**Fund Me Introduction**

**Introduction:**

In this section, we’ll create a \_decentralized crowdfunding\_ contract. The complete codebase is available in the Github repository (<https://github.com/Cyfrin/remix-fund-me-f23>).

**Overview:**

For this project, we will be using two contracts: `FundMe`, the main crowdfunding contract, and `PriceConverter`. They function much like Kickstarter, allowing users to send any native blockchain cryptocurrency. They also enable the owner of the contract to withdraw all the funds collected. We will then deploy these contracts on a testnet.

**Note:** Use testnet sparingly. Limiting testnet transactions helps prevent network congestion, ensuring a smoother testing experience for everyone.

**Fund and withdraw:**

Once `FundMe` is deployed on Remix, you’ll notice a set of functions, including a new red button labelled `fund`, indicating that the function is payable. A payable function allows you to send native blockchain currency (e.g., Ethereum, Polygon, Avalanche) to the contract.

We’ll additionally indicate a minimum USD amount to send to the contract when the function `fund` is called. To transfer funds to the `FundMe` contract, you can navigate to the value section of the Remix deployment tab, enter a value (e.g. 0.1 ether) then hit `fund`. A MetaMask transaction confirmation will appear, and the contract balance will remain zero until the transaction is finalized. Once completed, the contract balance will be updated to reflect the transferred amount.

The contract owner can then `withdraw` the funds. In this case, since we own the contract, the balance will be removed from the contract’s balance and transferred to our wallet.

**Conclusion:**

These 25 lessons will guide you step-by-step through the implementation of a crowdfunding contract, that supports cryptocurrency contributions and owner withdrawals.