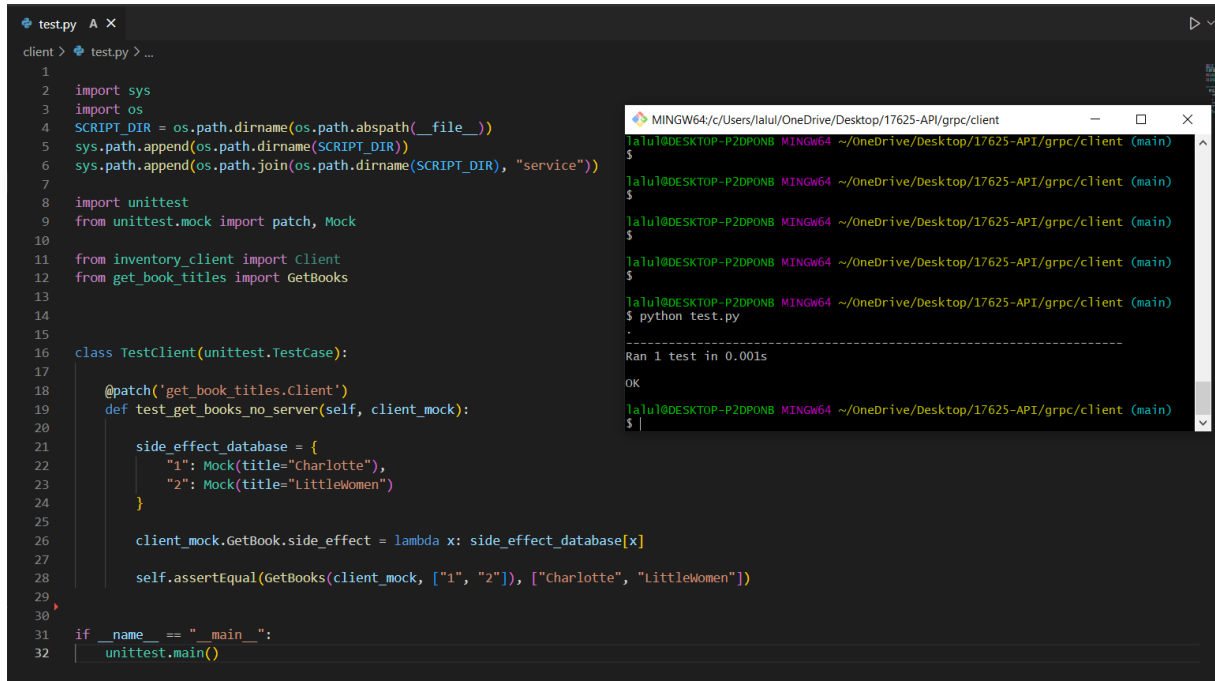


link to the repository: <https://github.com/Angelalalula/API-gRPC>

screenshots of test results step 5

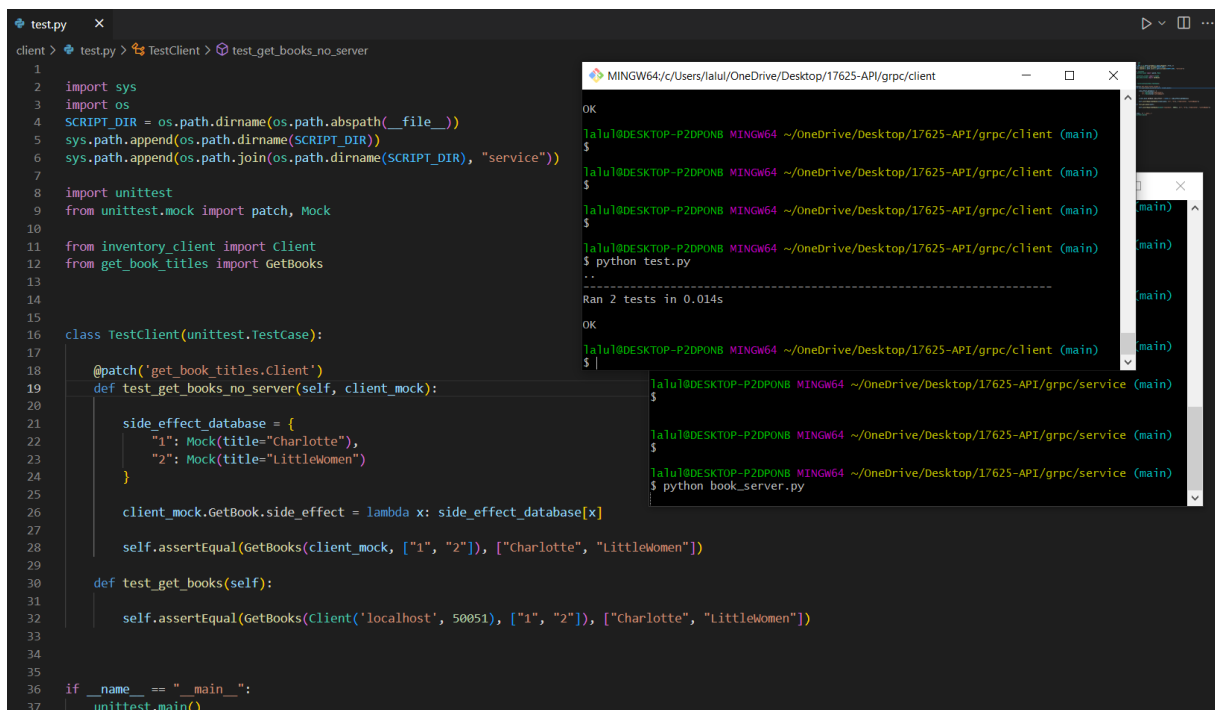


The screenshot shows a code editor with a Python test file and a terminal window. The code defines a test class `TestClient` with a test `test_get_books_no_server` that uses `unittest.mock` to patch the `get_book_titles.Client` and verify the `GetBooks` method. The terminal shows the command `python test.py` being executed, resulting in `Ran 1 test in 0.001s` and `OK`.

```
1 import sys
2 import os
3 SCRIPT_DIR = os.path.dirname(os.path.abspath(__file__))
4 sys.path.append(os.path.dirname(SCRIPT_DIR))
5 sys.path.append(os.path.join(os.path.dirname(SCRIPT_DIR), "service"))
6
7 import unittest
8 from unittest.mock import patch, Mock
9
10 from inventory_client import Client
11 from get_book_titles import GetBooks
12
13 class TestClient(unittest.TestCase):
14
15     @patch('get_book_titles.Client')
16     def test_get_books_no_server(self, client_mock):
17
18         side_effect_database = {
19             "1": Mock(title="Charlotte"),
20             "2": Mock(title="LittleWomen")
21         }
22
23         client_mock.GetBook.side_effect = lambda x: side_effect_database[x]
24
25         self.assertEqual(GetBooks(client_mock, ["1", "2"]), ["Charlotte", "LittleWomen"])
26
27 if __name__ == "__main__":
28     unittest.main()
```

```
MINGW64/c/Users/lalul/OneDrive/Desktop/17625-API/grpc/client
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$ python test.py
.
-----
Ran 1 test in 0.001s
OK
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$ |
```

screenshots of test results step 6



The screenshot shows the same code editor as before, but with an additional test `test_get_books` added to the `TestClient` class. This test calls the `GetBooks` method with a `Client` object and verifies the result. The terminal shows the command `python test.py` being executed, resulting in `Ran 2 tests in 0.014s` and `OK`.

```
1 import sys
2 import os
3 SCRIPT_DIR = os.path.dirname(os.path.abspath(__file__))
4 sys.path.append(os.path.dirname(SCRIPT_DIR))
5 sys.path.append(os.path.join(os.path.dirname(SCRIPT_DIR), "service"))
6
7 import unittest
8 from unittest.mock import patch, Mock
9
10 from inventory_client import Client
11 from get_book_titles import GetBooks
12
13 class TestClient(unittest.TestCase):
14
15     @patch('get_book_titles.Client')
16     def test_get_books_no_server(self, client_mock):
17
18         side_effect_database = {
19             "1": Mock(title="Charlotte"),
20             "2": Mock(title="LittleWomen")
21         }
22
23         client_mock.GetBook.side_effect = lambda x: side_effect_database[x]
24
25         self.assertEqual(GetBooks(client_mock, ["1", "2"]), ["Charlotte", "LittleWomen"])
26
27     def test_get_books(self):
28
29         self.assertEqual(GetBooks(Client('localhost', 50051), ["1", "2"]), ["Charlotte", "LittleWomen"])
30
31 if __name__ == "__main__":
32     unittest.main()
```

```
MINGW64/c/Users/lalul/OneDrive/Desktop/17625-API/grpc/client
OK
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$ python test.py
..
-----
Ran 2 tests in 0.014s
OK
lalul@DESKTOP-P2DPONB MINGW64 ~/OneDrive/Desktop/17625-API/grpc/client (main)
$ |
```

