

Coding Exercise: Integration with Convex or Yearn Protocols

Objective: Integrate with the Convex or Yearn protocols using Solidity to facilitate Deposit, Withdraw, and Manage User Rewards.

Users should be able to Withdraw their Deposited amount + rewards.

Background:

Convex Finance and **Yearn Finance** are decentralized finance (DeFi) platforms that allow users to deposit assets and earn yield. These platforms have smart contracts that enable developers to integrate and interact with their services.

In this exercise, you must write a Solidity smart contract that interfaces with either of these platforms. While you can choose either protocol for the integration, there are additional points if you decide on Convex.

Requirements:

Choose a Protocol:

- Convex (preferred for additional points)
- Yearn

Smart Contract Functions:

- Deposit: Accepts an amount and deposits it into the chosen protocol.
- Withdraw: Accepts an amount and removes it from the selected protocol.

Additional Notes:

- Ensure that the contract handles potential errors and edge cases gracefully.
- Consider gas efficiency when writing your functions.
- Ensure that the smart contract is secure and follows best practices.
- You can make use of interfaces or import external libraries if needed.
- Foundry Framework required.

Evaluation Criteria:

1. Correctness of the implemented functions.
2. Smart contract security and handling of potential vulnerabilities.
3. Gas efficiency of the contract functions.
4. Code readability.
5. Proper handling of edge cases.

Submission:

Please provide the following upon completion:

1. Add to your GitHub repo the users fepvenancio and 0xadrii
2. We require the code of the contract and the forge tests.

Deadline:

You have three days from receiving this exercise to submit your solution.