Flash Loans Exercise 1

Intro

In this exercise, your goal is to implement an ETH Pool that enables flash loans.

You will also implement 2 receivers contracts:

1. A "Receiver" contract which requests a flashloan and pays it back.
2. A "Greedy Receiver" contract which requests the loan and doesn't pay it back.

After implementing the contracts, you will have to test them and make sure that the logic works.

Accounts

- 0 - Deployer & Owner
- 1 - User

Tasks

Task 1

Implement the `Pool.sol` contract.

Complete the `flashLoan` function:

1. Revert the transaction in case there is not enough ETH in pool (Less than requested).
2. Call the `getEth` function on `msg.sender` with the requested amount of ETH.
3. Make sure that the ETH was paid back.

Task 2

Implement the `Receiver.sol` contract:

1. Complete the `flashLoan` function so it will request a flash loan from the pool.
2. Complete the `getEth` function so it sends back the ETH to the `Pool` contract.

Task 3

Implement the `GreedyReceiver.sol` contract:

1. Complete the `flashLoan` function so it will request a flash loan from the pool.
2. Complete the `getEth` function so it DOESN'T send back the ETH to the `Pool` contract.

Note: Should you consider anything else is needed to be implemented for the contracts to work, please proceed. The instructions given are not exhaustive.

Task 4

Complete the tests in the `tests.js` file.