NFT 실습

NFT 실습

간단한 NFT 발행 및 전송 실습

컨트랙트 소유자 / 디지털 작품 생산자

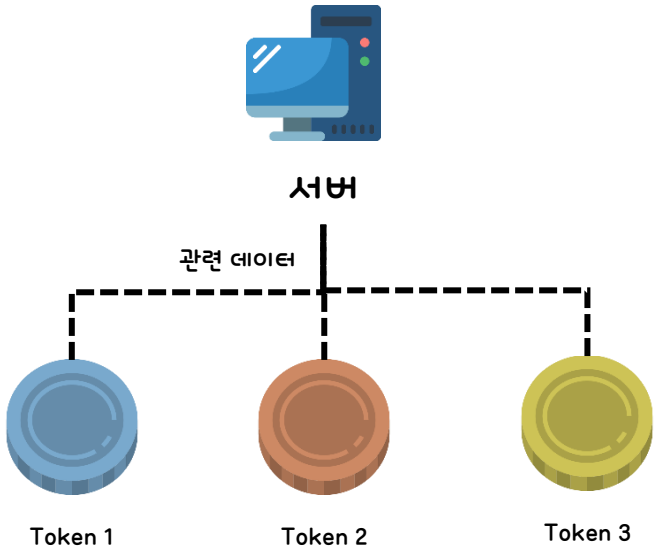
토큰 생성 및 발행

서버 제공

NFT 토큰 발행

NFT 토큰 소유

토큰 ID 관리



NFT 실습

https://eips.ethereum.org/EIPS/eip-721

Every ERC-721 compliant contract must implement the `ERC721` and `ERC165` interfaces (subject to "caveats" below):

```
pragma solidity ^0.4.20;

/// @title ERC-721 Non-Fungible Token Standard
/// @dev See https://eips.ethereum.org/EIPS/eip-721
///  Note: the ERC-165 identifier for this interface is 0x80ac58cd.
interface ERC721 /* is ERC165 */ {
    /// @dev This emits when ownership of any NFT changes by any mechanism.
    ///  This event emits when NFTs are created (`from` == 0) and destroyed
    ///  (`to` == 0). Exception: during contract creation, any number of NFTs
    ///  may be created and assigned without emitting Transfer. At the time of
    ///  any transfer, the approved address for that NFT (if any) is reset to none.
    event Transfer(address indexed _from, address indexed _to, uint256 indexed _tokenId);

    /// @dev This emits when the approved address for an NFT is changed or
    ///  reaffirmed. The zero address indicates there is no approved address.
    ///  When a Transfer event emits, this also indicates that the approved
    ///  address for that NFT (if any) is reset to none.
    event Approval(address indexed _owner, address indexed _approved, uint256 indexed _tokenId);

    /// @dev This emits when an operator is enabled or disabled for an owner.
    ///  The operator can manage all NFTs of the owner.
    event ApprovalForAll(address indexed _owner, address indexed _operator, bool _approved);

    /// @notice Count all NFTs assigned to an owner
    /// @dev NFTs assigned to the zero address are considered invalid, and this
    ///  function throws for queries about the zero address.
    /// @param _owner An address for whom to query the balance
    /// @return The number of NFTs owned by `_owner`, possibly zero
    function balanceOf(address _owner) external view returns (uint256);
```

```
    /// @notice Find the owner of an NFT
    /// @dev NFTs assigned to zero address are considered invalid, and queries
    ///  about them do throw.
    /// @param _tokenId The identifier for an NFT
    /// @return The address of the owner of the NFT
    function ownerOf(uint256 _tokenId) external view returns (address);

    /// @notice Transfers the ownership of an NFT from one address to another address
    /// @dev Throws unless `msg.sender` is the current owner, an authorized
    ///  operator, or the approved address for this NFT. Throws if `_from` is
    ///  not the current owner. Throws if `_to` is the zero address. Throws if
    ///  `_tokenId` is not a valid NFT. When transfer is complete, this function
    ///  checks if `_to` is a smart contract (code size > 0). If so, it calls
    ///  `onERC721Received` on `_to` and throws if the return value is not
    ///  `bytes4(keccak256("onERC721Received(address,address,uint256,bytes)"))`.
    /// @param _from The current owner of the NFT
    /// @param _to The new owner
    /// @param _tokenId The NFT to transfer
    /// @param data Additional data with no specified format, sent in call to `_to`
    function safeTransferFrom(address _from, address _to, uint256 _tokenId, bytes data) external payable;

    /// @notice Transfers the ownership of an NFT from one address to another address
    /// @dev This works identically to the other function with an extra data parameter,
    ///  except this function just sets data to "".
    /// @param _from The current owner of the NFT
    /// @param _to The new owner
    /// @param _tokenId The NFT to transfer
    function safeTransferFrom(address _from, address _to, uint256 _tokenId) external payable;
```

NFT 실습

https://eips.ethereum.org/EIPS/eip-721

Every ERC-721 compliant contract must implement the `ERC721` and `ERC165` interfaces (subject to "caveats" below):

```
function transferFrom(address _from, address _to, uint256 _tokenId) external payable;

/// @notice Change or reaffirm the approved address for an NFT
/// @dev The zero address indicates there is no approved address.
///  Throws unless `msg.sender` is the current NFT owner, or an authorized
///  operator of the current owner.
/// @param _approved The new approved NFT controller
/// @param _tokenId The NFT to approve
function approve(address _approved, uint256 _tokenId) external payable;

/// @notice Enable or disable approval for a third party ("operator") to manage
///  all of `msg.sender`'s assets
/// @dev Emits the ApprovalForAll event. The contract MUST allow
///  multiple operators per owner.
/// @param _operator Address to add to the set of authorized operators
/// @param _approved True if the operator is approved, false to revoke approval
function setApprovalForAll(address _operator, bool _approved) external;

/// @notice Get the approved address for a single NFT
/// @dev Throws if `_tokenId` is not a valid NFT.
/// @param _tokenId The NFT to find the approved address for
/// @return The approved address for this NFT, or the zero address if there is none
function getApproved(uint256 _tokenId) external view returns (address);

/// @notice Query if an address is an authorized operator for another address
/// @param _owner The address that owns the NFTs
/// @param _operator The address that acts on behalf of the owner
/// @return True if `_operator` is an approved operator for `_owner`, false otherwise
function isApprovedForAll(address _owner, address _operator) external view returns (bool);
```

```
interface ERC165 {
    /// @notice Query if a contract implements an interface
    /// @param interfaceID The interface identifier, as specified in ERC-165
    /// @dev Interface identification is specified in ERC-165. This function
    ///  uses less than 30,000 gas.
    /// @return `true` if the contract implements `interfaceID` and
    ///  `interfaceID` is not 0xffffffff, `false` otherwise
    function supportsInterface(bytes4 interfaceID) external view returns (bool);
}
```

NFT 실습

https://eips.ethereum.org/EIPS/eip-721

A wallet/broker/auction application MUST implement the **wallet interface** if it will accept safe transfers.

```
/// @dev Note: the ERC-165 identifier for this interface is 0x150b7a02.
interface ERC721TokenReceiver {
    /// @notice Handle the receipt of an NFT
    /// @dev The ERC721 smart contract calls this function on the recipient
    ///  after a `transfer`. This function MAY throw to revert and reject the
    ///  transfer. Return of other than the magic value MUST result in the
    ///  transaction being reverted.
    ///  Note: the contract address is always the message sender.
    /// @param _operator The address which called `safeTransferFrom` function
    /// @param _from The address which previously owned the token
    /// @param _tokenId The NFT identifier which is being transferred
    /// @param _data Additional data with no specified format
    /// @return `bytes4(keccak256("onERC721Received(address,address,uint256,bytes)"))`
    ///  unless throwing
    function onERC721Received(address _operator, address _from, uint256 _tokenId, bytes _data) external returns(bytes4);
}
```

NFT 실습

https://eips.ethereum.org/EIPS/eip-721

The **metadata extension** is OPTIONAL for ERC-721 smart contracts (see "caveats", below). This allows your smart contract to be interrogated for its name and for details about the assets which your NFTs represent.

```
/// @title ERC-721 Non-Fungible Token Standard, optional metadata extension
/// @dev See https://eips.ethereum.org/EIPS/eip-721
///  Note: the ERC-165 identifier for this interface is 0x5b5e139f.
interface ERC721Metadata /* is ERC721 */ {
    /// @notice A descriptive name for a collection of NFTs in this contract
    function name() external view returns (string _name);

    /// @notice An abbreviated name for NFTs in this contract
    function symbol() external view returns (string _symbol);

    /// @notice A distinct Uniform Resource Identifier (URI) for a given asset.
    /// @dev Throws if `_tokenId` is not a valid NFT. URIs are defined in RFC
    ///  3986. The URI may point to a JSON file that conforms to the "ERC721
    ///  Metadata JSON Schema".
    function tokenURI(uint256 _tokenId) external view returns (string);
}
```

NFT 실습

https://eips.ethereum.org/EIPS/eip-721

The **enumeration extension** is OPTIONAL for ERC-721 smart contracts (see "caveats", below). This allows your contract to publish its full list of NFTs and make them discoverable.

```
/// @title ERC-721 Non-Fungible Token Standard, optional enumeration extension
/// @dev See https://eips.ethereum.org/EIPS/eip-721
///  Note: the ERC-165 identifier for this interface is 0x780e9d63.
interface ERC721Enumerable /* is ERC721 */ {
    /// @notice Count NFTs tracked by this contract
    /// @return A count of valid NFTs tracked by this contract, where each one of
    ///  them has an assigned and queryable owner not equal to the zero address
    function totalSupply() external view returns (uint256);

    /// @notice Enumerate valid NFTs
    /// @dev Throws if `_index` >= `totalSupply()`.
    /// @param _index A counter less than `totalSupply()`
    /// @return The token identifier for the `_index`th NFT,
    ///  (sort order not specified)
    function tokenByIndex(uint256 _index) external view returns (uint256);

    /// @notice Enumerate NFTs assigned to an owner
    /// @dev Throws if `_index` >= `balanceOf(_owner)` or if
    ///  `_owner` is the zero address, representing invalid NFTs.
    /// @param _owner An address where we are interested in NFTs owned by them
    /// @param _index A counter less than `balanceOf(_owner)`
    /// @return The token identifier for the `_index`th NFT assigned to `_owner`,
    ///   (sort order not specified)
    function tokenOfOwnerByIndex(address _owner, uint256 _index) external view returns (uint256);
}
```

NFT 실습

https://ethereum.org/en/developers/docs/standards/tokens/erc-721/

Methods

Show less | Copy

```
    function balanceOf(address _owner) external view returns (uint256);
    function ownerOf(uint256 _tokenId) external view returns (address);
    function safeTransferFrom(address _from, address _to, uint256 _tokenId, bytes data) external payable;
    function safeTransferFrom(address _from, address _to, uint256 _tokenId) external payable;
    function transferFrom(address _from, address _to, uint256 _tokenId) external payable;
    function approve(address _approved, uint256 _tokenId) external payable;
    function setApprovalForAll(address _operator, bool _approved) external;
    function getApproved(uint256 _tokenId) external view returns (address);
    function isApprovedForAll(address _owner, address _operator) external view returns (bool);

```

Events

Copy

```
    event Transfer(address indexed _from, address indexed _to, uint256 indexed _tokenId);
    event Approval(address indexed _owner, address indexed _approved, uint256 indexed _tokenId);
    event ApprovalForAll(address indexed _owner, address indexed _operator, bool _approved);

```

NFT 실습

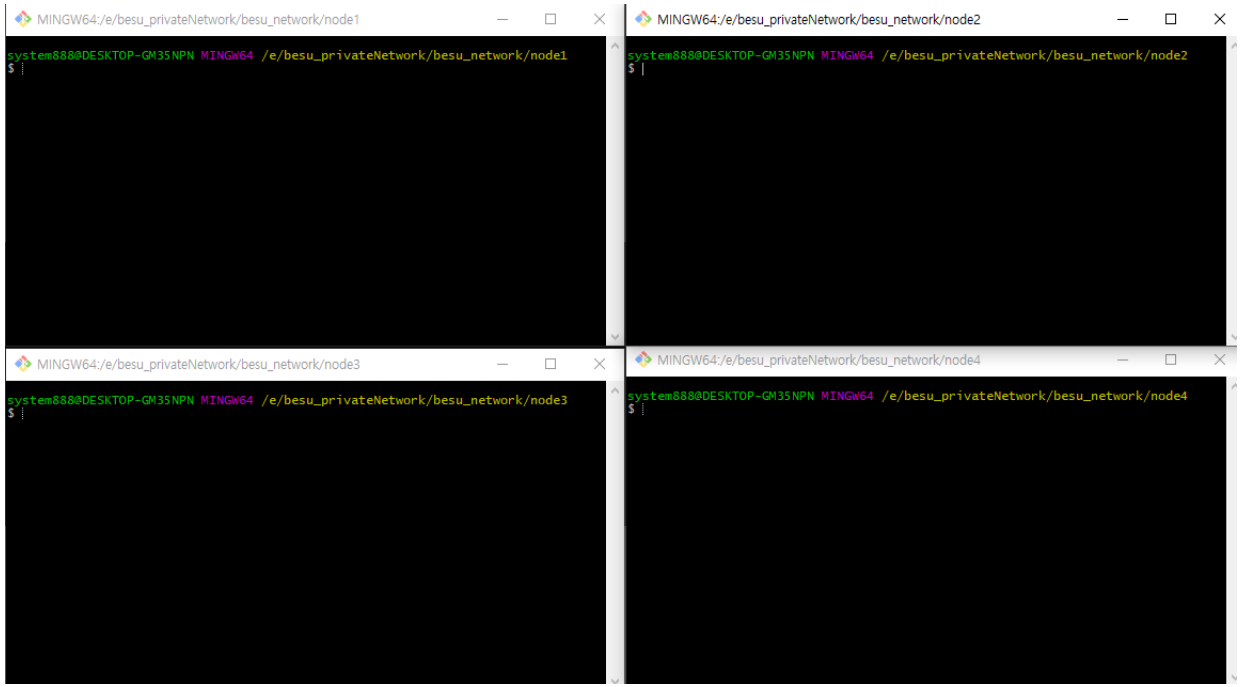
간단한 NFT 발행 및 전송 실습

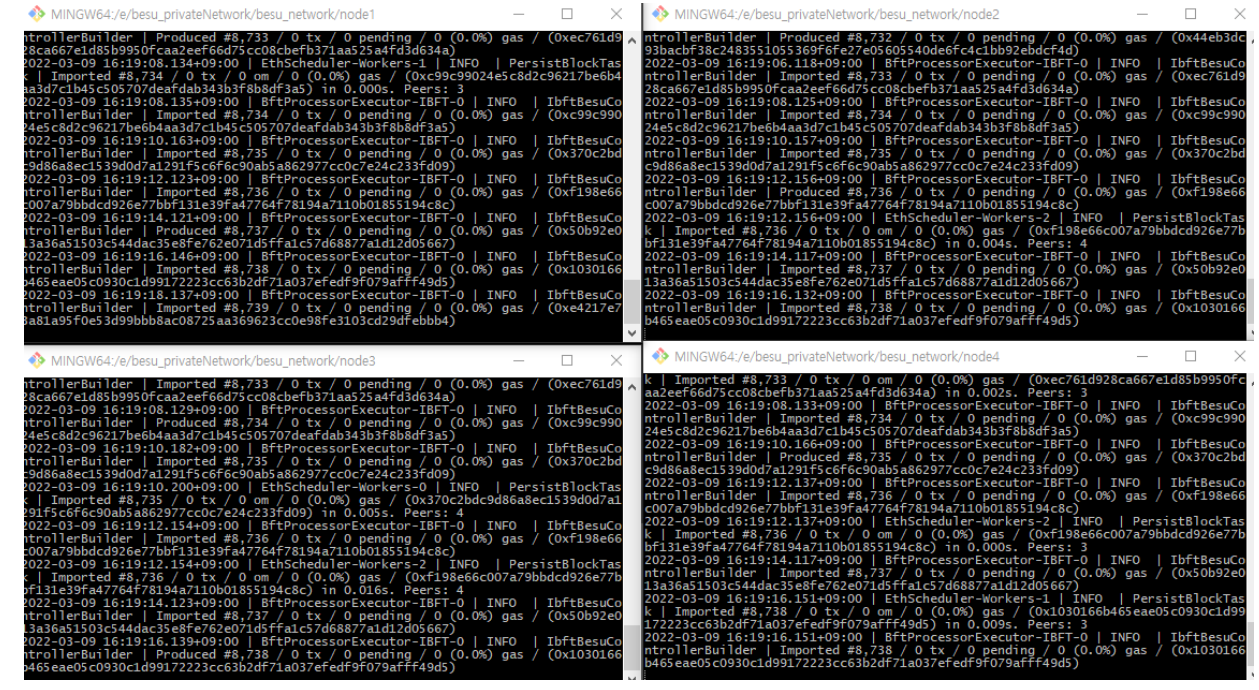
```solidity
pragma solidity ^0.4.20;

contract Contract721 {

    string private _name;
    string private _symbol;

    mapping(uint256 => address) private _owners;
    mapping(address => uint256) private _balances;
    mapping(uint256 => address) private _tokenApprovals;
    mapping(uint256 => string) private _metadata;

    constructor(string name_, string symbol_, string metadata_, uint256 tokenId){
        _name = name_;
        _symbol= symbol_;
        _mint(tokenId, metadata_);
    }

    function _mint(uint256 tokenId, string metadata_) internal {
        require(!_exists(tokenId), "Token already minted");

        _balances[msg.sender] +=1;
        _owners[tokenId] = msg.sender;
        _metadata[tokenId] = metadata_;

    }

    function name() public view returns (string){
        return _name;
    }

    function symbol() public view returns (string){
        return _symbol;
    }

    function _exists(uint256 tokenId) internal returns (bool) {
        return _owners[tokenId] != address(0);
    }

    function additionalMint(uint tokenId, string metadata) external {
        _mint(tokenId, metadata);
    }
```

```solidity
    function additionalMint(uint tokenId, string metadata) external {
        _mint(tokenId, metadata);
    }


    function balanceOf(address _owner) external view returns (uint256) {
        return _balances[_owner];
    }


    function ownerOf(uint256 _tokenId) external view returns (address) {
        address owner = _owners[_tokenId];
        return owner;
    }

    function getMetaData(uint256 _tokenId) external view returns (string){
        string metadata = _metadata[_tokenId];
        return metadata;
    }

    function transferFrom(address _from, address _to, uint256 _tokenId) external payable {
        _transfer(_from, _to, _tokenId);
    }

    function _transfer(address from, address to, uint256 tokenId) internal{
        _balances[from] -= 1;
        _balances[to] += 1;
        _owners[tokenId] = to;

    }

}
```

NFT 실습

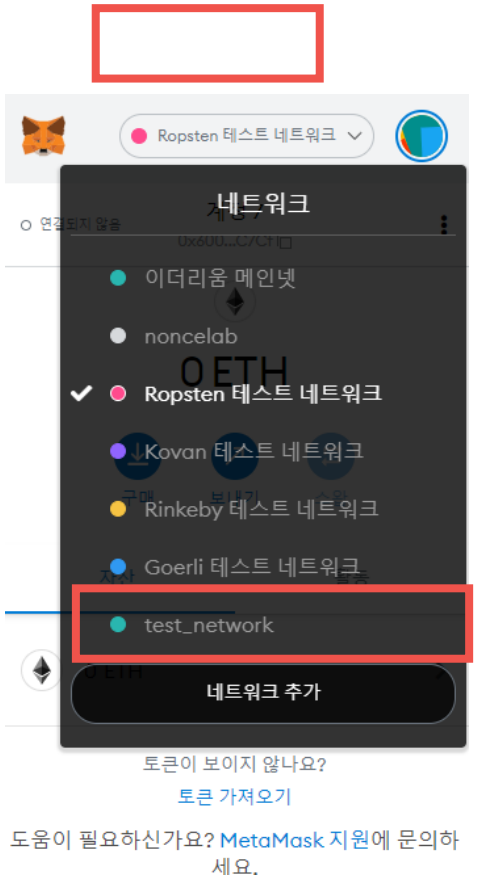
하이퍼레저 베수 구동

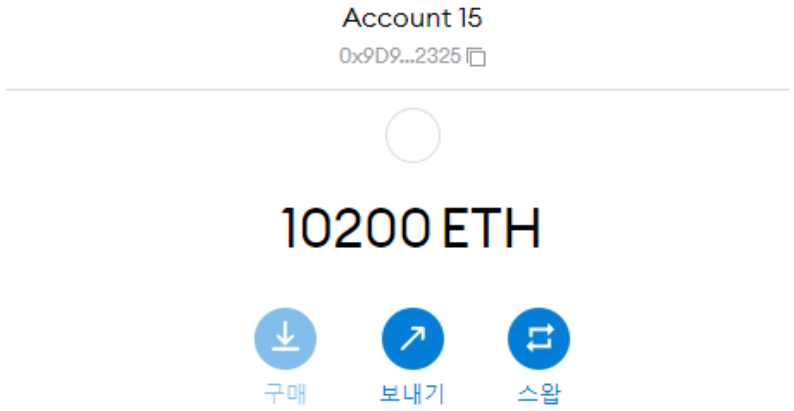




NFT 실습

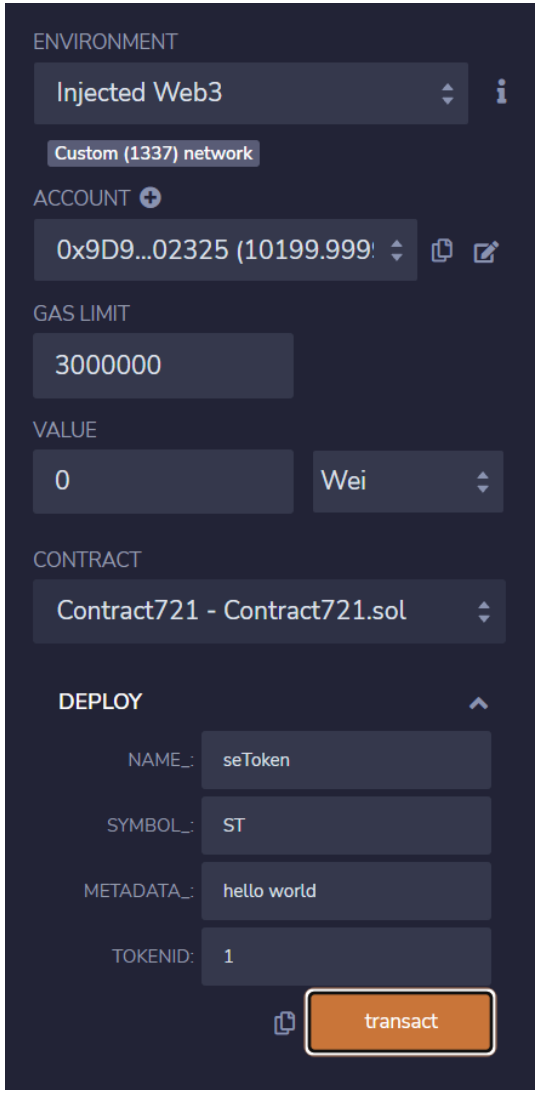
하이퍼레저 베수와 메타마스크 연결
이후, remix 연동

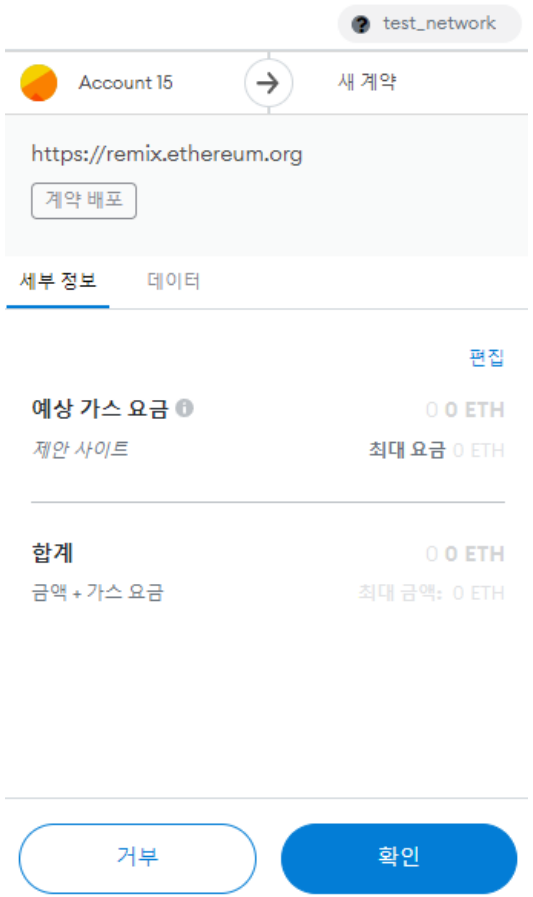


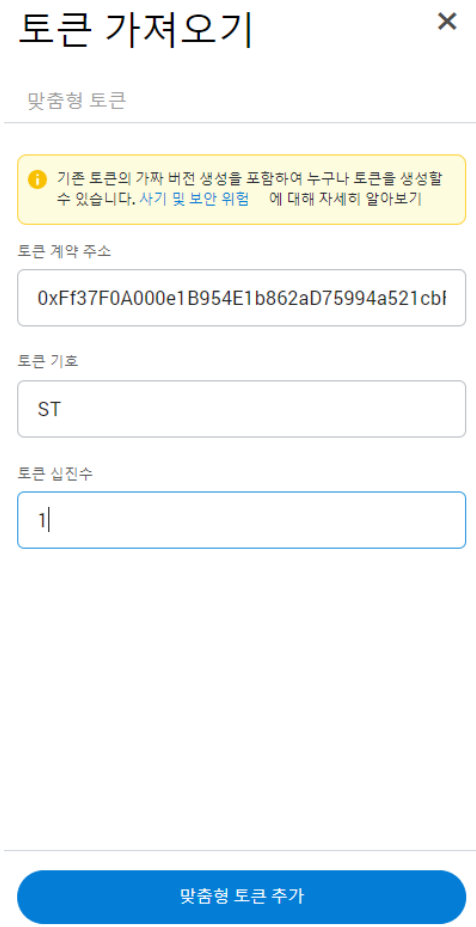


NFT 실습

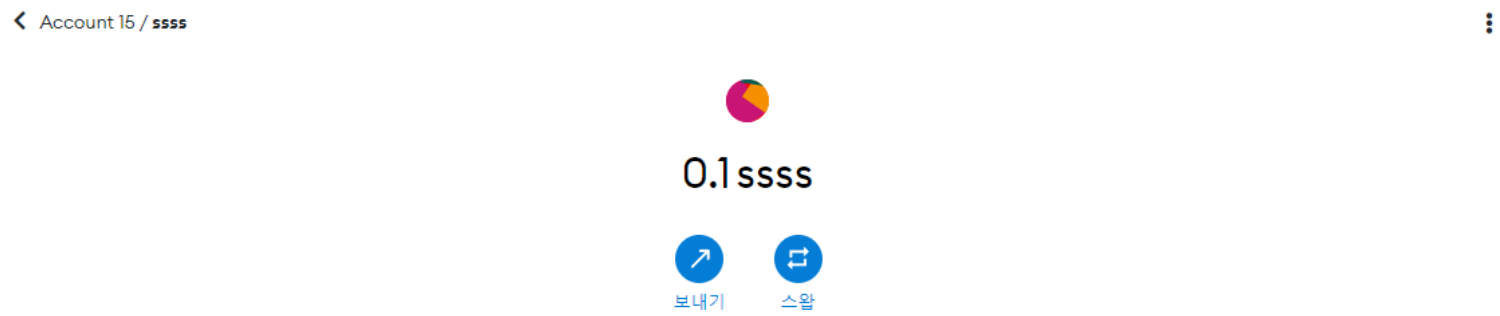
스마트 컨트랙트 배포

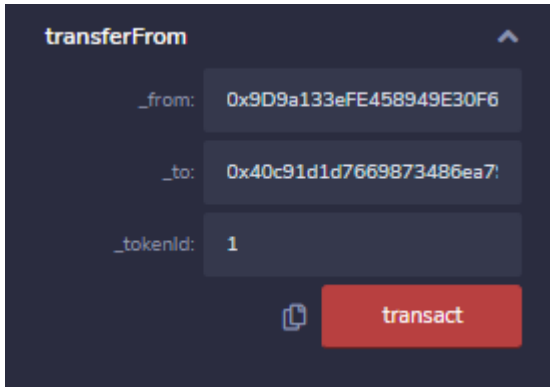


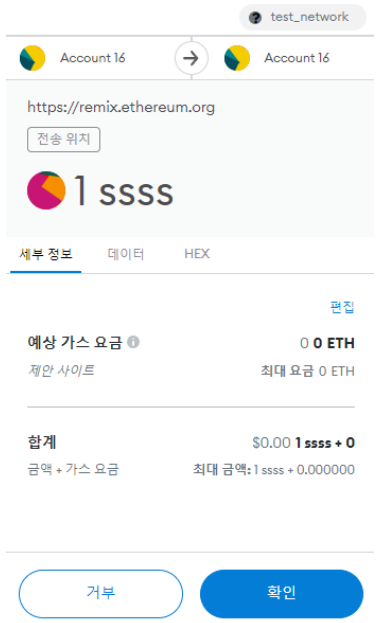




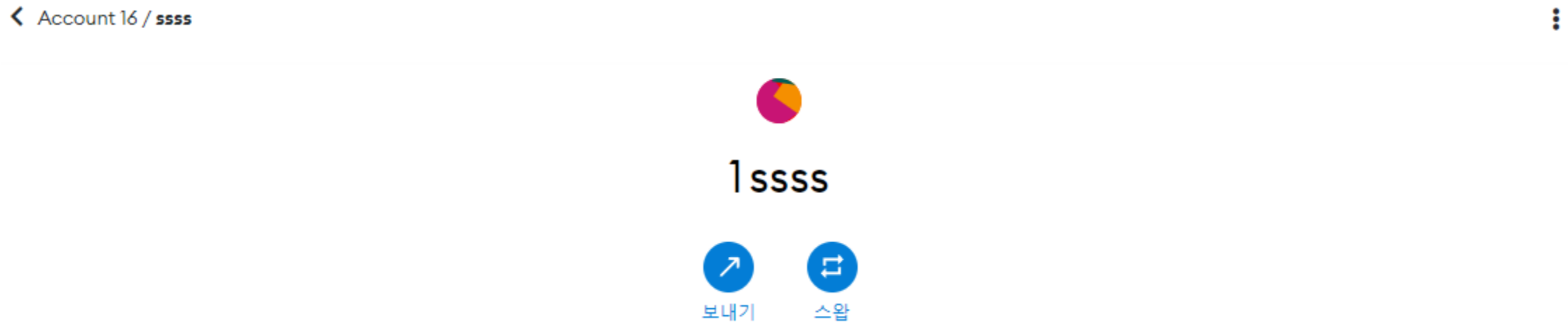
NFT 실습


Account 15 / ssss
0.1 ssss
보내기
스왑


transferFrom
_from:
0x9D9a133eFE458949E30F6
_to:
0x40c91d1d7669873486ea7
_tokenId:
1
transact


test_network
Account 16
Account 16
https://remix.ethereum.org
전송 위치
1 ssss
세부 정보
데이터
HEX
편집
예상 가스 요금
0 0 ETH
제안 사이트
최대 요금 0 ETH
합계
$0.00 1 ssss + 0
금액 + 가스 요금
최대 금액: 1 ssss + 0.000000
거부
확인

NFT 실습



NFT 실습

NFT 발행의 핵심

```
pragma solidity ^0.4.20;

contract Contract721 {

    string private _name;
    string private _symbol;

    mapping(uint256 => address) private _owners;
    mapping(address => uint256) private _balances;
    mapping(uint256 => address) private _tokenApprovals;
    mapping(uint256 => string) private _metadata;

    constructor(string name_, string symbol_, string metadata_, uint256 tokenId){
        _name = name_;
        _symbol= symbol_;
        _mint(tokenId, metadata_);
    }

    function _mint(uint256 tokenId, string metadata_) internal {
        require(!_exists(tokenId), "Token already minted");

        _balances[msg.sender] +=1;
        _owners[tokenId] = msg.sender;
        _metadata[tokenId] = metadata_;

    }

    function name() public view returns (string){
        return _name;
    }

    function symbol() public view returns (string){
        return _symbol;
    }

    function _exists(uint256 tokenId) internal returns (bool) {
        return _owners[tokenId] != address(0);
    }

    function additionalMint(uint tokenId, string metadata) external {
        _mint(tokenId, metadata);
    }
```

- Token ID와 사용자의 매칭이 잘 되는가
- TokenID가 잘 생성이 되는가
- 소유자 개념이 잘 등록이 되는가

NFT 실습

컨트랙트 소유자 및 관리자

토큰 생성 및 발행

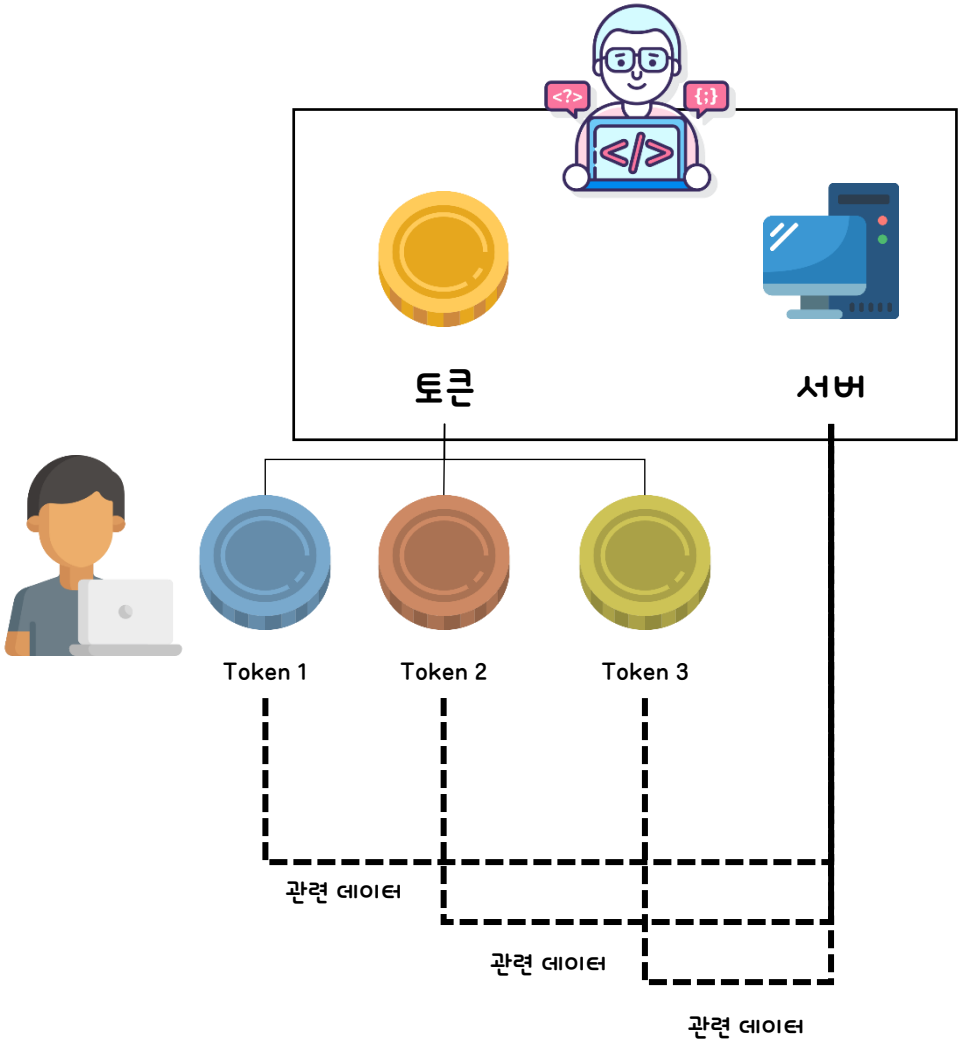
서버 제공

NFT 토큰 발행

디지털 작품 생산자

NFT 토큰 소유

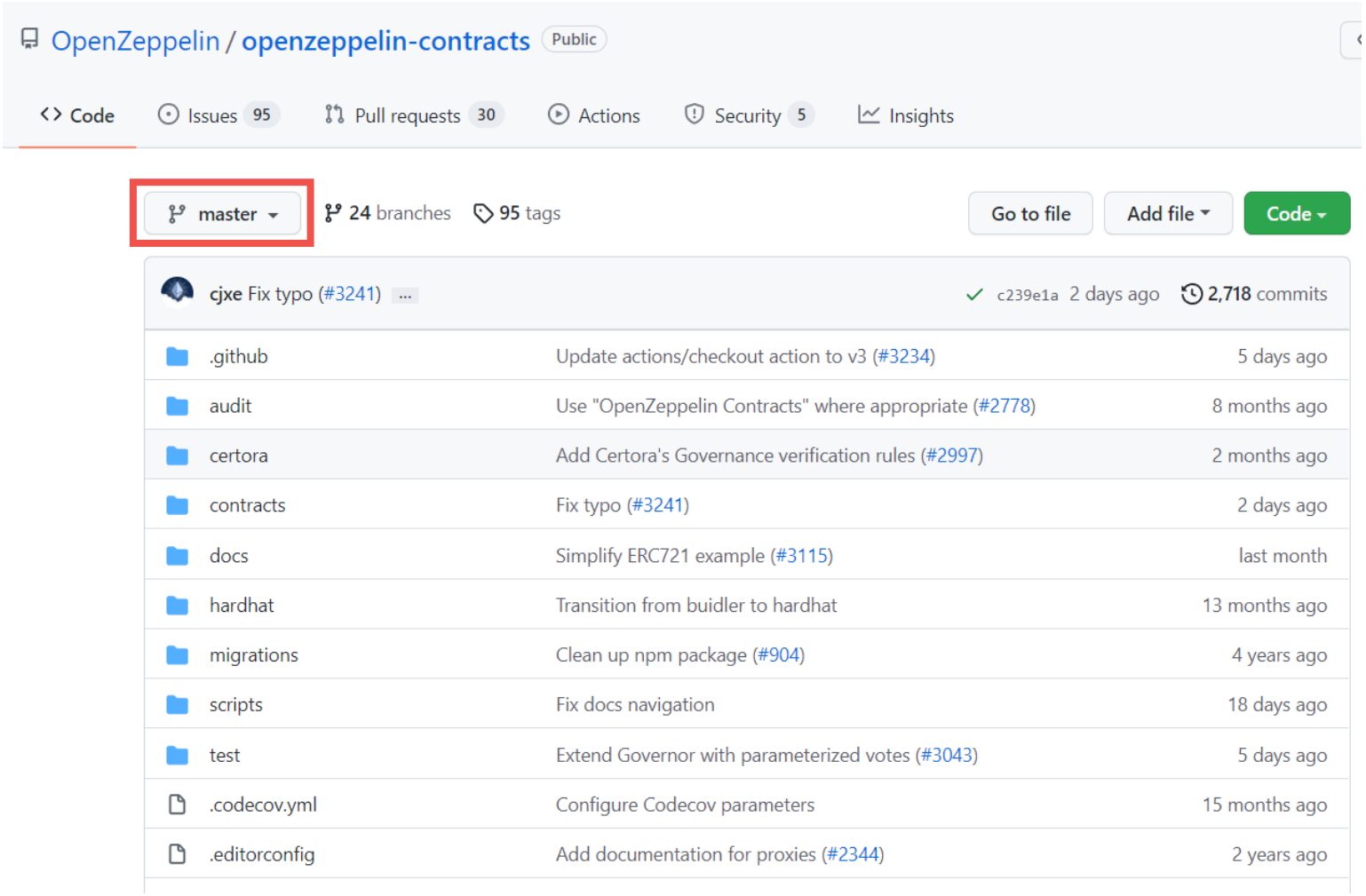
토큰 ID 관리

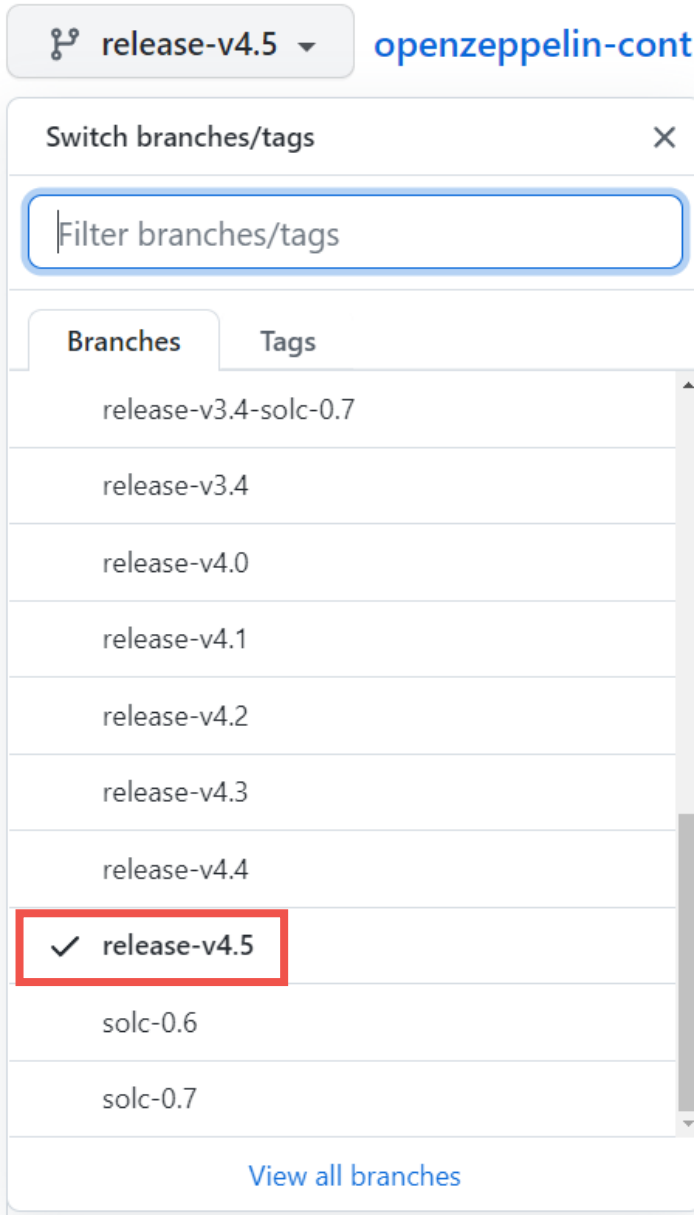


NFT 실습

NFT 실습

https://github.com/OpenZeppelin/openzeppelin-contracts




release-v4.5
openzeppelin-cont
Switch branches/tags
Filter branches/tags
Branches
Tags
release-v3.4-solc-0.7
release-v3.4
release-v4.0
release-v4.1
release-v4.2
release-v4.3
release-v4.4
release-v4.5
solc-0.6
solc-0.7
View all branches

NFT 실습

https://docs.openzeppelin.com/contracts/4.x/api/token/erc721

The EIP specifies four interfaces:

- `IERC721` : Core functionality required in all compliant implementation.
- `IERC721Metadata` : Optional extension that adds name, symbol, and token URI, almost always included.
- `IERC721Enumerable` : Optional extension that allows enumerating the tokens on chain, often not included since it requires large gas overhead.
- `IERC721Receiver` : An interface that must be implemented by contracts if they want to accept tokens through `safeTransferFrom` .

OpenZeppelin Contracts provides implementations of all four interfaces:

- `ERC721` : The core and metadata extensions, with a base URI mechanism.
- `ERC721Enumerable` : The enumerable extension.
- `ERC721Holder` : A bare bones implementation of the receiver interface.

Additionally there are a few of other extensions:

- `ERC721URIStorage` : A more flexible but more expensive way of storing metadata.
- `ERC721Votes` : Support for voting and vote delegation.
- `ERC721Royalty` : A way to signal royalty information following ERC2981.
- `ERC721Pausable` : A primitive to pause contract operation.
- `ERC721Burnable` : A way for token holders to burn their own tokens.

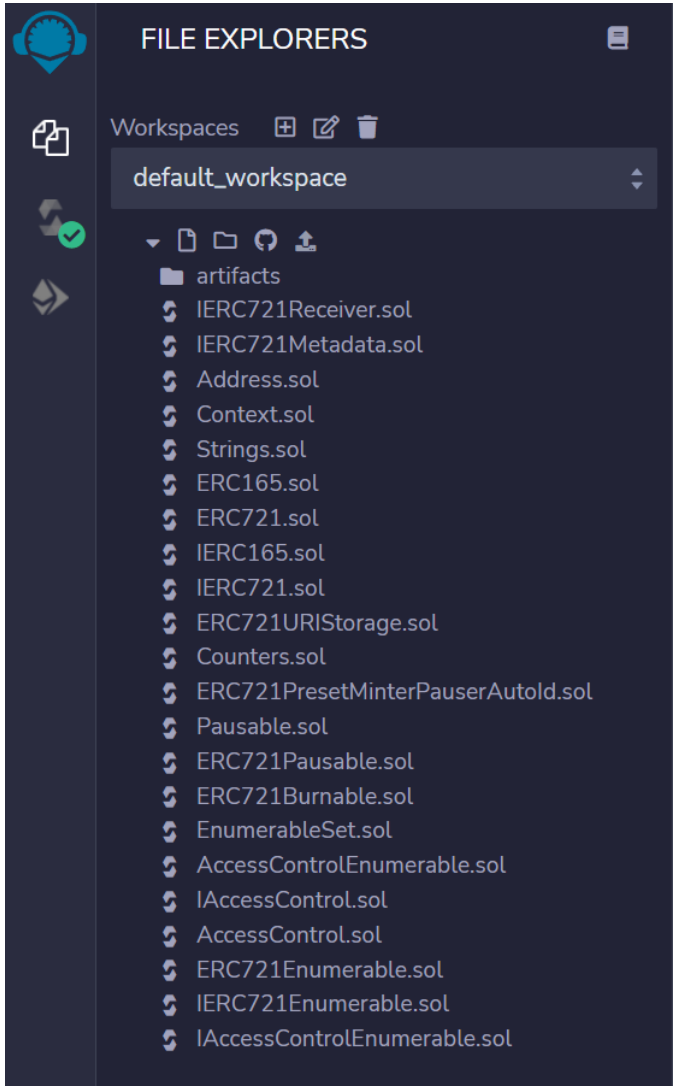
- 제공하는 Smart Contract들은 Core 요소들일뿐
- 해당 Core 요소들을 돌리기 위한 스마트 컨트랙트가 한개 더 필요함

NFT 실습

Git 허브에서 가져와야 하는 코드 정리

Index	파일명	파일 위치
1	ERC721.sol	openzeppelin-contracts/contracts/token/ERC721/ERC721.sol
2	IERC721.sol	openzeppelin-contracts/contracts/token/ERC721/IERC721.sol
3	IERC721Receiver.sol	openzeppelin-contracts/contracts/token/ERC721/IERC721Receiver.sol
4	Address.sol	openzeppelin-contracts/contracts/utils/Address.sol
5	Context.sol	openzeppelin-contracts/contracts/utils/Context.sol
6	Strings.sol	openzeppelin-contracts/contracts/utils/Strings.sol
7	ERC165.sol	openzeppelin-contracts/contracts/utils/introspection/ERC165.sol
8	IERC165.sol	openzeppelin-contracts/contracts/utils/introspection/IERC165.sol
9	IERC721Metadata.sol	openzeppelin-contracts/contracts/token/ERC721/extensions/IERC721Metadata.sol
10	ERC721URIStorage.sol	openzeppelin-contracts/contracts/token/ERC721/extensions/ERC721URIStorage.sol
11	ERC721PresetMinterPauserAutoId.sol	openzeppelin-contracts/contracts/token/ERC721/presets/ERC721PresetMinterPauserAutoId.sol
12	ERC721Enumerable.sol	openzeppelin-contracts/contracts/token/ERC721/extensions/ERC721Enumerable.sol
13	ERC721Burnable.sol	openzeppelin-contracts/contracts/token/ERC721/extensions/ERC721Burnable.sol
14	ERC721Pausable.sol	openzeppelin-contracts/contracts/token/ERC721/extensions/ERC721Pausable.sol
15	AccessControlEnumerable.sol	openzeppelin-contracts/contracts/access/AccessControlEnumerable.sol
16	Counters.sol	openzeppelin-contracts/contracts/utils/Counters.sol
17	Pausable.sol	openzeppelin-contracts/contracts/security/Pausable.sol
18	EnumerableSet.sol	openzeppelin-contracts/contracts/utils/structs/EnumerableSet.sol
19	IAcessControlEnumerable.sol	openzeppelin-contracts/contracts/access/IAccessControlEnumerable.sol
20	AccessControl.sol	openzeppelin-contracts/contracts/access/AccessControl.sol
21	IERC721Enumerable.sol	openzeppelin-contracts/contracts/token/ERC721/extensions/IERC721Enumerable.sol
22	IAccessControl.sol	openzeppelin-contracts/contracts/access/IAccessControl.sol

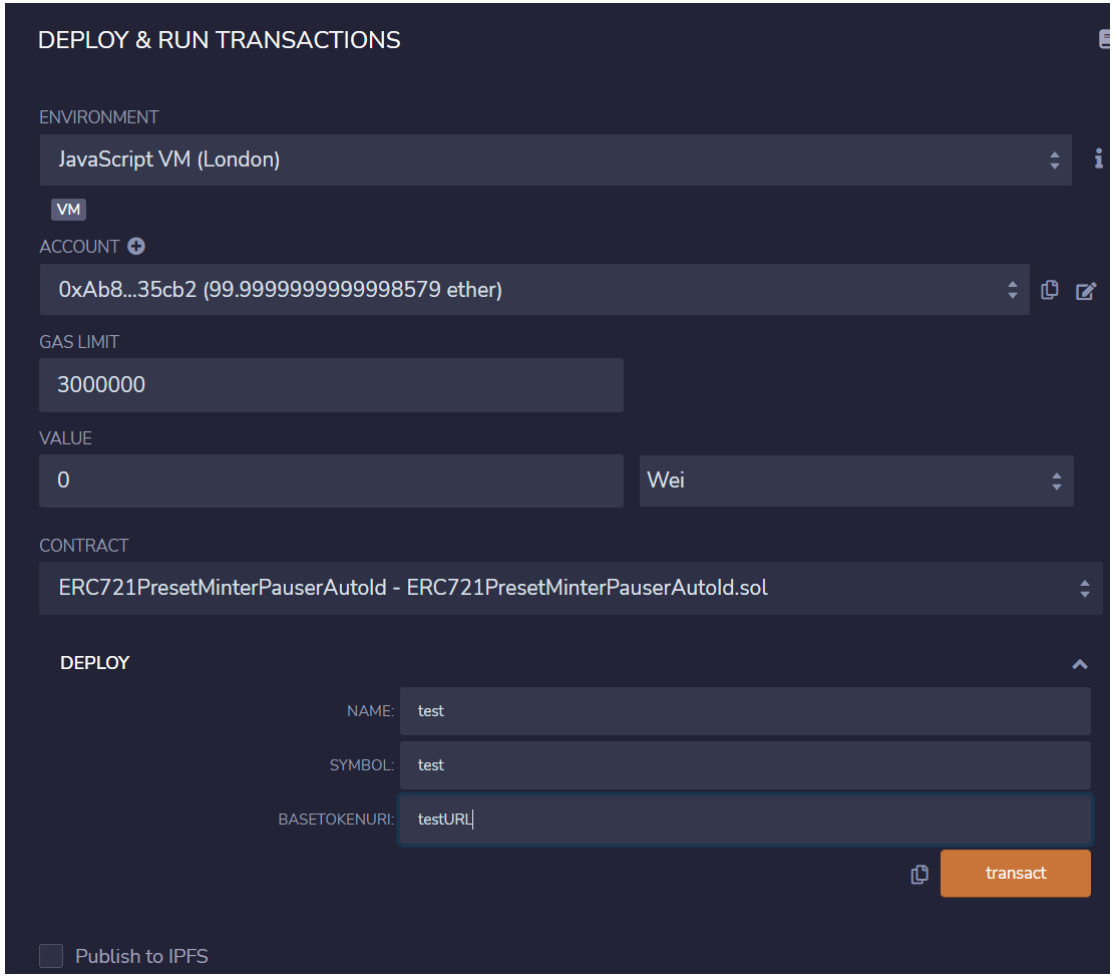
Remix에 넣으면서 import 경로 정리

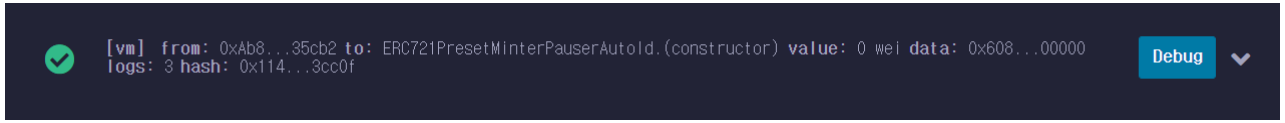


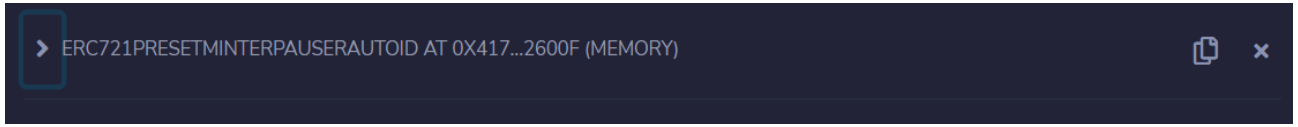

```
import "./IERC721.sol";
import "./IERC721Receiver.sol";
import "./IERC721Metadata.sol";
import "./Address.sol";
import "./Context.sol";
import "./Strings.sol";
import "./ERC165.sol";
```

모든 sol 파일 확인해서 동일 경로에 있도록 수정

Remix VM 테스트


DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
JavaScript VM (London)
VM
ACCOUNT
0xAb8...35cb2 (99.9999999999998579 ether)
GAS LIMIT
3000000
VALUE
0
Wei
CONTRACT
ERC721PresetMinterPauserAutoId - ERC721PresetMinterPauserAutoId.sol
DEPLOY
NAME:
test
SYMBOL:
test
BASETOKENURI:
testURL
transact
Publish to IPFS


[vm] from: 0xAb8...35cb2 to: ERC721PresetMinterPauserAutoId.(constructor) value: 0 wei data: 0x608...00000
logs: 3 hash: 0x114...3cc0f
Debug

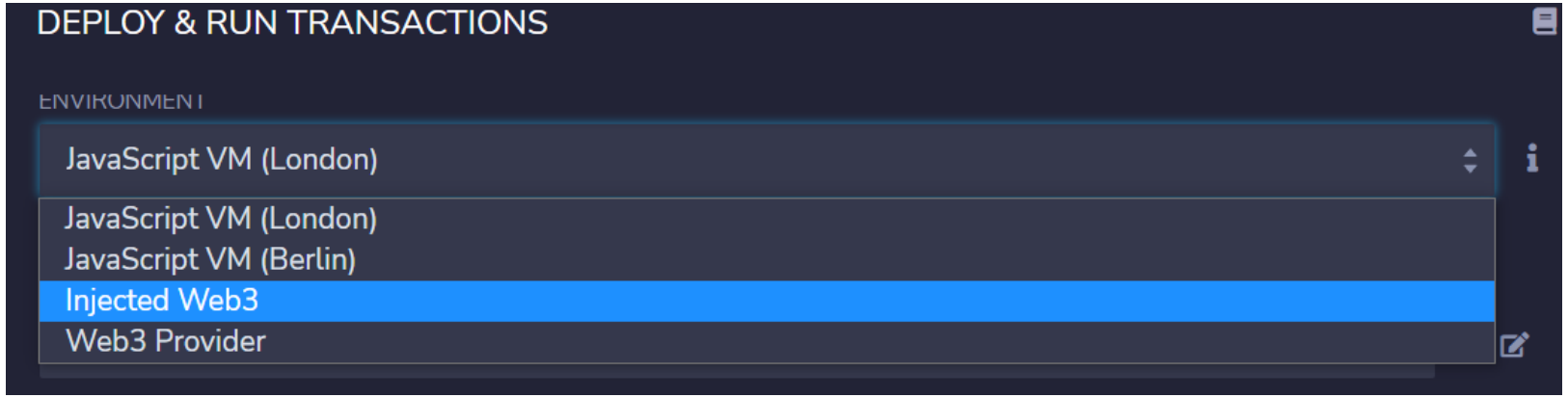

ERC721PRESETMINTERPAUSERAUTOID AT 0X417...2600F (MEMORY)

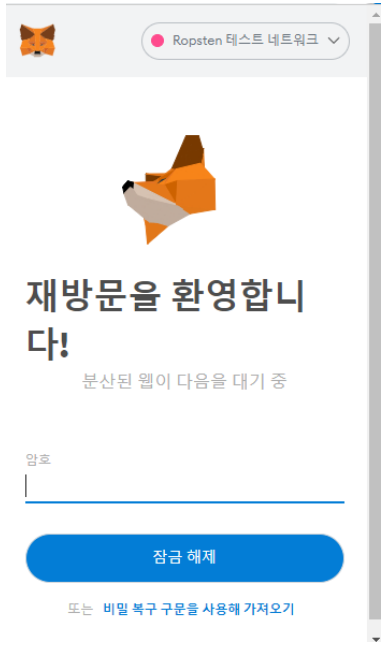
NFT 실습


Deployed Contracts
ERC721PRESETMINTERPAUSERAUTOID AT 0X417...2600F (MEMORY)
approve
address to, uint256 tokenId
burn
uint256 tokenId
grantRole
bytes32 role, address account
mint
address to
pause
renounceRole
bytes32 role, address account
revokeRole
bytes32 role, address account
safeTransferFr...
address from, address to, uint256 tokenId
safeTransferFr...
address from, address to, uint256 tokenId, bytes _data
setApprovalFo...
address operator, bool approved
transferFrom
address from, address to, uint256 tokenId
unpause


balanceOf
address owner
DEFAULT_AD...
getApproved
uint256 tokenId
getRoleAdmin
bytes32 role
getRoleMember
bytes32 role, uint256 index
getRoleMemb...
bytes32 role
hasRole
bytes32 role, address account
isApprovedFor...
address owner, address operator
MINTER_ROLE
name
ownerOf
uint256 tokenId
paused
PAUSER_ROLE
supportsInterf...
bytes4 interfaceId
symbol
tokenByIndex
uint256 index
tokenOfOwner...
address owner, uint256 index
tokenURI
uint256 tokenId
totalSupply

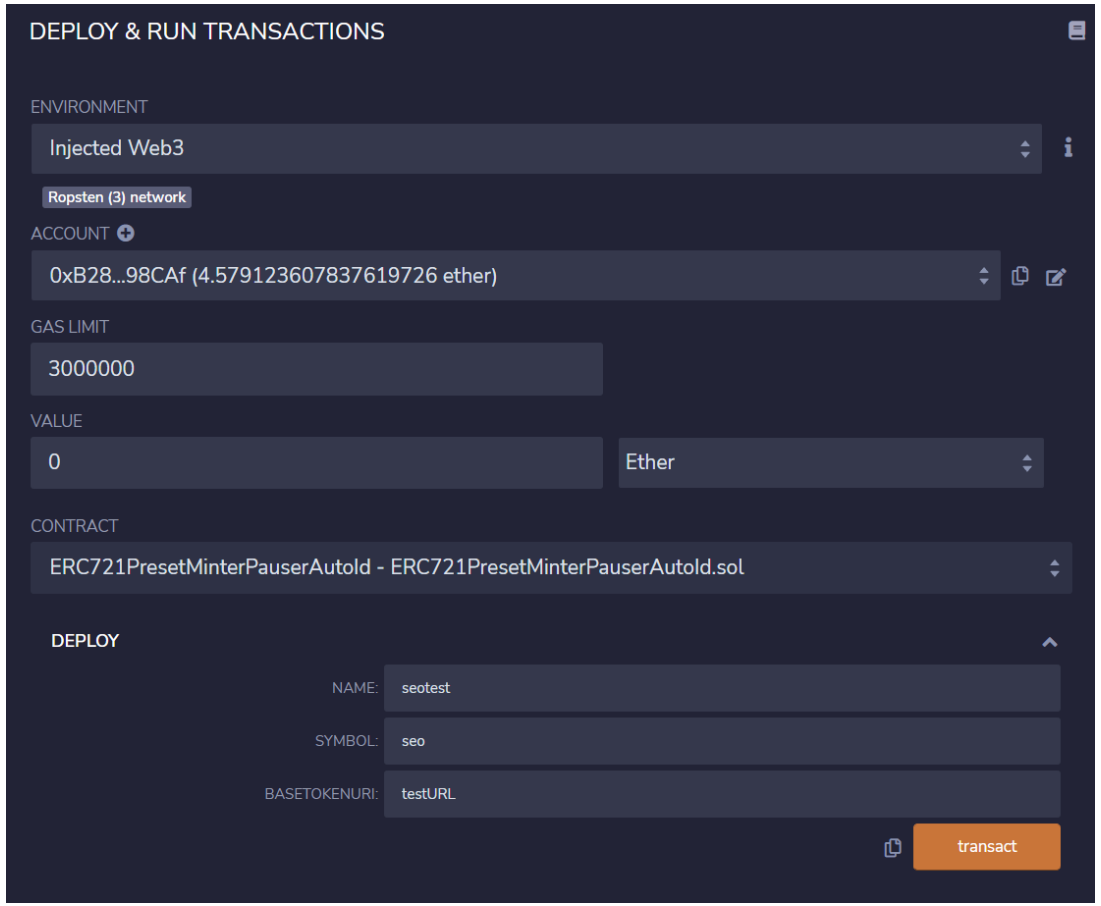
테스트넷 배포 해보기

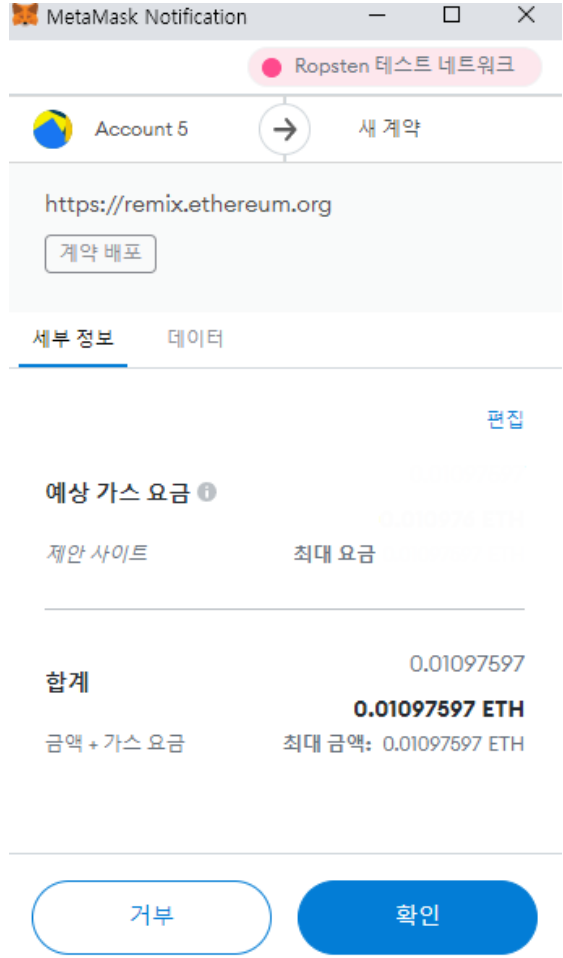

DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
JavaScript VM (London)
JavaScript VM (London)
JavaScript VM (Berlin)
Injected Web3
Web3 Provider


Ropsten 테스트 네트워크
재방문을 환영합니다!
분산된 웹이 다음을 대기 중
암호
잠금 해제
또는 비밀 복구 구문을 사용해 가져오기

NFT 실습

테스트넷 배포 해보기

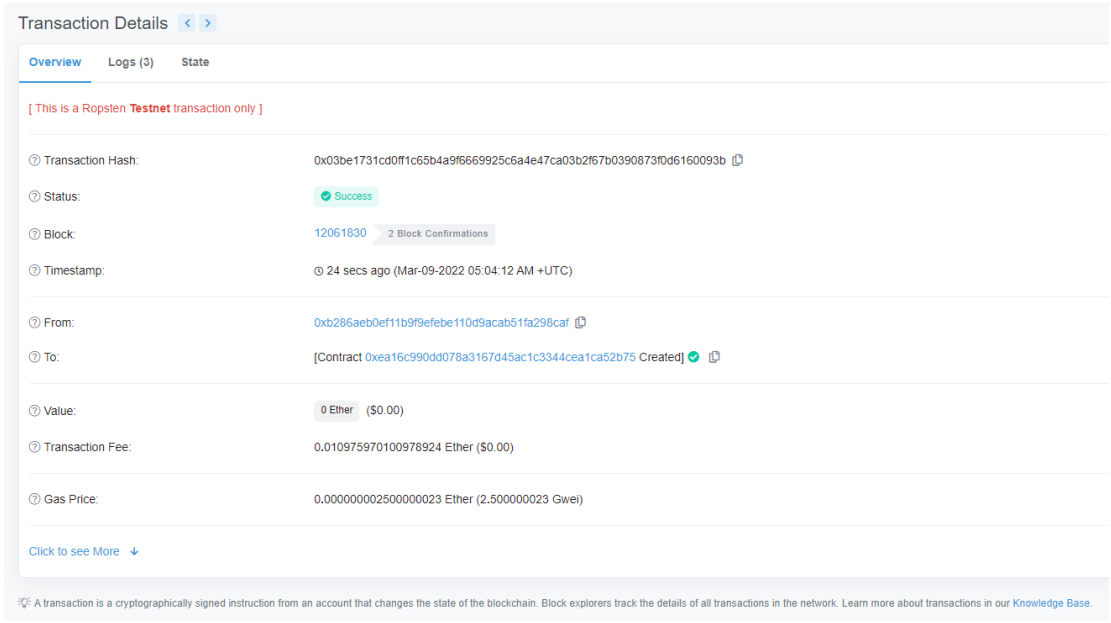


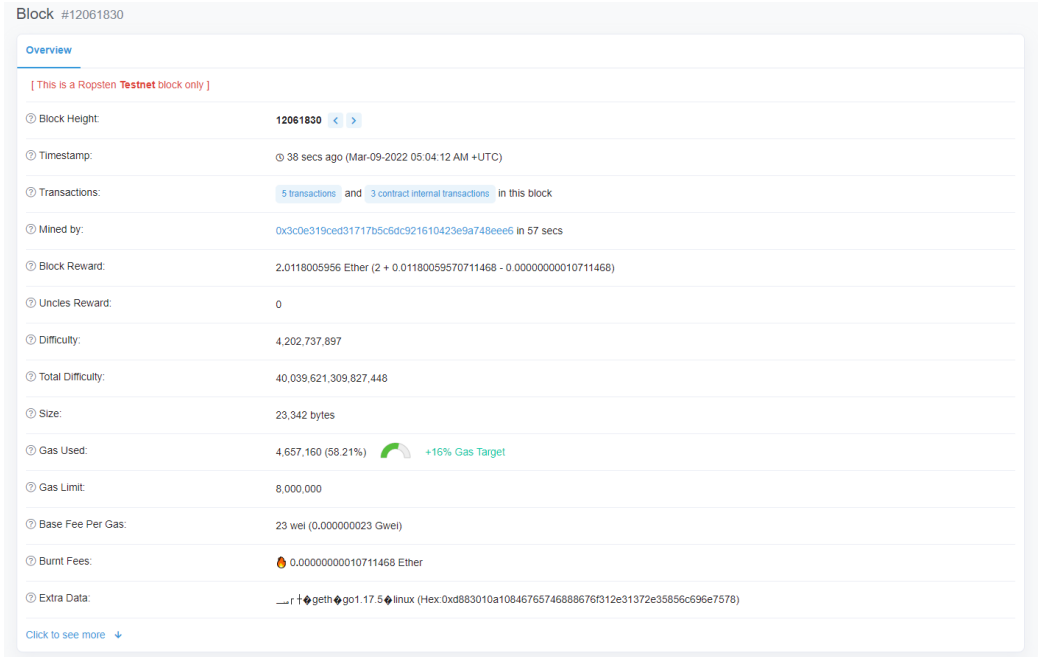


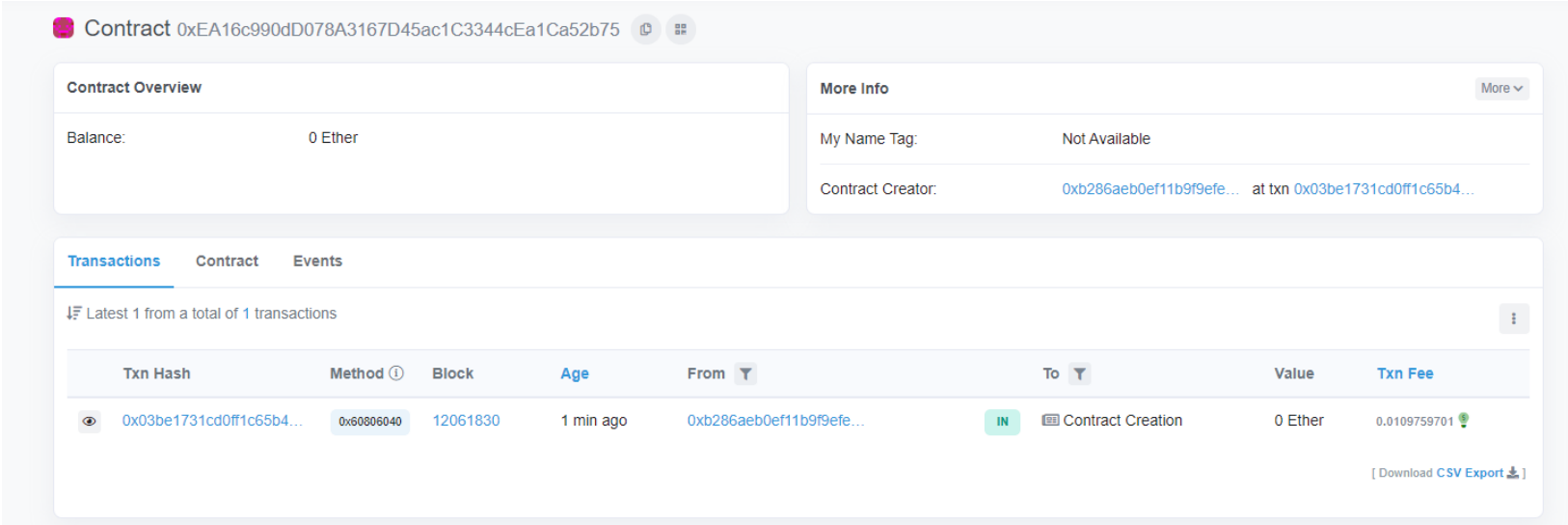
NFT 실습

테스트넷 배포 해보기

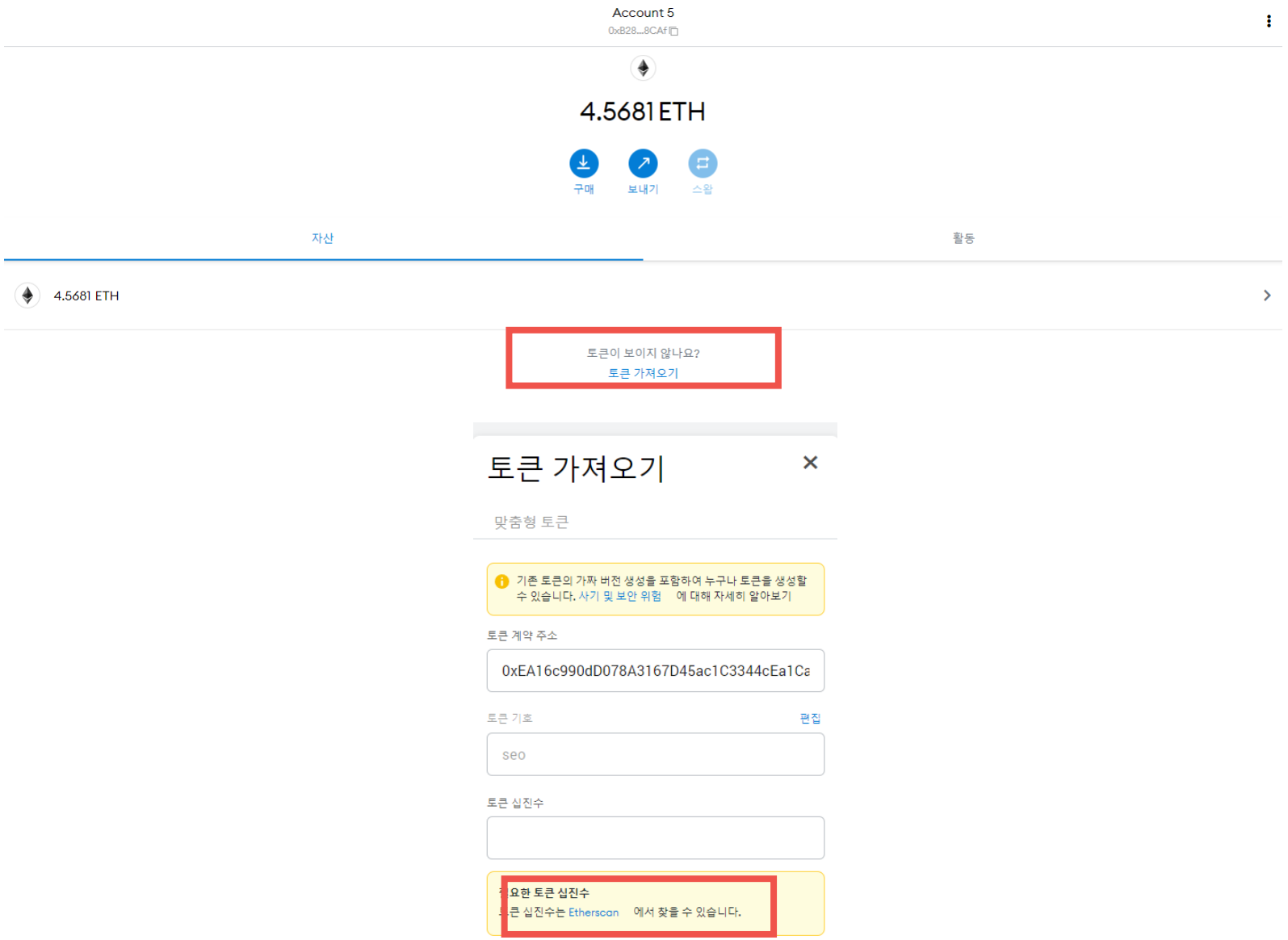
NFT 실습


Transaction Details
Overview
Logs (3)
State
[This is a Ropsten Testnet transaction only]
Transaction Hash: 0x03be1731cd0ff1c65b4a9f6669925c6a4e47ca03b2f67b0390873f0d6160093b
Status: Success
Block: 12061830 2 Block Confirmations
Timestamp: 24 secs ago (Mar-09-2022 05:04:12 AM +UTC)
From: 0xb286aeb0ef11b9f9efebe110d9acab51fa298caf
To: [Contract 0xea16c990dd078a3167d45ac1c3344cea1ca52b75 Created]
Value: 0 Ether ($0.00)
Transaction Fee: 0.010975970100978924 Ether ($0.00)
Gas Price: 0.000000002500000023 Ether (2.500000023 Gwei)
Click to see More
A transaction is a cryptographically signed instruction from an account that changes the state of the blockchain. Block explorers track the details of all transactions in the network. Learn more about transactions in our Knowledge Base.


Block #12061830
Overview
[This is a Ropsten Testnet block only]
Block Height: 12061830
Timestamp: 38 secs ago (Mar-09-2022 05:04:12 AM +UTC)
Transactions: 5 transactions and 3 contract internal transactions in this block
Mined by: 0x3c0e319ced31717b5c6dc921610423e9a748eee6 in 57 secs
Block Reward: 2.0118005956 Ether (2 + 0.011800595707114 68 - 0.000000001071146 8)
Uncles Reward: 0
Difficulty: 4,202,737,897
Total Difficulty: 40,039,621,309,827,448
Size: 23,342 bytes
Gas Used: 4,657,160 (58.21%) +16% Gas Target
Gas Limit: 8,000,000
Base Fee Per Gas: 23 wei (0.000000023 Gwei)
Burnt Fees: 0.000000001071146 8 Ether
Extra Data: __r十◆geth◆go1.17.5◆linux (Hex:0xd883010a10846765746888676f312e31372e35856c696e7578)
Click to see more

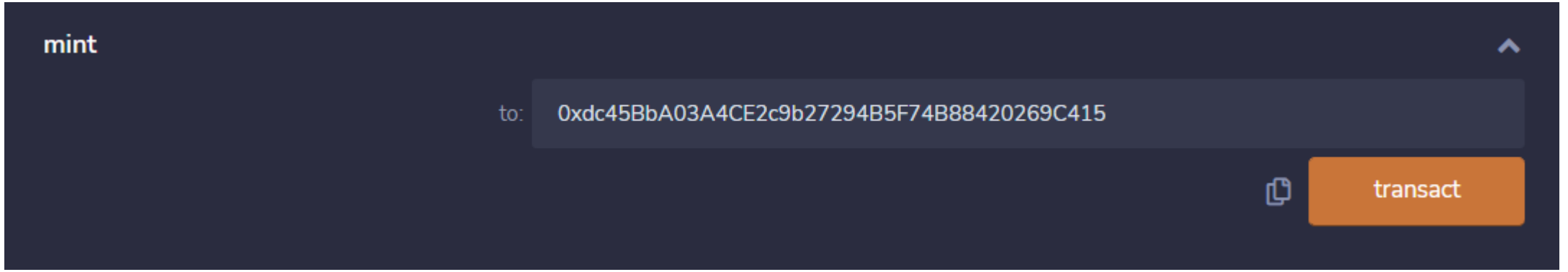

Contract 0xEA16c990dD078A3167D45ac1C3344cEa1Ca52b75
Contract Overview
Balance: 0 Ether
More Info
More
My Name Tag: Not Available
Contract Creator: 0xb286aeb0ef11b9f9efe... at txn 0x03be1731cd0ff1c65b4...
Transactions
Contract
Events
Latest 1 from a total of 1 transactions
Txn Hash
Method
Block
Age
From
To
Value
Txn Fee
0x03be1731cd0ff1c65b4...
0x60806040
12061830
1 min ago
0xb286aeb0ef11b9f9efe...
IN
Contract Creation
0 Ether
0.0109759701
[Download CSV Export]

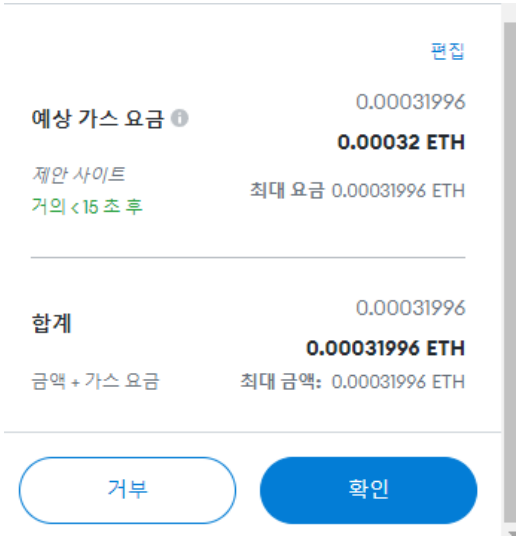
NFT 실습


Account 5
0xB28...8CAf
4.5681 ETH
구매
보내기
스왑
자산
활동
4.5681 ETH
토큰이 보이지 않나요?
토큰 가져오기
토큰 가져오기
맞춤형 토큰
기존 토큰의 가짜 버전 생성을 포함하여 누구나 토큰을 생성할 수 있습니다. 사기 및 보안 위험 에 대해 자세히 알아보기
토큰 계약 주소
0xEA16c990dD078A3167D45ac1C3344cEa1Ca
토큰 기호
편집
seo
토큰 십진수
요한 토큰 십진수
큰 십진수는 Etherscan 에서 찾을 수 있습니다.

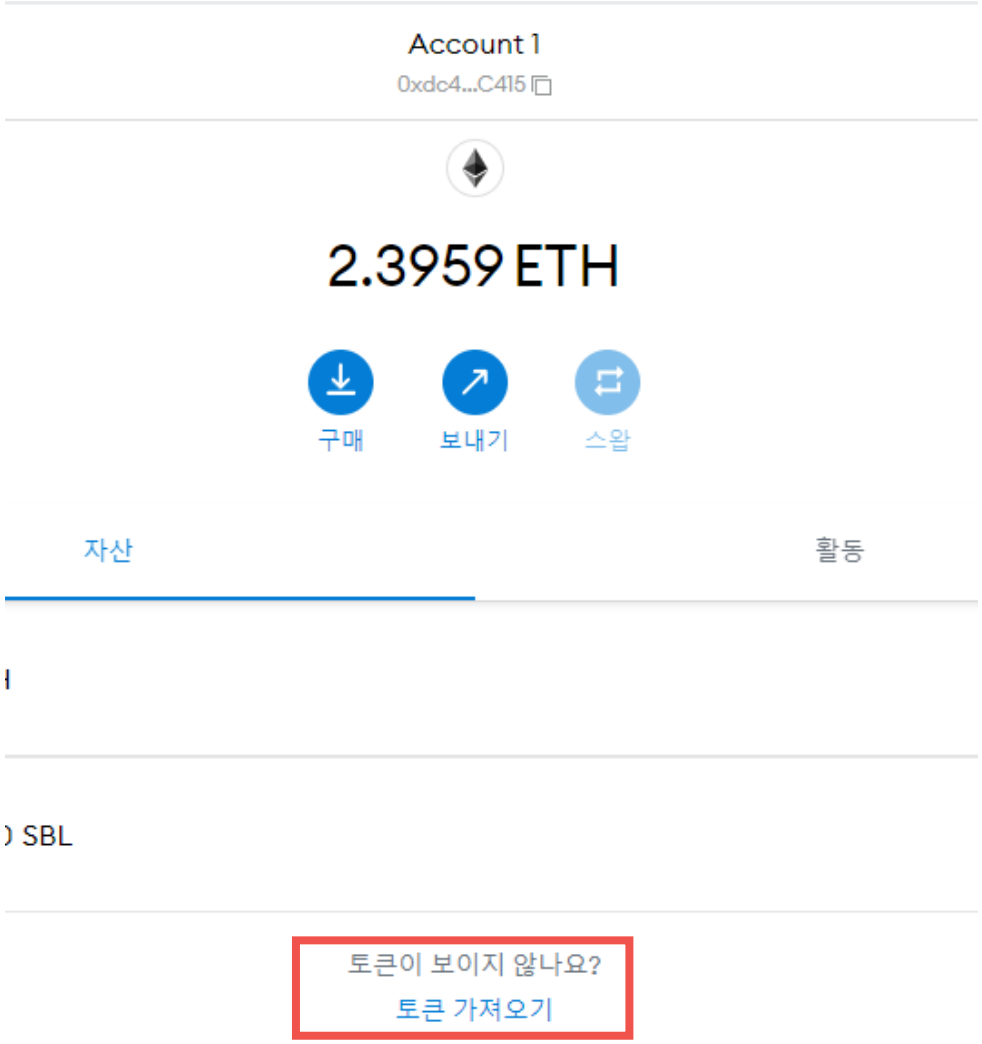
NFT 실습


mint
address to


mint
to:
0xdc45BbA03A4CE2c9b27294B5F74B88420269C415
transact

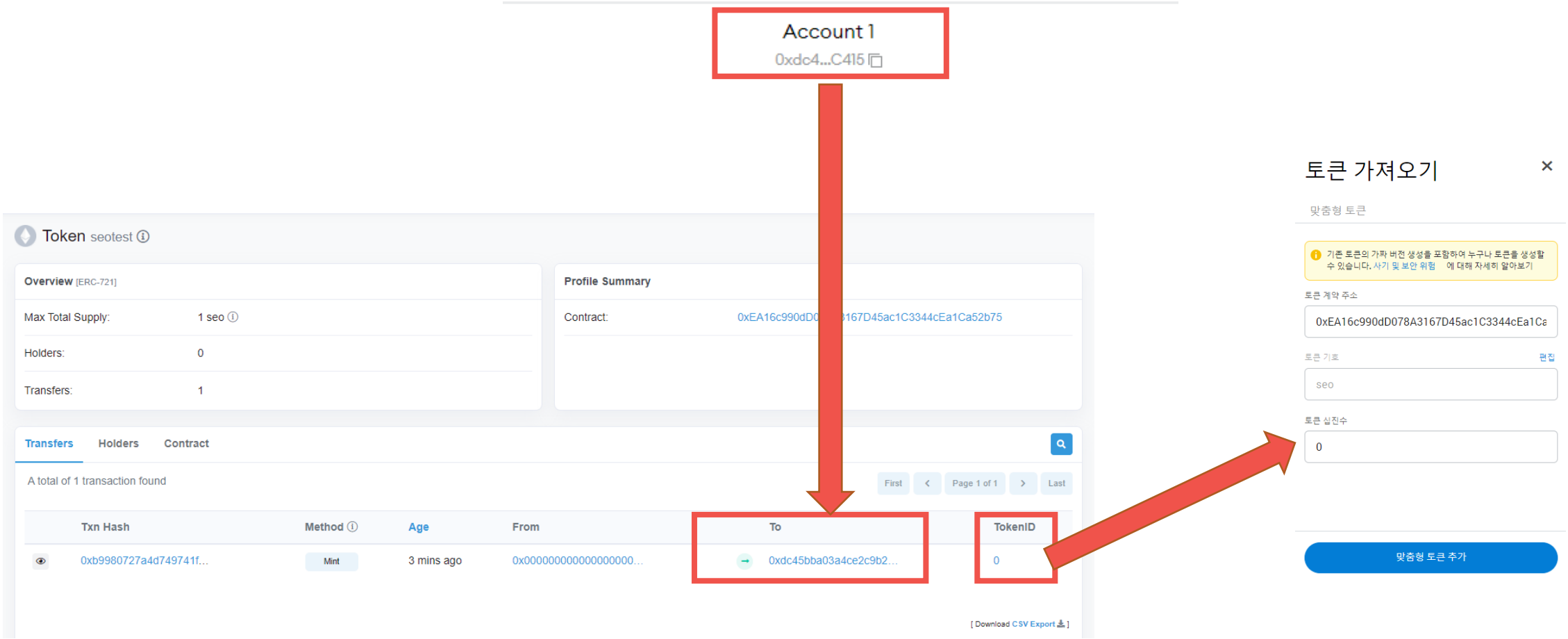

편집
예상 가스 요금
0.00031996
0.00032 ETH
제안 사이트
거의 < 15 초 후
최대 요금 0.00031996 ETH
합계
0.00031996
0.00031996 ETH
금액 + 가스 요금
최대 금액: 0.00031996 ETH
거부
확인

NFT 실습


Account 1
0xdc4...C415
2.3959 ETH
구매
보내기
스왑
자산
활동
) SBL
토큰이 보이지 않나요?
토큰 가져오기

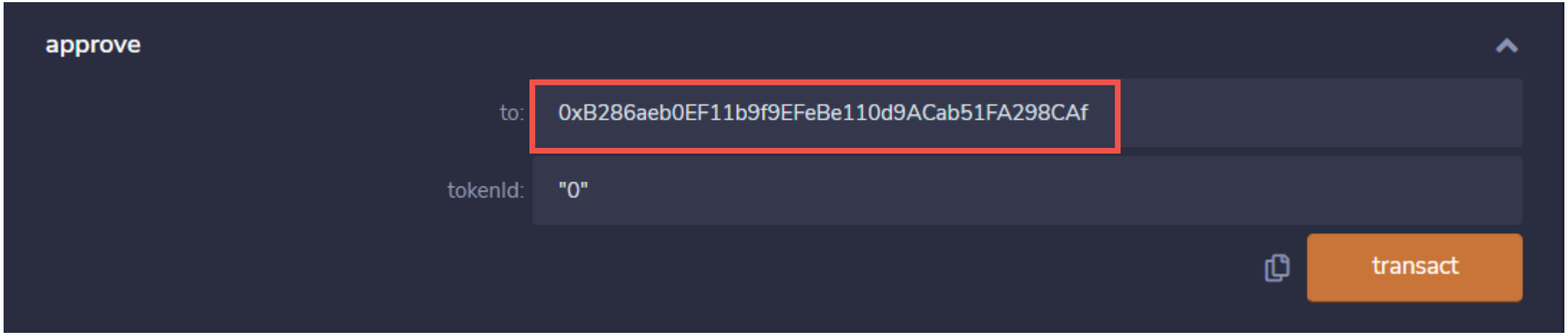

토큰 가져오기
맞춤형 토큰
기존 토큰의 가짜 버전 생성을 포함하여 누구나 토큰을 생성할 수 있습니다. 사기 및 보안 위험 에 대해 자세히 알아보기
토큰 계약 주소
0xEA16c990dD078A3167D45ac1C3344cEa1Ca
토큰 기호
편집
seo
토큰 십진수
필요한 토큰 십진수
토큰 십진수는 Etherscan 에서 찾을 수 있습니다.
맞춤형 토큰 추가

NFT 실습


Account 1
0xdc4...C415
Token seotest
Overview [ERC-721]
Max Total Supply: 1 seo
Holders: 0
Transfers: 1
Profile Summary
Contract: 0xEA16c990dD0 3167D45ac1C3344cEa1Ca52b75
Transfers
Holders
Contract
A total of 1 transaction found
First
Page 1 of 1
Last
Txn Hash
Method
Age
From
To
TokenID
0xb9980727a4d749741f...
Mint
3 mins ago
0x0000000000000000000...
0xdc45bba03a4ce2c9b2...
0
[Download CSV Export]
토큰 가져오기
맞춤형 토큰
기존 토큰의 가짜 버전 생성을 포함하여 누구나 토큰을 생성할 수 있습니다. 사기 및 보안 위험 에 대해 자세히 알아보기
토큰 계약 주소
0xEA16c990dD078A3167D45ac1C3344cEa1Ca
토큰 기호
편집
seo
토큰 십진수
0
맞춤형 토큰 추가

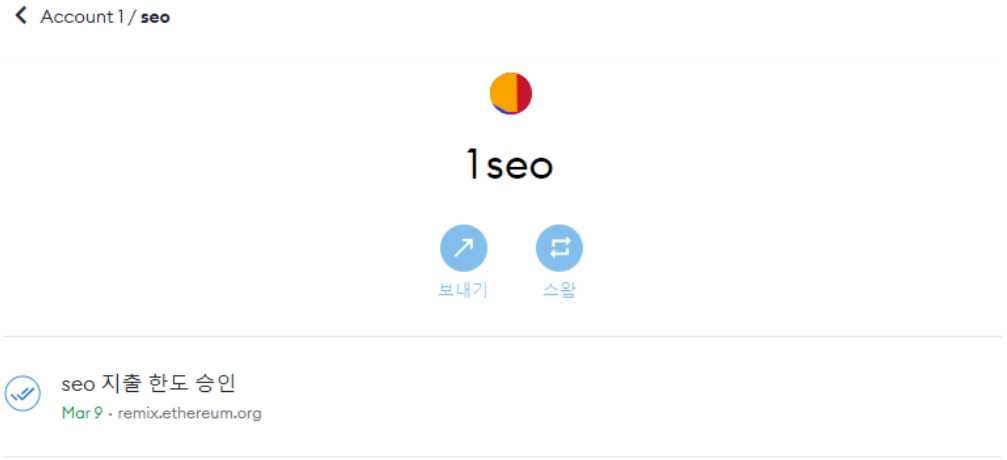
NFT 실습

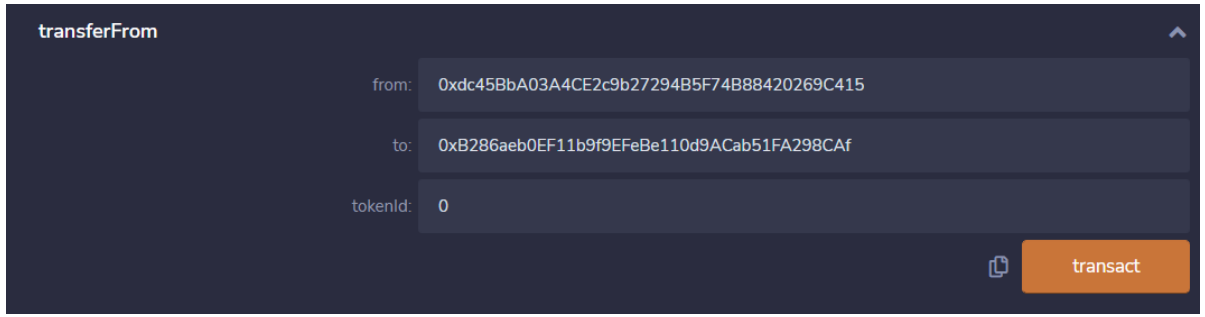
전송전 스마트 컨트랙트로 전송 승인을 해야됨
Address 에는 보내고자 하는 사람의 주소



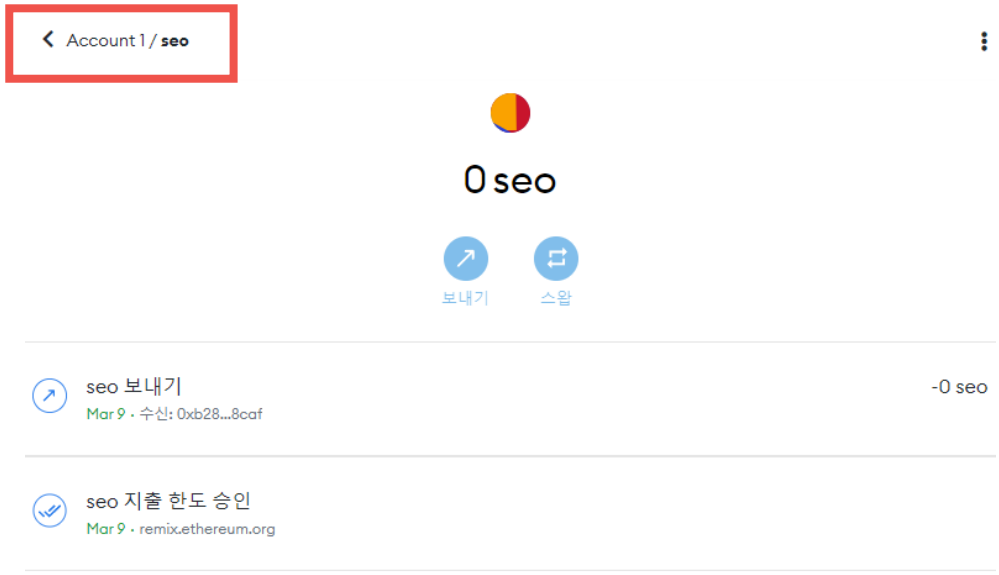
NFT 실습

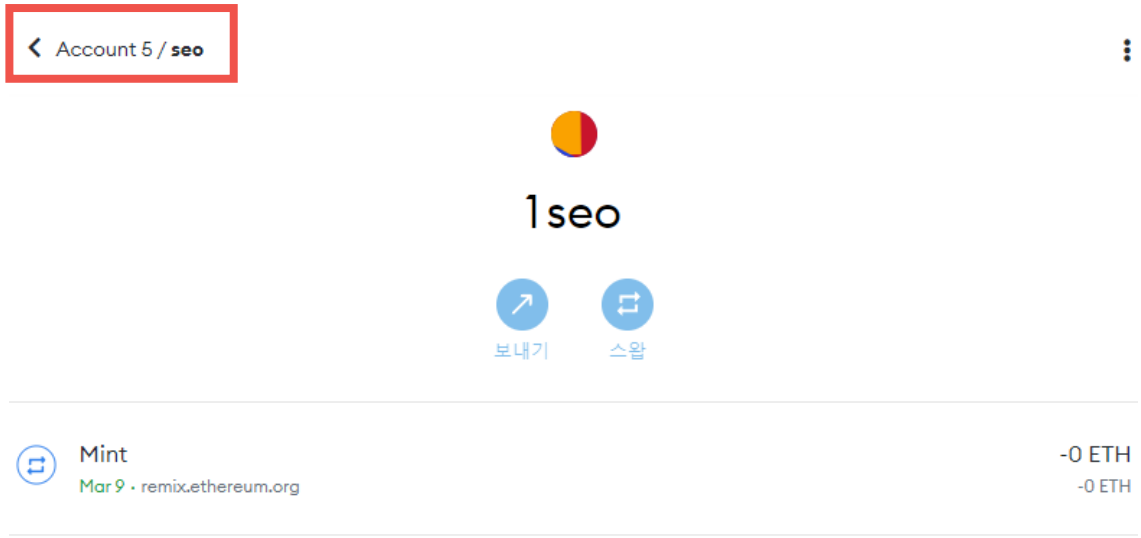
보내기 위해서는 스마트컨트랙트의 transferFrom 사용





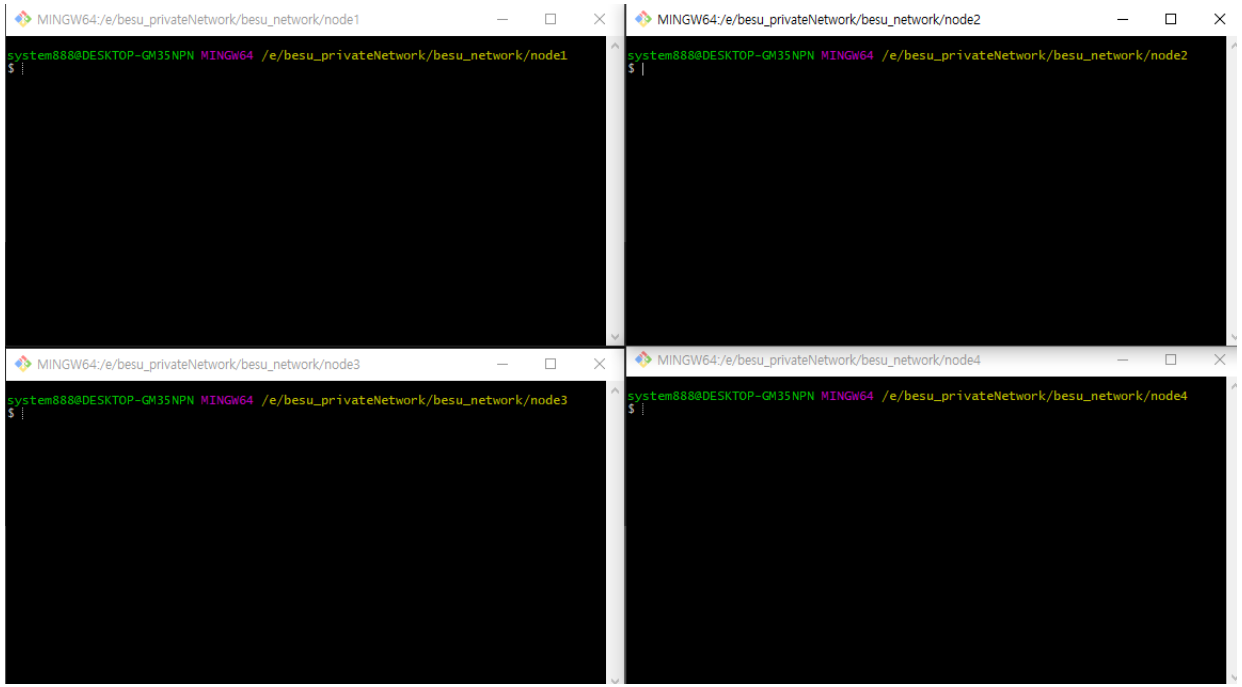
NFT 실습

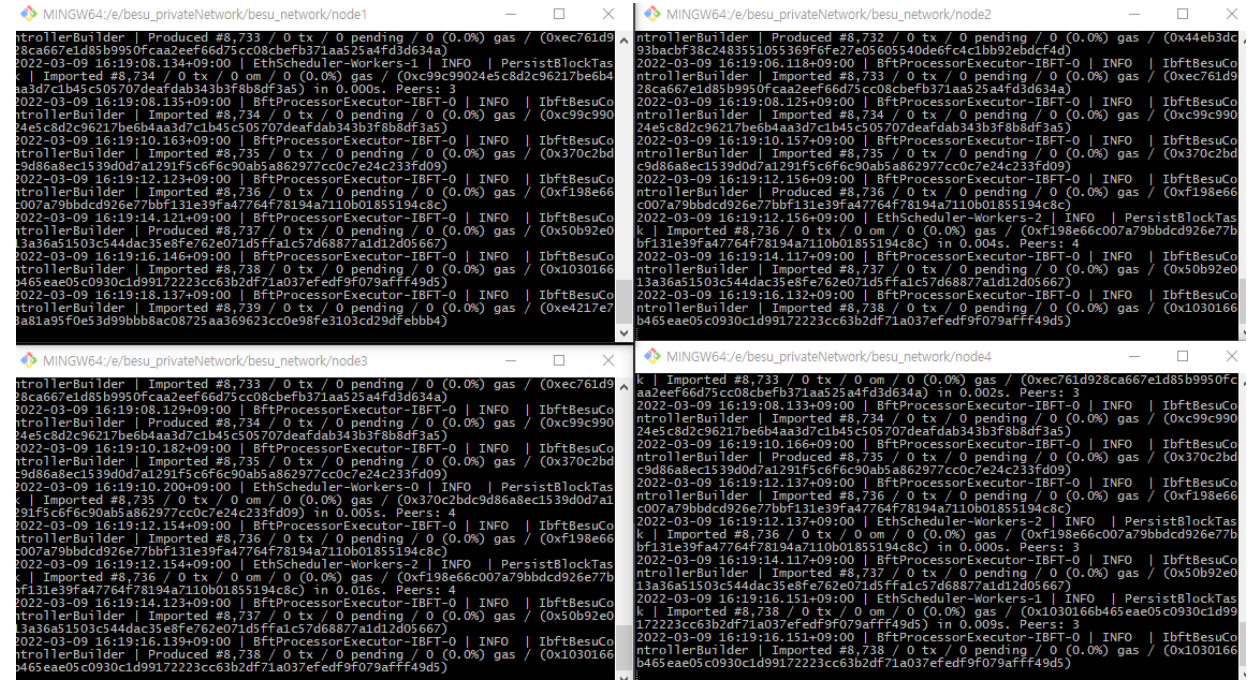

Account 1 / seo
0 seo
보내기
스왑
seo 보내기
Mar 9 · 수신: 0xb28...8caf
-0 seo
seo 지출 한도 승인
Mar 9 · remix.ethereum.org


Account 5 / seo
1 seo
보내기
스왑
Mint
Mar 9 · remix.ethereum.org
-0 ETH
-0 ETH

NFT 실습

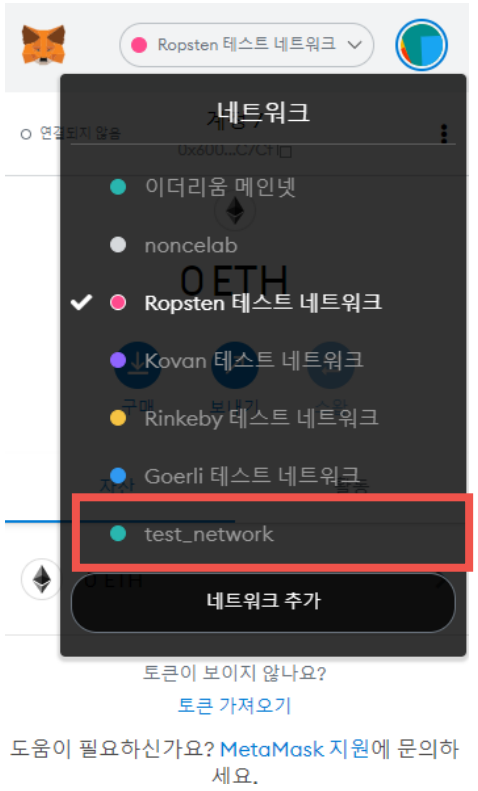
하이퍼레저 베수 구동

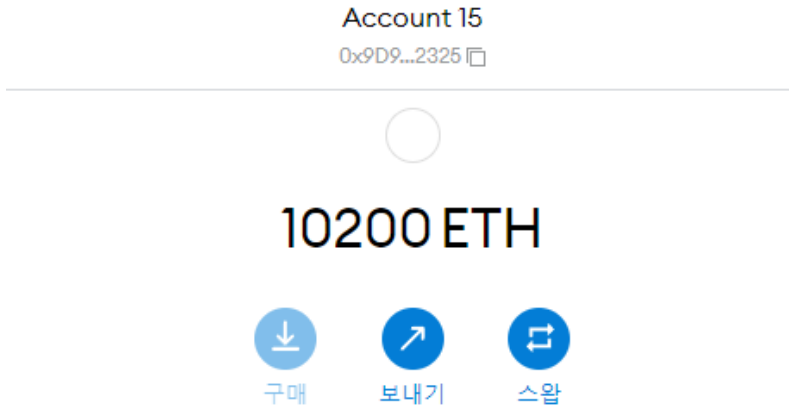




NFT 실습

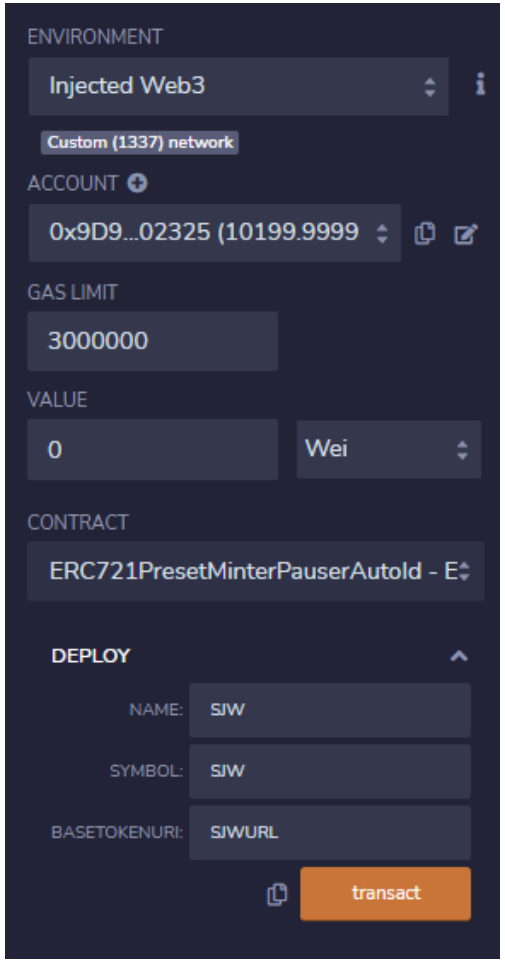
하이퍼레저 베수와 메타마스크 연결
이후, remix 연동



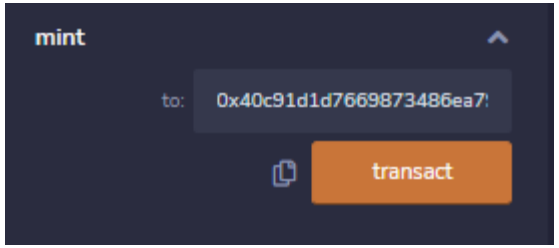


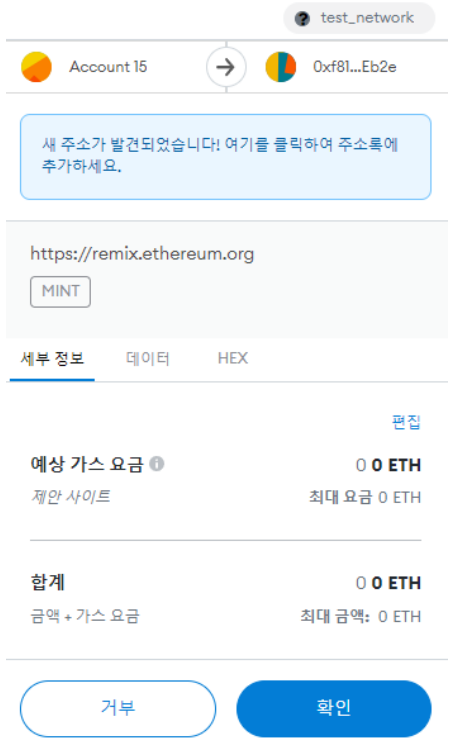
NFT 실습

스마트 컨트랙트 베포만 된거임 Mint를 통해서 다른사용자에게 Token ID 발행 필요

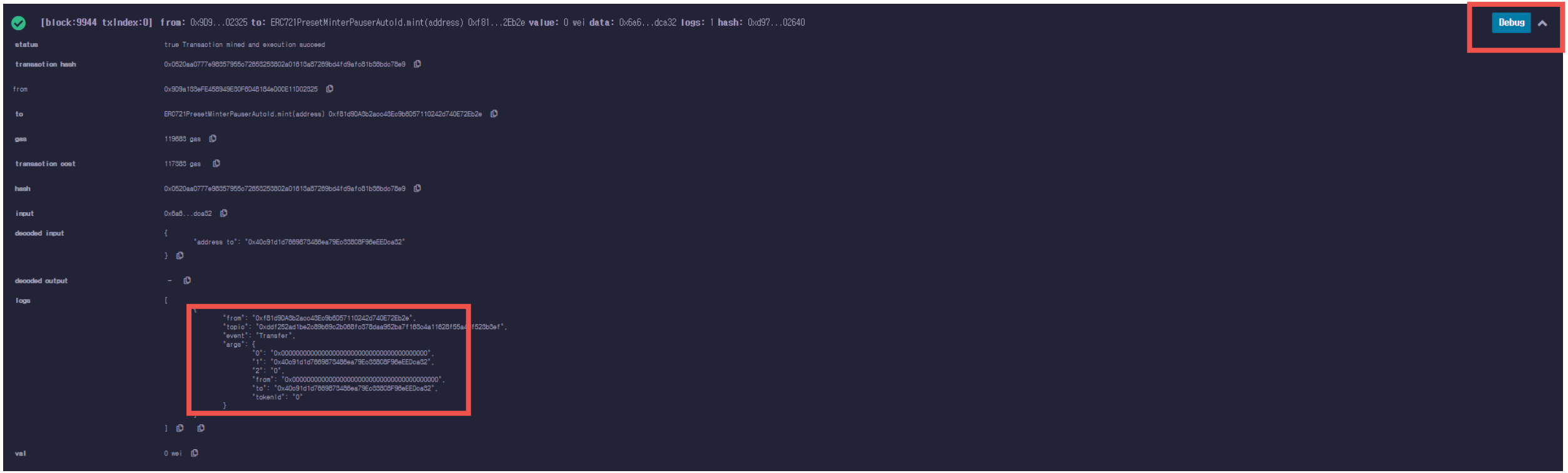


NFT 실습

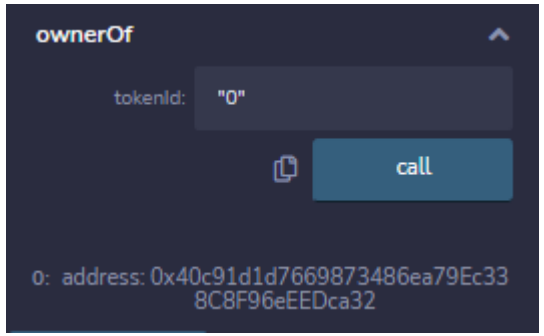

mint
to:
0x40c91d1d7669873486ea7
transact


test_network
Account 15
0xf81...Eb2e
새 주소가 발견되었습니다! 여기를 클릭하여 주소록에 추가하세요.
https://remix.ethereum.org
MINT
세부 정보
데이터
HEX
편집
예상 가스 요금
0 ETH
제안 사이트
최대 요금 0 ETH
합계
0 ETH
금액 + 가스 요금
최대 금액: 0 ETH
거부
확인

NFT 실습


[block:9944 txIndex:0] from: 0x9D9...02325 to: ERC721PresetMinterPauserAutoId.mint(address) 0xf81...2Eb2e value: 0 wei data: 0x6a6...dca32 logs: 1 hash: 0xd97...02640
Debug
status
true Transaction mined and execution succeed
transaction hash
0x0520aa0777e98357955e72658253802a01618a87269bd4fd9afe81b36bdc78e9
from
0x9D9a163eFE458949E30F6048184e000E11002325
to
ERC721PresetMinterPauserAutoId.mint(address) 0xf81d90A9b2aec43Ec9b6057110242d740E72Eb2e
gas
119683 gas
transaction cost
117683 gas
hash
0x0520aa0777e98357955e72658253802a01618a87269bd4fd9afe81b36bdc78e9
input
0x6a6...dca32
decoded input
{
"address to": "0x40c91d1d7869873486ea79Ec33808F96eEEDca32"
}
decoded output
-
logs
[
{
"from": "0xf81d90A9b2aec43Ec9b6057110242d740E72Eb2e",
"topic": "0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef",
"event": "Transfer",
"args": {
"0": "0x0000000000000000000000000000000000000000",
"1": "0x40c91d1d7869873486ea79Ec33808F96eEEDca32",
"2": "0",
"from": "0x0000000000000000000000000000000000000000",
"to": "0x40c91d1d7869873486ea79Ec33808F96eEEDca32",
"tokenId": "0"
}
}
]
val
0 wei

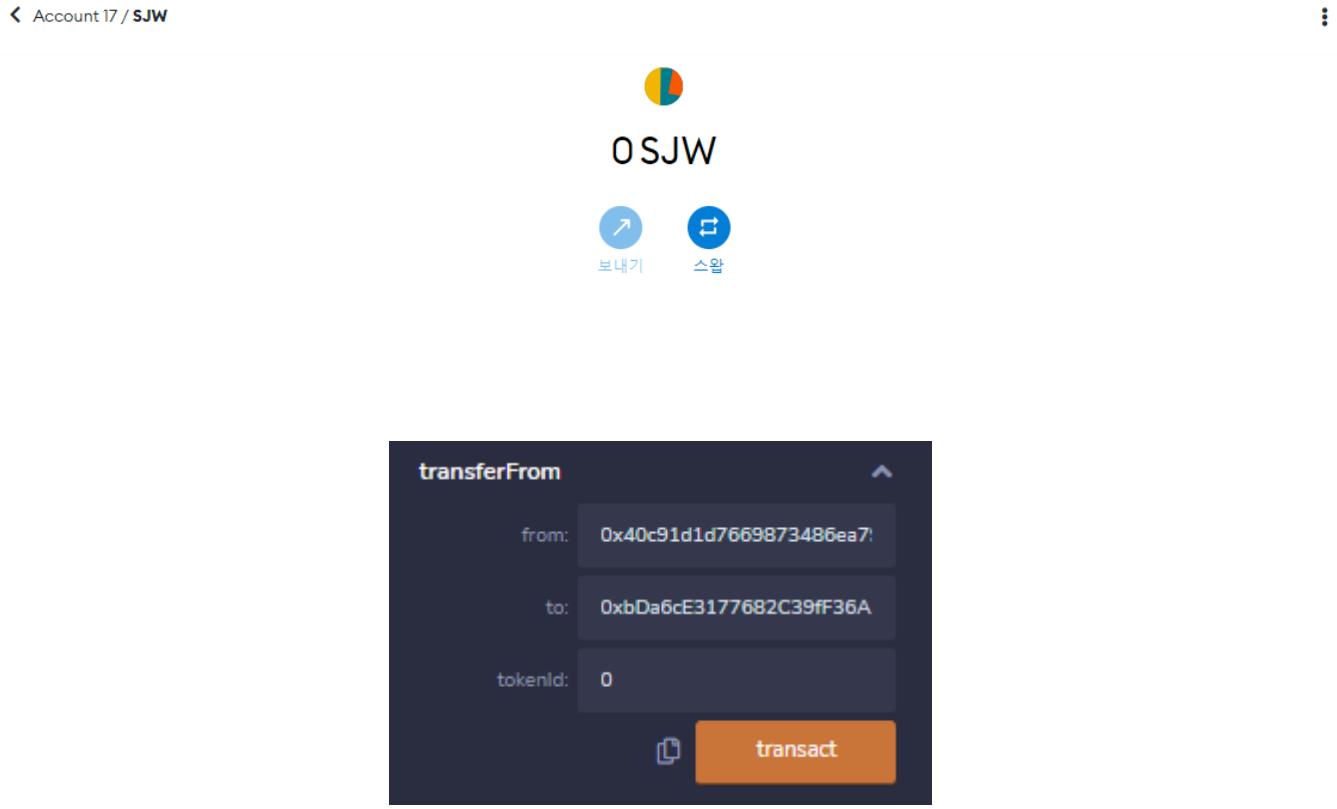
NFT 실습



토큰 아이디 입력

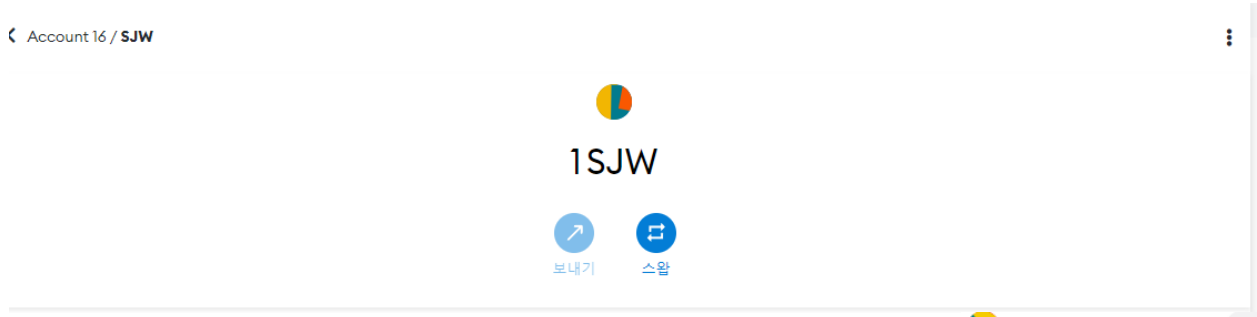
NFT 실습

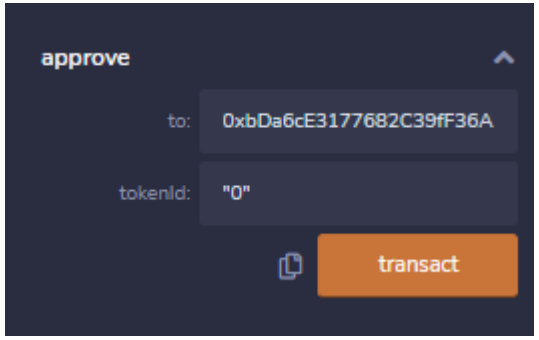
전송

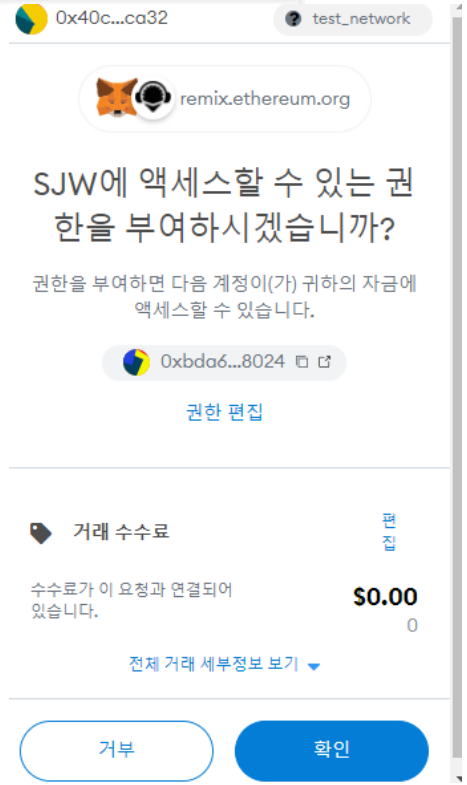

Account 17 / SJW
0 SJW
보내기
스왑
transferFrom
from:
0x40c91d1d7669873486ea7!
to:
0xbDa6cE3177682C39fF36A
tokenId:
0
transact

NFT 실습

토큰 접근권한 부여







받는사람에게 Token을 보내겠다고 승인 요청

Q & A