**Debug your Solidity tests**

**First failed test**

Let's continue writing tests for our FundMe contract. Let's test if the owner (which should be us) is recorded properly.

Add the following function to your testing file:

function testOwnerIsMsgSender() public {

assertEq(fundMe.i\_owner(), msg.sender);

}

Run it via forge test.

Output:

Ran 2 tests for test/FundMe.t.sol:FundMeTest

[PASS] testMinimumDollarIsFive() (gas: 5453)

[FAIL. Reason: assertion failed] testOwnerIsMsgSender() (gas: 22521)

Suite result: FAILED. 1 passed; 1 failed; 0 skipped; finished in 3.85ms (623.00µs CPU time)

Ran 1 test suite in 367.24ms (3.85ms CPU time): 1 tests passed, 1 failed, 0 skipped (2 total tests)

Failing tests:

Encountered 1 failing test in test/FundMe.t.sol:FundMeTest

[FAIL. Reason: assertion failed] testOwnerIsMsgSender() (gas: 22521)

So ... why did it fail? Didn't we call the new FundMe(); to deploy, making us the owner?

We can find the answer to these questions in various ways, in the last lesson we learned about console.log, let's add some console.logs to see more information about the two elements of the assertion.

function testOwnerIsMsgSender() public {

console.log(fundMe.i\_owner());

console.log(msg.sender);

assertEq(fundMe.i\_owner(), msg.sender);}

Let's run forge test -vv:

Ran 2 tests for test/FundMe.t.sol:FundMeTest

[PASS] testMinimumDollarIsFive() (gas: 5453)

[FAIL. Reason: assertion failed] testOwnerIsMsgSender() (gas: 26680)

Logs:

0x7FA9385bE102ac3EAc297483Dd6233D62b3e1496

0x1804c8AB1F12E6bbf3894d4083f33e07309d1f38

Error: a == b not satisfied [address]

Left: 0x7FA9385bE102ac3EAc297483Dd6233D62b3e1496

Right: 0x1804c8AB1F12E6bbf3894d4083f33e07309d1f38

Suite result: FAILED. 1 passed; 1 failed; 0 skipped; finished in 975.40µs (449.20µs CPU time)

Ran 1 test suite in 301.60ms (975.40µs CPU time): 1 tests passed, 1 failed, 0 skipped (2 total tests)

Failing tests:

Encountered 1 failing test in test/FundMe.t.sol:FundMeTest

[FAIL. Reason: assertion failed] testOwnerIsMsgSender() (gas: 26680)

Ok, so the addresses are different, but why?

Technically we are not the ones that deployed the FundMe contract. The FundMe contract was deployed by the setUp function, which is part of the FundMeTest contract. So, even though we are the ones who called setUp via forge test, the actual testing contract is the deployer of the FundMe contract.

To test the above let's tweak the testOwnerIsMsgSender function:

function testOwnerIsMsgSender() public {

assertEq(fundMe.i\_owner(), address(this));

}

Run forge test. It passes! Congratulations!

Feel free to try and write other tests!