STARKWARE

Protostar

Unit-testing with Cairo

StarkNet edu team

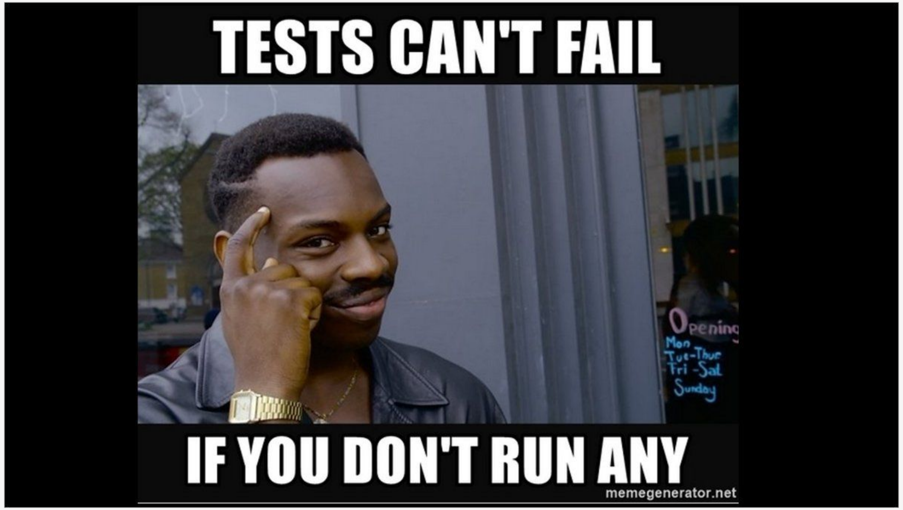
https://github.com/starknet-edu | @starkwareltd

September 2022

STARKWARE

Plan: Test, Deploy and Customize an ERC721

1. Code a custom ERC721
2. Unit testing with Protostar
3. Testing tips and cheatcodes



The sanity steps

Unit tests -> Devnet/Local -> Testnet -> Mainnet

Don't waste your time testing your StarkNet contract when the client has Starkscan, "Write contract" and a mouse.

Protostar is to StarkNet

Foundry/Truffle are for Ethereum

To what...

Code a custom ERC721

starknet-edu

ERC721 Automated workshop

Help?

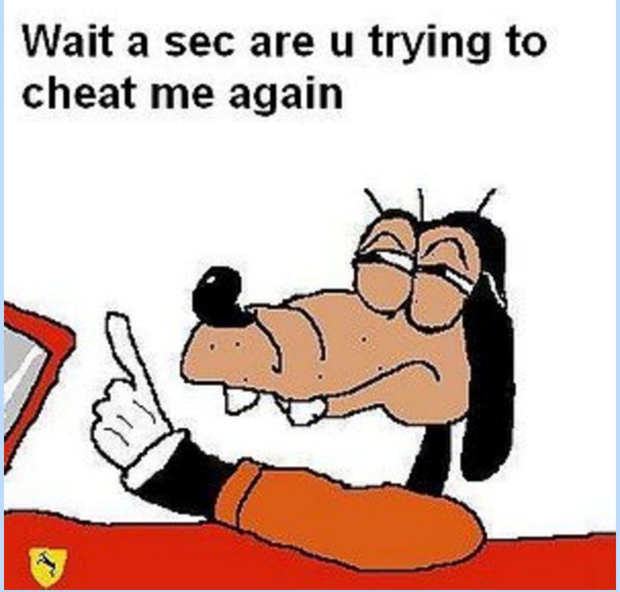
David´s tutorials

Testing

Cheatcodes

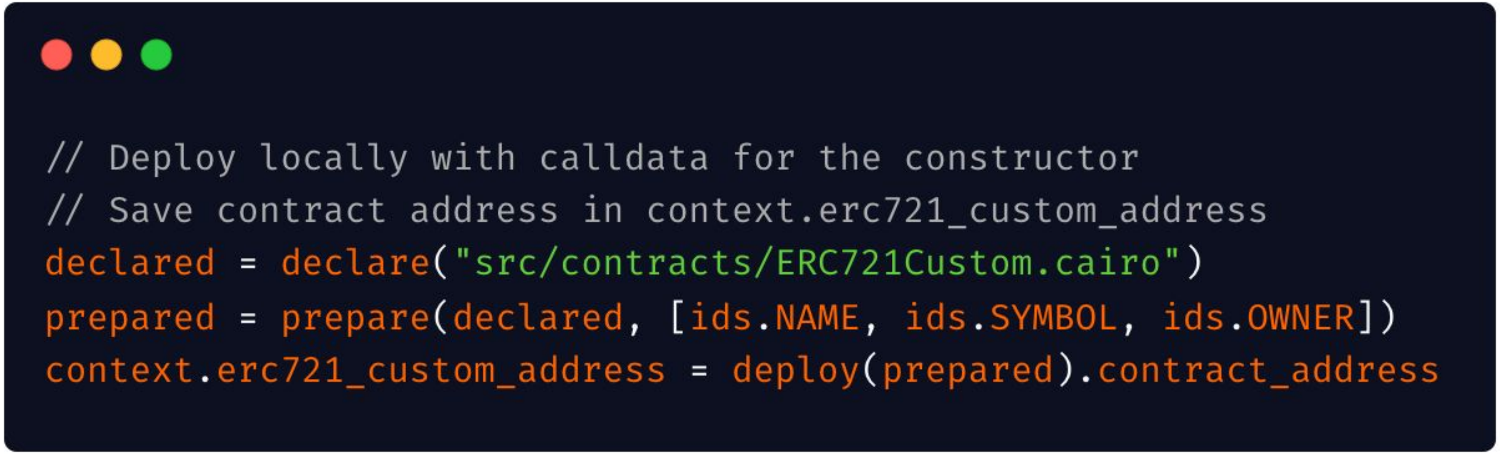
Inspired by **Foundry**

Fool me that you are manipulating the state of the blockchain without deploying it to any network.



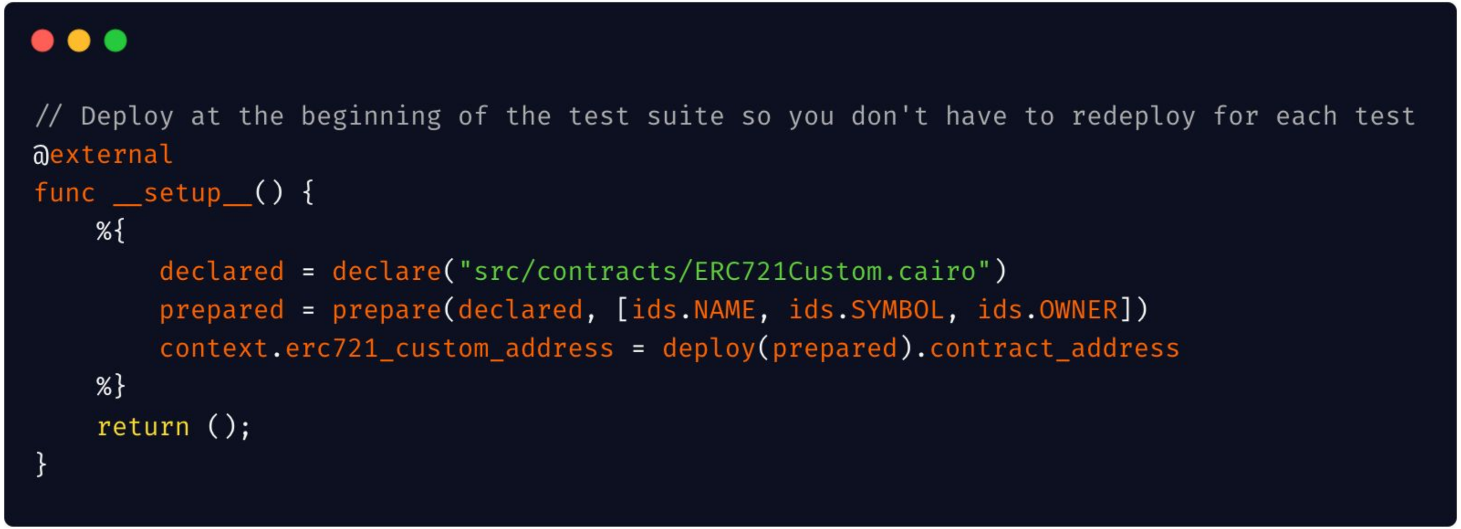
Simulate (cheat) deployment

declare(contract_path: str, *, config: Optional[CheatcodeNetworkConfig] = None) -> DeclaredContract



```
// Deploy locally with calldata for the constructor
// Save contract address in context.erc721_custom_address
declared = declare("src/contracts/ERC721Custom.cairo")
prepared = prepare(declared, [ids.NAME, ids.SYMBOL, ids.OWNER])
context.erc721_custom_address = deploy(prepared).contract_address
```

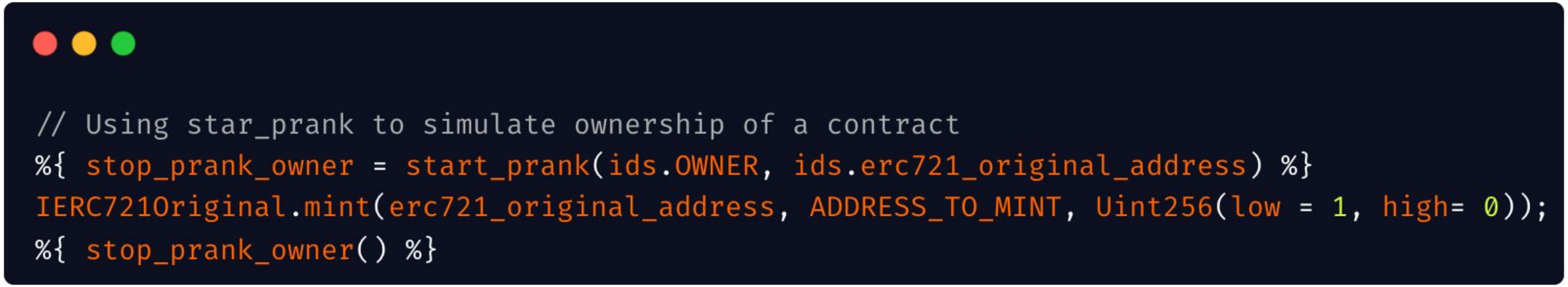
One cheat deploy to rule them all



```
// Deploy at the beginning of the test suite so you don't have to redeploy for each test
@external
func __setup__() {
    %{
        declared = declare("src/contracts/ERC721Custom.cairo")
        prepared = prepare(declared, [ids.NAME, ids.SYMBOL, ids.OWNER])
        context.erc721_custom_address = deploy(prepared).contract_address
    %}
    return ();
}
```

Simulate (cheat) ownership

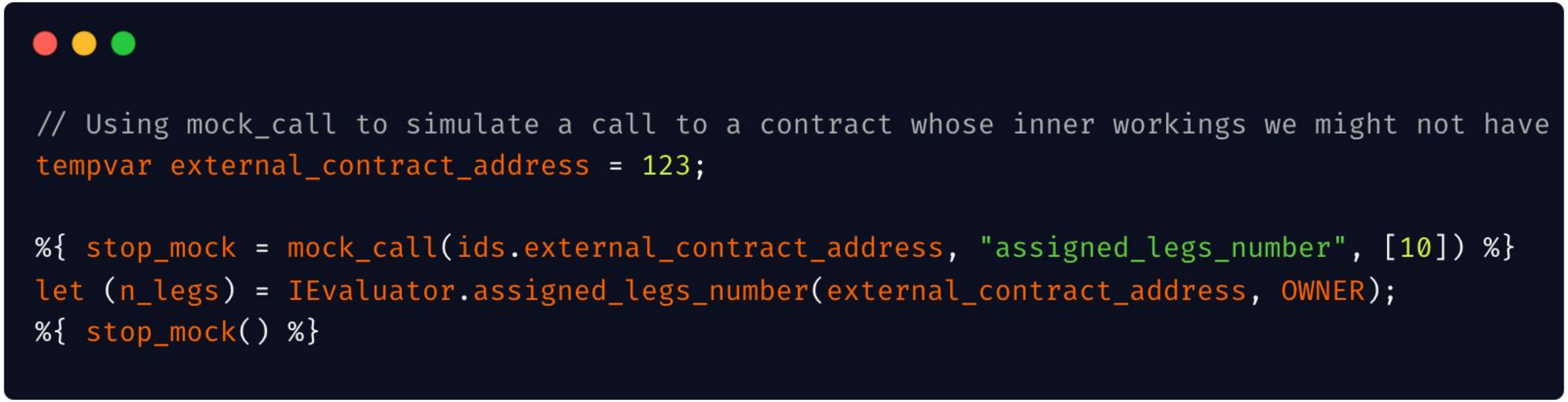
start_prank(caller_address: int, target_contract_address: Optional[int] = None)



```
// Using star_prank to simulate ownership of a contract
%{ stop_prank_owner = start_prank(ids.OWNER, ids.erc721_original_address) %}
IERC721Original.mint(erc721_original_address, ADDRESS_TO_MINT, Uint256(low = 1, high= 0));
%{ stop_prank_owner() %}
```

Simulate a call

mock_call(contract_address: int, fn_name: str, ret_data: Union[List[int], Dict])



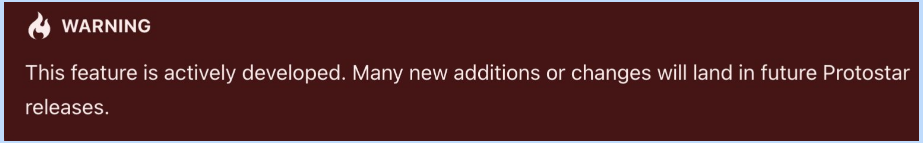
```
// Using mock_call to simulate a call to a contract whose inner workings we might not have
tempvar external_contract_address = 123;

%{ stop_mock = mock_call(ids.external_contract_address, "assigned_legs_number", [10]) %}
let (n_legs) = IEvaluator.assigned_legs_number(external_contract_address, OWNER);
%{ stop_mock() %}
```

Next step: Migrations

Migrations

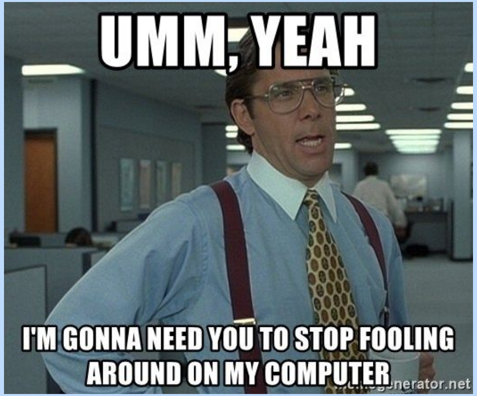
Inspired by **Truffle**

Actually deploy and manage contracts in an StarkNet network.



WARNING

This feature is actively developed. Many new additions or changes will land in future Protostar releases.



Questions?

STARKWARE

Thanks!

StarkNet edu team

@starkwareltd

Omar Espejel | @espejelomar