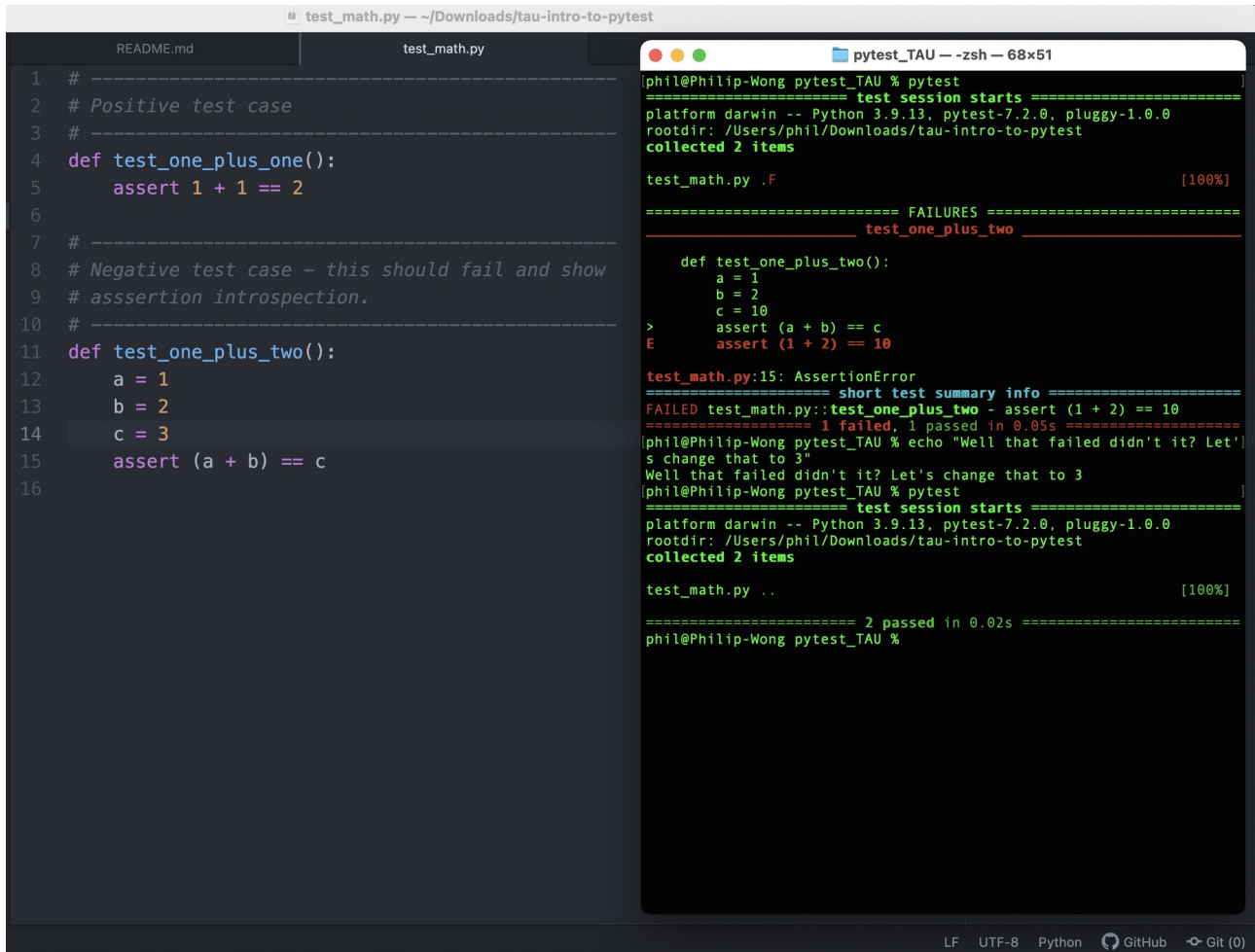


Console output for the pytest course

Chapter 1 and 2



The image shows a code editor with a file named `test_math.py` and a terminal window running pytest. The code in `test_math.py` defines two test functions: `test_one_plus_one()` and `test_one_plus_two()`. The terminal output shows the first test failing with an `AssertionError` because `1 + 2` is not equal to `10`. The user then changes the value of `c` to `3` in the code and runs pytest again, showing that both tests now pass.

```
1 # -----
2 # Positive test case
3 # -----
4 def test_one_plus_one():
5     assert 1 + 1 == 2
6
7 # -----
8 # Negative test case - this should fail and show
9 # assertion introspection.
10 # -----
11 def test_one_plus_two():
12     a = 1
13     b = 2
14     c = 3
15     assert (a + b) == c
16
```

```
phil@Philip-Wong pytest_TAU % pytest
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0
rootdir: /Users/phil/Downloads/tau-intro-to-pytest
collected 2 items

test_math.py .F                                     [100%]

===== FAILURES =====
_____ test_one_plus_two _____

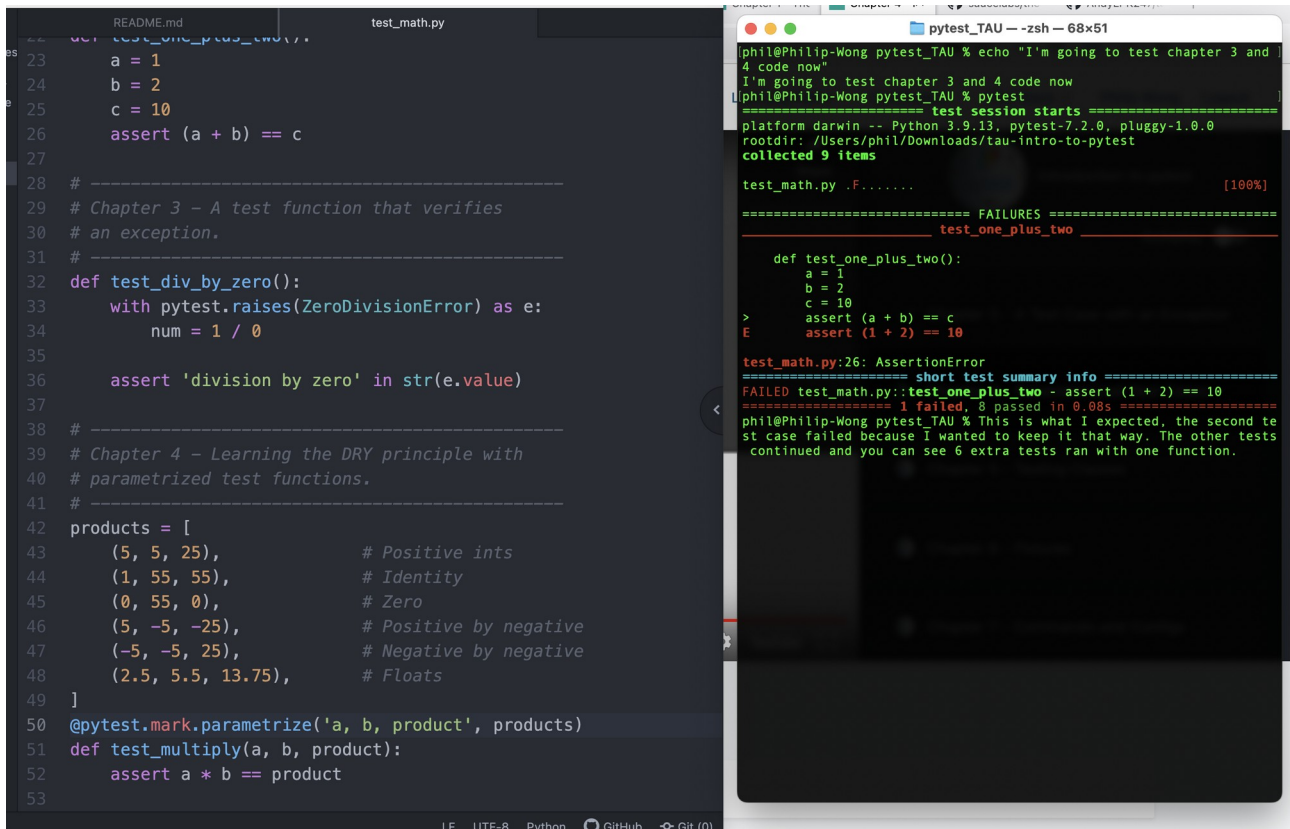
  def test_one_plus_two():
      a = 1
      b = 2
      c = 10
>     assert (a + b) == c
E       assert (1 + 2) == 10

test_math.py:15: AssertionError
===== short test summary info =====
FAILED test_math.py::test_one_plus_two - assert (1 + 2) == 10
===== 1 failed, 1 passed in 0.05s =====
[phil@Philip-Wong pytest_TAU % echo "Well that failed didn't it? Let's
change that to 3"
Well that failed didn't it? Let's change that to 3
[phil@Philip-Wong pytest_TAU % pytest
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0
rootdir: /Users/phil/Downloads/tau-intro-to-pytest
collected 2 items

test_math.py ..                                     [100%]

===== 2 passed in 0.02s =====
phil@Philip-Wong pytest_TAU %
```

Chapter 3 and 4



The screenshot shows a code editor with two files: `test_math.py` and `test_accump.py`. The `test_math.py` file contains the following code:

```
23 a = 1
24 b = 2
25 c = 10
26 assert (a + b) == c
27
28 # -----
29 # Chapter 3 - A test function that verifies
30 # an exception.
31 # -----
32 def test_div_by_zero():
33     with pytest.raises(ZeroDivisionError) as e:
34         num = 1 / 0
35
36     assert 'division by zero' in str(e.value)
37
38 # -----
39 # Chapter 4 - Learning the DRY principle with
40 # parametrized test functions.
41 # -----
42 products = [
43     (5, 5, 25),          # Positive ints
44     (1, 55, 55),         # Identity
45     (0, 55, 0),          # Zero
46     (5, -5, -25),        # Positive by negative
47     (-5, -5, 25),        # Negative by negative
48     (2.5, 5.5, 13.75),   # Floats
49 ]
50 @pytest.mark.parametrize('a, b, product', products)
51 def test_multiply(a, b, product):
52     assert a * b == product
53
```

The terminal window shows the output of the pytest command:

```
phil@Philip-Wong pytest_TAU % echo "I'm going to test chapter 3 and
4 code now"
I'm going to test chapter 3 and 4 code now
phil@Philip-Wong pytest_TAU % pytest
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0
rootdir: /Users/phil/Downloads/tau-intro-to-pytest
collected 9 items

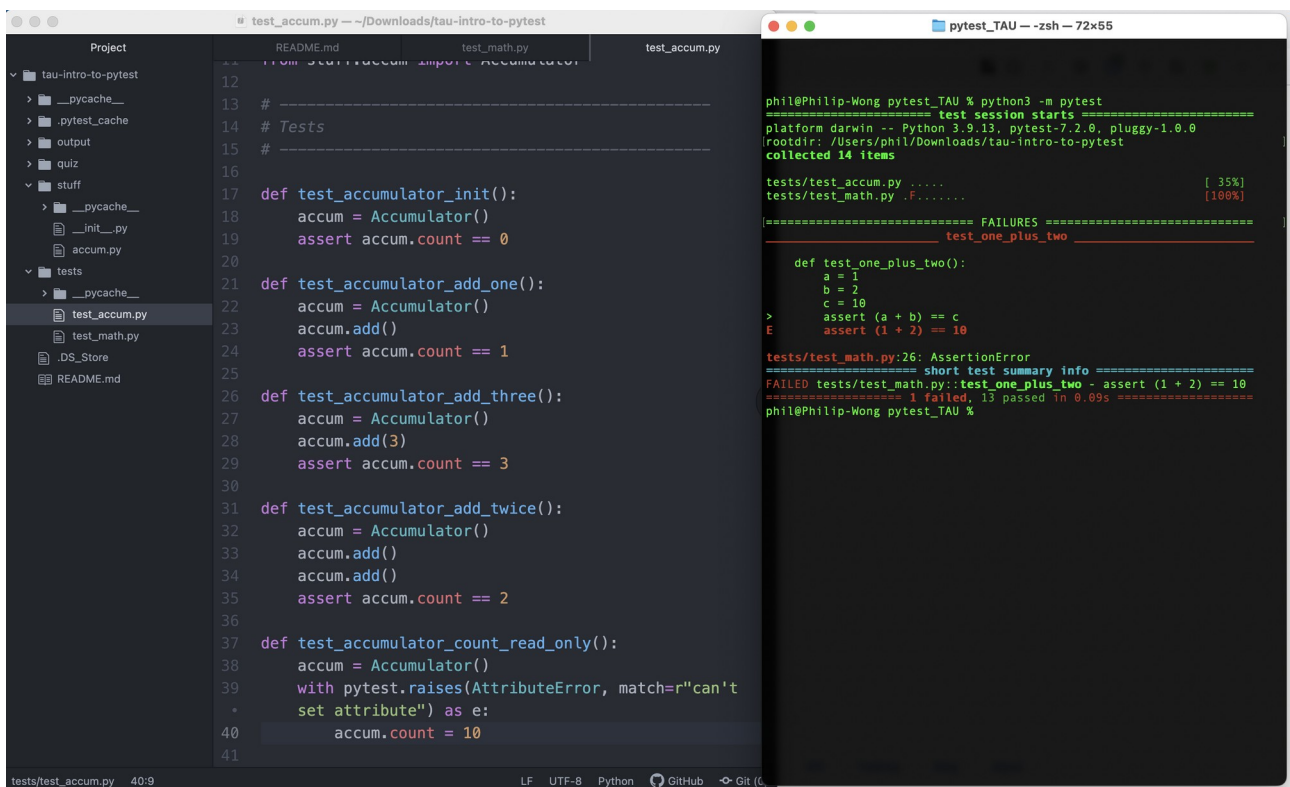
test_math.py .F..... [100%]

===== FAILURES =====
test_one_plus_two

def test_one_plus_two():
    a = 1
    b = 2
    c = 10
> assert (a + b) == c
E assert (1 + 2) == 10

test_math.py:26: AssertionError
===== short test summary info =====
FAILED test_math.py::test_one_plus_two - assert (1 + 2) == 10
===== 1 failed, 8 passed in 0.08s =====
phil@Philip-Wong pytest_TAU % This is what I expected, the second te
st case failed because I wanted to keep it that way. The other tests
continued and you can see 6 extra tests ran with one function.
```

Chapter 5



The screenshot shows a code editor with two files: `test_math.py` and `test_accump.py`. The `test_accump.py` file contains the following code:

```
12 from test_accum import Accumulator
13 # -----
14 # Tests
15 # -----
16
17 def test_accumulator_init():
18     accum = Accumulator()
19     assert accum.count == 0
20
21 def test_accumulator_add_one():
22     accum = Accumulator()
23     accum.add()
24     assert accum.count == 1
25
26 def test_accumulator_add_three():
27     accum = Accumulator()
28     accum.add(3)
29     assert accum.count == 3
30
31 def test_accumulator_add_twice():
32     accum = Accumulator()
33     accum.add()
34     accum.add()
35     assert accum.count == 2
36
37 def test_accumulator_count_read_only():
38     accum = Accumulator()
39     with pytest.raises(AttributeError, match=r"can't
40         set attribute") as e:
41         accum.count = 10
42
```

The terminal window shows the output of the pytest command:

```
phil@Philip-Wong pytest_TAU % python3 -m pytest
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0
rootdir: /Users/phil/Downloads/tau-intro-to-pytest
collected 14 items

tests/test_accump.py ..... [ 35%]
tests/test_math.py .F..... [100%]

===== FAILURES =====
test_one_plus_two

def test_one_plus_two():
    a = 1
    b = 2
    c = 10
> assert (a + b) == c
E assert (1 + 2) == 10

tests/test_math.py:26: AssertionError
===== short test summary info =====
FAILED tests/test_math.py::test_one_plus_two - assert (1 + 2) == 10
===== 1 failed, 13 passed in 0.09s =====
phil@Philip-Wong pytest_TAU %
```

Chapter 6

The image shows a code editor with a file explorer on the left and a terminal window on the right. The file explorer shows a project named 'tau-intro-to-pytest' with subdirectories like 'stuff' and 'tests'. The 'tests' directory contains files like 'test_accum_fixture_ver.py', 'test_accum.py', and 'test_math.py'. The code editor displays the content of 'test_accum_fixture_ver.py', which defines an accumulator fixture and several tests. The terminal window shows the output of running 'python3 -m pytest', displaying a summary of 18 tests passed and 1 failed, along with a detailed failure report for 'test_one_plus_two'.

```
Project
└─ tau-intro-to-pytest
  └─ stuff
    └─ tests
      └─ test_accum_fixture_ver.py
         test_accum.py
         test_math.py
         .DS_Store
         README.md
```

```
17 #
18
19 @pytest.fixture
20 def accum():
21     return Accumulator()
22
23 #
24 # Tests
25 #
26
27 def test_accumulator_init(accum):
28     assert accum.count == 0
29
30 def test_accumulator_add_one(accum):
31     accum.add()
32     assert accum.count == 1
33
34 def test_accumulator_add_three(accum):
35     accum.add(3)
36     assert accum.count == 3
37
38 def test_accumulator_add_twice(accum):
39     accum.add()
40     accum.add()
41     assert accum.count == 2
42
43 def test_accumulator_count_read_only(accum):
44     with pytest.raises(AttributeError, match=r"can't
45         *
46         set attribute") as e:
47         accum.count = 10
```

```
phil@Philip-Wong pytest_TAU % python3 -m pytest
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0
rootdir: /Users/phil/Downloads/tau-intro-to-pytest
collected 19 items

tests/test_accum.py ..... [ 26%]
tests/test_accum_fixture_ver.py ..... [ 52%]
tests/test_math.py .F..... [100%]

===== FAILURES =====
test_one_plus_two

def test_one_plus_two():
    a = 1
    b = 2
    c = 10
    > assert (a + b) == c
    E assert (1 + 2) == 10

tests/test_math.py:26: AssertionError
===== short test summary info =====
FAILED tests/test_math.py::test_one_plus_two - assert (1 + 2) == 10
===== 1 failed, 18 passed in 0.12s =====
phil@Philip-Wong pytest_TAU %
```

Chapter 7

```
pytest_TAU — -zsh — 72x53
phil@Philip-Wong pytest_TAU % python3 -m pytest --quiet
.....F..... [100%]
===== FAILURES =====
_____ test_one_plus_two _____

    def test_one_plus_two():
        a = 1
        b = 2
        c = 10
>       assert (a + b) == c
E       assert (1 + 2) == 10

tests/test_math.py:26: AssertionError
===== short test summary info =====
FAILED tests/test_math.py::test_one_plus_two - assert (1 + 2) == 10
1 failed, 18 passed in 0.10s
phil@Philip-Wong pytest_TAU %
```



```

pytest_TAU — -zsh — 72x53
[phil@Philip-Wong pytest_TAU % python3 -m pytest --verbose
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0 -- /usr/local/opt/python@3.9/bin/python3.9
cachedir: .pytest_cache
rootdir: /Users/phil/Downloads/tau-intro-to-pytest
collected 19 items

tests/test_accum.py::test_accumulator_init PASSED [ 5%]
tests/test_accum.py::test_accumulator_add_one PASSED [ 10%]
tests/test_accum.py::test_accumulator_add_three PASSED [ 15%]
tests/test_accum.py::test_accumulator_add_twice PASSED [ 21%]
tests/test_accum.py::test_accumulator_count_read_only PASSED [ 26%]
tests/test_accum_fixture_ver.py::test_accumulator_init PASSED [ 31%]
tests/test_accum_fixture_ver.py::test_accumulator_add_one PASSED [ 36%]
tests/test_accum_fixture_ver.py::test_accumulator_add_three PASSED [ 42%]
]
tests/test_accum_fixture_ver.py::test_accumulator_add_twice PASSED [ 47%]
]
tests/test_accum_fixture_ver.py::test_accumulator_count_read_only PASSED [ 52%]
tests/test_math.py::test_one_plus_one PASSED [ 57%]
tests/test_math.py::test_one_plus_two FAILED [ 63%]
tests/test_math.py::test_div_by_zero PASSED [ 68%]
tests/test_math.py::test_multiply[5-5-25] PASSED [ 73%]
tests/test_math.py::test_multiply[1-55-55] PASSED [ 78%]
tests/test_math.py::test_multiply[0-55-0] PASSED [ 84%]
tests/test_math.py::test_multiply[5--5--25] PASSED [ 89%]
tests/test_math.py::test_multiply[-5--5-25] PASSED [ 94%]
tests/test_math.py::test_multiply[2.5-5.5-13.75] PASSED [100%]

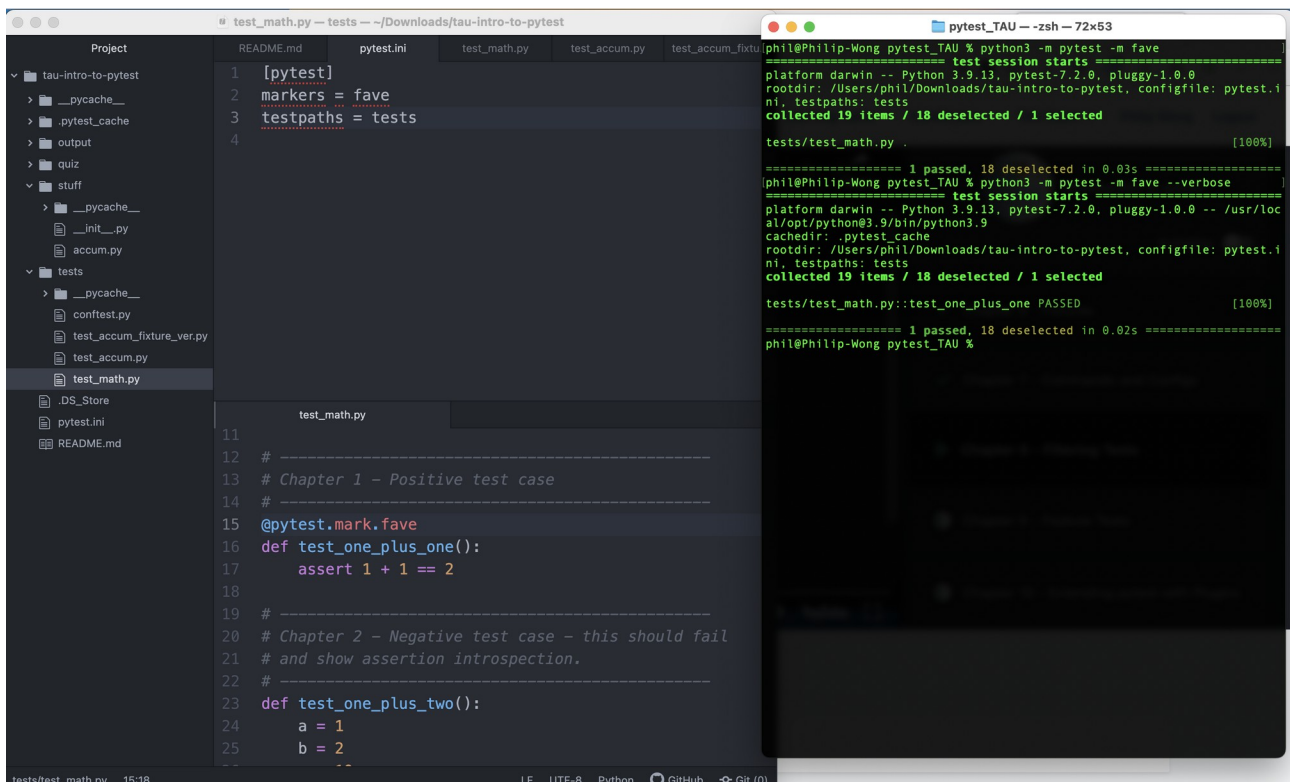
===== FAILURES =====
_____ test_one_plus_two _____

    def test_one_plus_two():
        a = 1
        b = 2
        c = 10
>       assert (a + b) == c
E       assert (1 + 2) == 10

tests/test_math.py:26: AssertionError
===== short test summary info =====
FAILED tests/test_math.py::test_one_plus_two - assert (1 + 2) == 10
===== 1 failed, 18 passed in 0.12s =====
phil@Philip-Wong pytest_TAU %

```

Chapter 8



The screenshot shows an IDE with a project named 'tau-intro-to-pytest'. The file explorer on the left shows the project structure, including a 'tests' directory. The main editor window displays the 'test_math.py' file, which contains a pytest marker 'fave' and two test functions: 'test_one_plus_one()' and 'test_one_plus_two()'. The 'test_one_plus_one()' function asserts that 1 + 1 equals 2. The 'test_one_plus_two()' function is partially visible. A terminal window on the right shows the output of running 'pytest -m fave' and 'pytest -m fave --verbose'. Both commands result in 1 passed test and 18 deselected tests. The terminal output also shows the pytest session starting and the configuration file used.

```
Project
└─ tau-intro-to-pytest
   └─ __pycache__
   └─ .pytest_cache
   └─ output
   └─ quiz
   └─ stuff
      └─ __pycache__
      └─ __init__.py
      └─ accum.py
      └─ tests
         └─ __pycache__
         └─ conftest.py
         └─ test_accum_fixture_ver.py
         └─ test_accum.py
         └─ test_math.py
      └─ .DS_Store
      └─ pytest.ini
      └─ README.md

test_math.py
1  [pytest]
2  markers = fave
3  testpaths = tests
4

test_math.py
11
12 # -----
13 # Chapter 1 - Positive test case
14 # -----
15 @pytest.mark.fave
16 def test_one_plus_one():
17     assert 1 + 1 == 2
18
19 # -----
20 # Chapter 2 - Negative test case - this should fail
21 # and show assertion introspection.
22 # -----
23 def test_one_plus_two():
24     a = 1
25     b = 2
```

```
phil@Philip-Wong pytest_TAU % python3 -m pytest -m fave
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0
rootdir: /Users/phil/Downloads/tau-intro-to-pytest, configfile: pytest.ini
ni, testpaths: tests
collected 19 items / 18 deselected / 1 selected

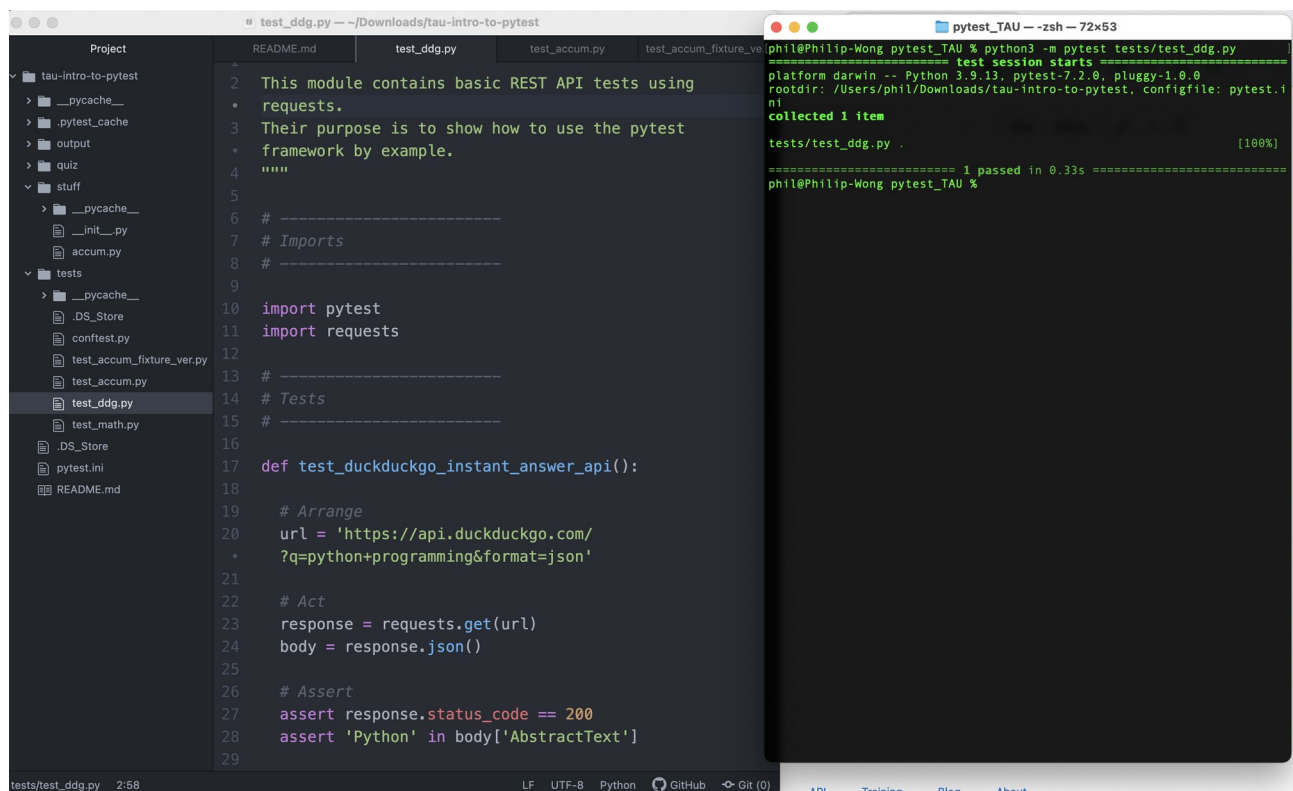
tests/test_math.py . [100%]

===== 1 passed, 18 deselected in 0.03s =====
phil@Philip-Wong pytest_TAU % python3 -m pytest -m fave --verbose
===== test session starts =====
platform darwin -- Python 3.9.13, pytest-7.2.0, pluggy-1.0.0 -- /usr/local/opt/python@3.9/bin/python3.9
cachedir: .pytest_cache
rootdir: /Users/phil/Downloads/tau-intro-to-pytest, configfile: pytest.ini
ni, testpaths: tests
collected 19 items / 18 deselected / 1 selected

tests/test_math.py::test_one_plus_one PASSED [100%]

===== 1 passed, 18 deselected in 0.02s =====
phil@Philip-Wong pytest_TAU %
```

Chapter 9



Chapter 10

