

한 번에 끝내는 블록체인 개발 A to Z

Chapter 3

Defi 기초 컨셉 구현

Chapter 3

Defi 기초 컨셉 구현

SmartContract 개발 및 테스트 환경

Goal

앞선 여러 강의에서 Solidity 개발에 대한 환경설정을 진행했을 것이다.

이번 강의에서는 이번 ‘Defi 기초 컨셉 구현’ 챕터에서 사용할 환경에 대한 설명을 진행한다.

본인의 현재 환경에서 개발을 원하거나 어떤 환경에서도 상관없다고 한다면 이번 강의는 건너 뛰어도 괜찮다.

개발 및 테스트 환경

- Solidity 개발 IDE: VisualStudio Code
- VisualStudio Code Solidity Extension: Solidity by Juan Blanco
- Solidity 개발, 배포 및 테스트 툴: Hardhat, Typechain, Waffle
 - Typechain, Waffle 모두 Hardhat에서 제공하는 툴이다.
 - truffle 도 많이 사용되지만 다양한 기능들을 제공하는 HardHat도 최근에 많이 사용되는 추세이다.
- Ethereum RPC Node: All That Node
 - 대한민국 기업의 서비스로 다양한 블록체인 네트워크의 RPC 노드, faucet 제공

① Remix

Remix는 규모가 큰 프로젝트 보다는 간단한 SmartContract를 개발 하고 테스트 하는 데 많이 사용이 된다. 물론 제공하는 기능이 많기 때문에 큰 규모의 프로젝트도 개발이 가능하다. 개발을 시작 할 때 특별히 설정해줘야 하는 부분이 별로 없다. 그리고 자체 VM으로 테스트를 매우 쉽고 간단하게 할 수 있는 것이 장점이기도 하다.

② Truffle

Remix외에 가장 많이 사용되는 툴일 것이다. solidity 파일 컴파일, 배포, 테스트 코드 작성까지 모든 기능을 제공하고 있다. Truffle 생태계의 ganache는 프라이빗 노드를 쉽게 구성할 수 있어서 dapp 개발의 테스트에도 많이 사용이 된다.

③ Hardhat

Truffle보다 더 개선된 기능들을 많이 제공하고 있다. 가장 사람들이 좋아하는 기능은 console.log를 사용할 수 있는 것이다. 뿐만 아니라 etherscan의 verify 자동화 기능, TypeScript 지원 등으로 최근에 많이 사용 되고 있다.

다음 강의

다음 강의에서 본격적으로 디파이 구현에 들어가본다.
유니스왑 v1 소스코드를 활용하여 기본적으로 디파이가 어떻게 동작하는지 알 수 있다.
(약 3-4시간 예정)