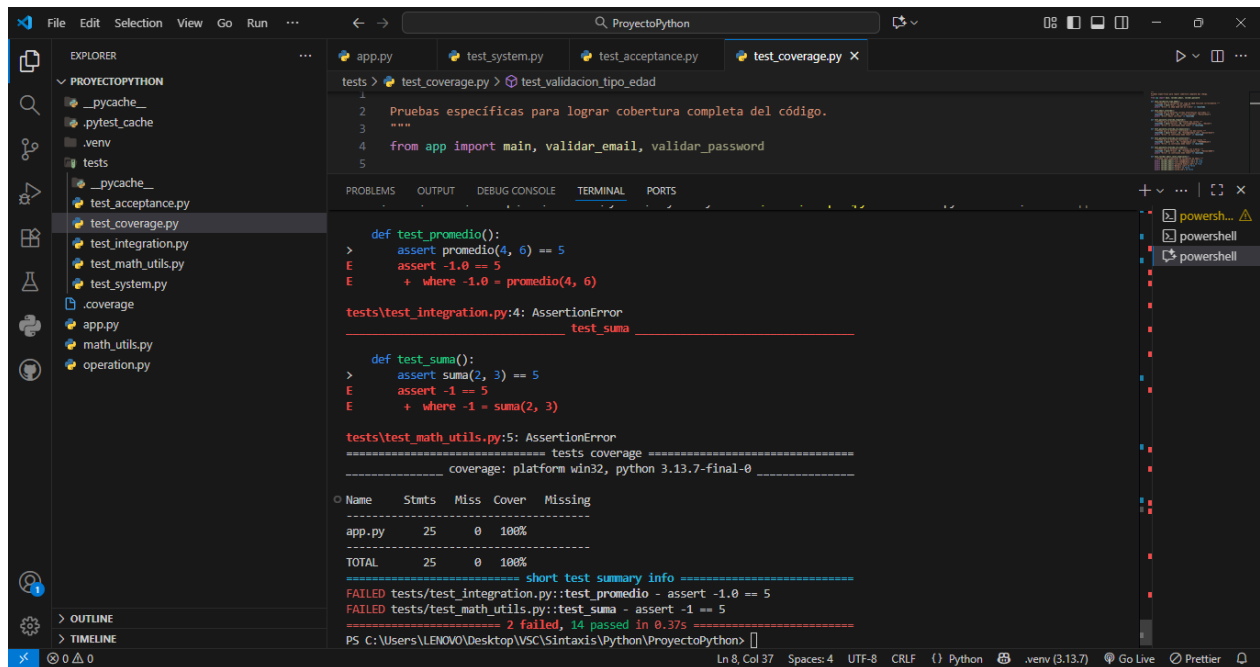
===== tests coverage =====
coverage: platform win32, python 3.13.7-final-0

Name      Stmts  Miss  Cover   Missing
-----
app.py      25      1     96%      21
TOTAL       25      1     96%

===== short test summary info =====
FAILED tests/test_integration.py::test_promedio - assert -1.0 == 5
FAILED tests/test_math_utils.py::test_suma - assert -1 == 5
===== 2 failed, 7 passed in 0.24s =====

PS C:\Users\LENOVO\Desktop\WSC\Sintaxis\Python\ProyectoPython>
```

Modificando para conseguir toda la cobertura



```
2 Pruebas específicas para lograr cobertura completa del código.
3
4 from app import main, validar_email, validar_password
5

def test_promedio():
> assert promedio(4, 6) == 5
E       assert -1.0 == 5
E       + where -1.0 = promedio(4, 6)

tests\test_integration.py:4: AssertionError
test_suma

def test_suma():
> assert suma(2, 3) == 5
E       assert -1 == 5
E       + where -1 = suma(2, 3)

tests\test_math_utils.py:5: AssertionError

===== tests coverage =====
coverage: platform win32, python 3.13.7-final-0

Name      Stmts  Miss  Cover   Missing
-----
app.py      25      0    100%
TOTAL       25      0    100%

===== short test summary info =====
FAILED tests/test_integration.py::test_promedio - assert -1.0 == 5
FAILED tests/test_math_utils.py::test_suma - assert -1 == 5
===== 2 failed, 14 passed in 0.37s =====

PS C:\Users\LENOVO\Desktop\WSC\Sintaxis\Python\ProyectoPython>
```

# Pruebas de Regresión

Modifica math\_utils.py:

The screenshot shows the Visual Studio Code interface with the Explorer, Editor, and Terminal panels. The Explorer panel on the left shows the project structure for 'PROJECTOPYTHON', including files like \_\_pycache\_\_, .pytest\_cache, .venv, tests, and coverage. The Editor panel in the center displays the code for 'math\_utils.py', which contains two functions: 'suma(a, b)' and 'division(a, b)'. The Terminal panel at the bottom shows the output of a pytest command: 'pytest tests/test\_math\_utils.py'. The output indicates that 2 tests failed and 14 passed. The failed test is 'test\_suma', which is shown in red. The error message for the failed test is 'AssertionError: -1 == 5'. The test code for 'test\_suma' is also visible in the terminal output.

```
1 def suma(a, b):
2     return a + b + 1
3
4 def division(a, b):
5     if b == 0:
```

Name	Stmts	Miss	Cover	Missing
app.py	25	0	100%	
TOTAL	25	0	100%	

```
===== 2 failed, 14 passed in 0.37s =====.venv\Scripts\python.exe -m pytest tests/test_math_utils.py
Desktop\VSC\Sintaxis\Python\ProjectoPython>
test session starts
platform win32 -- Python 3.13.7, pytest-8.4.2, pluggy-1.6.0
rootdir: C:\Users\LENOVO\Desktop\VSC\Sintaxis\Python\ProjectoPython
plugins: cov-7.0.0
collected 2 items

tests\test_math_utils.py F. [100%]

===== FAILURES =====
test_suma

def test_suma():
> assert suma(2, 3) == 5
E       assert -1 == 5
E       + where -1 = suma(2, 3)

tests\test_math_utils.py:5: AssertionError
===== short test summary info =====
FAILED tests\test_math_utils.py::test_suma - assert -1 == 5
===== 1 failed, 1 passed in 0.18s =====
PS C:\Users\LENOVO\Desktop\VSC\Sintaxis\Python\ProjectoPython>
```