

내부 키워드 정리

≡ 태그

- 태그: 작성하려는 글에 맞거나 연관되어 있는 태그를 선택합니다.
- 상태: 해당 글의 상태를 의미합니다.
 - 글을 쓸 예정으로 등록했다면 **작성 예정**, 글을 쓰는 중이거나 아직 완료한 것 같지 않다면 **작성 중**, 완료 후에는 **완료**로 표기합니다.

▲가독성을 위해 개요와 결론을 작성하시는 것을 추천하지만, 느낌에 따라 자유롭게 작성하시면 됩니다!

개요

이 과제를 하다가 너무 생소한 단어들을 많이 마주해서 정리하고자 이 문서를 작성하게 되었다.

본론

SmartContract

스마트 계약(smart contract) 또는 **스마트 컨트랙트**란 블록체인 기반으로 금융거래, 부동산 계약, 공증 등 다양한 형태의 계약을 체결하고 이행하는 것을 말한다.

Mint, Burn

OpenZeppelin에서 제공하는 IERC20 인터페이스에 맞게 사용할 때, `_mint`라는 메소드를 통해서 토큰을 발급하고 `_burn`이라는 메소드를 통해서 토큰을 삭제하게 된다.

`_mint`

```
_mint(address account, uint256 amount)
```

Creates amount tokens and assigns them to account, increasing the total supply.
Emits a transfer event `with` from `set` to the zero address.

Requirements

- `account` cannot be the zero address.

`_burn`

```
_burn(address account, uint256 amount)
```

Destroys amount tokens from account, reducing the total supply.
Emits a transfer event `with` to `set` to the zero address.

Requirements

- account cannot be the zero address.
- account must have at least amount tokens.

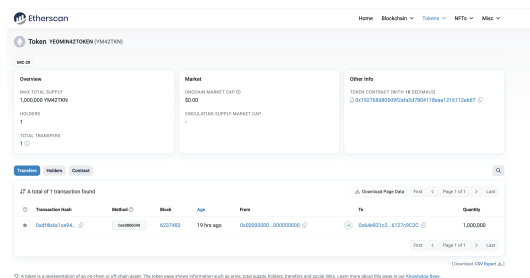
Ticker

Ticker

Ticker == Symbol

Ticker 자체는 주식시장에서 나온 단어이다.

두 단어는 코인 시장 내에선 동일한 의미로, 토큰 명의 축약형태라고 보면된다.(ex. BitCoin == BTC)



symbol

symbol() → string

Returns the symbol of the token, usually a shorter version of the name.

Payable

Solidity 언어에서 payable 키워드는 이더리움 플랫폼 위에서 이더(ether) 코인을 전송하는 스마트 컨트랙트 (smart contract)를 작성하기 위해서 반드시 사용해야한다.

다시 말해서 payable을 작성한 함수에서만 이더(ether)를 보낼 수 있고, payable을 작성하지 않은 함수에서는 이더(ether)를 보낼 수 없는 것이다.

따라서, 개발자가 스마트 컨트랙트 코드를 만들고, 이더리움넷(ex, Main Net, Test Net)에 배포(deploy)를 했다
면, 그리고 스마트 컨트랙트 외부에서 스마트 컨트랙트에 있는 함수들 중에서 코인을 이동(=전송)시키는 함수를 사
용하려 한다면, 해당 함수는 반드시 payable 키워드가 함께 작성된 함수이어야한다.

Transaction

트랜잭션(Transactions)은 외부 소유 계정(EOA)에서 생성되어 이더리움 블록체인에 기록된 서명된 메시지다. 트랜잭션을 통해서만 이더(ether)를 전송하거나, 이더리움 가상 머신(EVM)에 있는 컨트랙트를 실행할 수 있다.

Gas

Gas란?

이더와는 별도의 가상화폐로, 이더리움에서 사용되는 모든 트랜잭션, 스마트 컨트랙트는 gas를 사용하는데, gas는 이더리움의 수수료이고 송금을 할 때나 스마트 컨트랙트의 코드를 실행할 때 사용된다.

이더리움에서 transaction당 필요한 gas량

이더리움의 수 당 가스 소비량 * 가스 가격 을 지불해야한다.

가스 가격이 높아질 수록 거래의 우선순위가 올라간다.

Deploy

생성된 스마트 컨트랙트를 배포한다는 뜻이다.

Balance

해당 어카운트(컨트랙트)가 소유하고 있는 Wei양이다.

(1 ETH= $10^{(-18)}$ Wei, 10^{18} Wei = 1 ETH)

Decimals

소수단위의 거래를 하기위해서 만들어진 체계로 기본적으로 ERC20에서는 10^{18} 로 되어있는데 이게 ETH 숫자 체계 중 가장 작은 단위인 Wei에 해당하는 단위이기 때문이다.

Explorer

각 스마트 컨트랙트들의 상태들을 조회할 수 있는 툴을 일컫는다. 거래내역, 소유권 이전 등 다양한 내용들을 조회할 수 있다. 나는 VM환경에서 Ether 혹은 Sepolia를 사용했기 때문에 EtherScan이라는 툴을 통해서 확인하였다.

MultiSig

각 트랜잭션에 대해서 지정된 사용자에게 지정된 횟수 만큼 허용이 되어야만 실행되는 체계를 말한다.

예를 들어, A, B, C 3명의 어카운트 유저에 대해서 2명의 허용이 필요한 체계 라고 가정할 때

임의의 유저 D에게 1Ether를 보내는 트랜잭션을 진행할 때 최소 2명에게 Confirm을 받아야 이 트랜잭션이 진행된다는 뜻이다.

결론

참고

SmartContract

스마트 계약

스마트 계약(smart contract) 또는 스마트 컨트랙트란 블록체인 기반으로 금융거래, 부동산 계약, 공증 등 다양한 형태의 계약을 체결하고 이행하는 것을 말한다. 블록체인 2.0이라고도 한다.[1]

℥ https://ko.wikipedia.org/wiki/스마트_계약

스마트 컨트랙트란 무엇인가? - 업비트 투자자보호센터

스마트 컨트랙트란? 스마트 컨트랙트(smart contract)는 블록체인이 1세대에서 2세대로 넘어갈 수 있게 되는 가장 중요한 계기 중 하나로 블록체인 기술을 활용해 제3의 인증기관 없이 개인 간 계약이 이루어질 수 있도록 하는 기술입니다. 더욱 자세히는 계약상의 급부와 반대급부를

<https://m.upbitcare.com/academy/education/blockchain/70>



Ticker

<https://sepolia.etherscan.io/token/0x192768d805d9f2afa2d7804118eaa1216112eb67>

What Is a Stock Ticker? Definition, How It Works, and Origins

A stock ticker is a report of the price for certain securities, updated continuously throughout the trading session by the various stock exchanges.

<https://www.investopedia.com/ask/answers/12/what-is-a-stock-ticker.asp>



Payable

solidity - payable (1) 개념

Payable (1) payable concept Solidity 언어는 코인(coin) 혹은 토큰(token)이라는 가상화폐를 다루는 언어이다. 다른 언어들(ex, C, Java, HTML)은 오로지 프로그램을 만들기 위한 언어로 탄생했지만, Solidity는 가상화폐라는 돈을 다루기 위한 언어로 탄생했기 때문에, 특히

<https://caileb.tistory.com/147>

```
// address payable
address payable owner;
function func(address payable to) public {
}

// function payable
constructor() payable public {
}
function func() payable external {
```

What are Payable Functions in Solidity?

Learn about payable functions in Solidity, their importance in handling Ether deposits, and how to create and use them in smart contracts.

<https://docs.alchemy.com/docs/solidity-payable-functions>

Transaction

[블록체인] 이더리움(Ethereum) 공부 #3 - 트랜잭션과 서명 — Steemit

안녕하세요. @anpigon입니다. 마스터 이더리움(Mastering Ethereum) 책을 보면서 정리한 글입니다. 아직 전체 내용을 다 보지는 못하였습니다. 하지만 궁금한 사항을 댓글로...

<https://steemit.com/busy/@anpigon/ethereum-3>



Gas


2. 이더리움 solidity 가스(gas) 개념 · Codinfox Lanyon

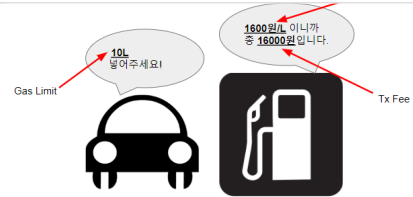
Contact:

<https://wkimdev.github.io/blockchain/2018/11/23/eth-solidity-gas/>


Ethereum Gas에 대한 이해

1. 요약

 <https://medium.com/tomak/ethereum-gas에-대한-이해-9fd4a7d169ac>




Solidity by Example

 <https://solidity-by-example.org/gas/>

Balance

이더리움 코어의 데이터 계층 - 어카운트(account)

어카운트 이더리움 플랫폼에서 어카운트는 모든 트랜잭션의 실행 주체로 가장 기본적인 단위입니다. 이더리움에서는 다음과 같이 2개의 어카운트 타입을 가지고 있습니다. 외부소유 어카운트 (EOA : Externally Owned Account) - 일반적으로 거래에 사용되는 사용자의 지갑주소를

 <https://ihpark92.tistory.com/44>

Externally owned account




Contract account

<code>
<code>
<code>

이더리움은 어떻게 동작할까? (번역본 Vol.2)

먼저, 블록체인(Not 이더리움)을 한 문장으로 정의하자면 아래와 같습니다. "암호학적으로 안전하고, 모...

 <https://blog.naver.com/hersheythings/221977798438>



Decimal

ERC-20 토큰: 네트워크의 정의와 기능들의 특징 - 코이니셜


핵심 요약 내용

 <https://coinicial.com/glossary/erc-20/>

Balance

How to Get ERC-20 Token Balance at a Given Block


Learn how to get the balance of a single token being held by a specific contract address at a given block or point in time.

 <https://docs.alchemy.com/docs/how-to-get-erc-20-token-balance-at-a-given-block>

Decimal

Why is ether divisible to 18 decimal places?

1 ether is 10^{18} wei, the smallest denomination. Why does ether have such a high degree of divisibility?


 <https://ethereum.stackexchange.com/questions/363/why-is-ether-divisible-to-18-decimal-places>



MultiSig


What are multi-signature contracts?

Multi-signature contracts require multiple signatures for transactions, providing security against lost or compromised keys. Gnosis Safe is a multi-signature smart contract deployer on Ethereum.

 <https://docs.alchemy.com/docs/multi-sig-contracts>


MULTISIG WALLET EXPLAINED (with code example)

The idea of Multi-Signature wallets appeared in 2012, with the first ones being created and implemented a year later for the Bitcoin...

 <https://medium.com/coinmonks/multisig-wallet-explained-with-code-example-6fb4a663ab29>



Solidity by Example

 <https://solidity-by-example.org/app/multi-sig-wallet/>