Lesson 8 - Tokenized Votes

The ERC20Votes ERC20 extension

- ERC20Votes properties
- Snapshots
- Creating snapshots when supply changes
- Using snapshots
- Self delegation
- Contract overall operation

References

https://docs.openzeppelin.com/contracts/4.x/api/token/erc20#ERC20Votes

https://docs.openzeppelin.com/contracts/4.x/api/token/erc20#ERC20Snapshot

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.4;

import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
import "@openzeppelin/contracts/token/ERC20/extensions/draft-
ERC20Permit.sol";
import "@openzeppelin/contracts/token/ERC20/extensions/ERC20Votes.sol";

contract MyToken is ERC20, ERC20Permit, ERC20Votes {
    constructor() ERC20("MyToken", "MTK") ERC20Permit("MyToken") {}

    // The following functions are overrides required by Solidity.

    function _afterTokenTransfer(address from, address to, uint256 amount)
        internal
        override(ERC20, ERC20Votes)
    {
        super._afterTokenTransfer(from, to, amount);
    }

    function _mint(address to, uint256 amount)
        internal
        override(ERC20, ERC20Votes)
    {
        super._mint(to, amount);
    }

    function _burn(address account, uint256 amount)
        internal
        override(ERC20, ERC20Votes)
    {
```

```
        super._burn(account, amount);
    }
}
```

ERC20Votes and Ballot.sol

- (Review) TDD
- Mapping scenarios
- Contracts structure

Homework

- Create Github Issues with your questions about this lesson
- Read the references
- (Optional) Study how ERC20Permit works https://docs.openzeppelin.com/contracts/4.x/api/token/erc20#ERC20Permit
- (Optional) Study and try out a full governance example from https://docs.openzeppelin.com/contracts/4.x/governance

Weekend project

- Form groups of 3 to 5 students
- Complete the contracts together
- Structure scripts to
    - Deploy everything
    - Interact with the ballot factory
    - Query proposals for each ballot
    - Operate scripts
- Publish the project in Github
- Run the scripts with a few set of proposals, play around with token balances, cast and delegate votes, create ballots from snapshots, interact with the ballots and inspect results
- Write a report detailing the addresses, transaction hashes, description of the operation script being executed and console output from script execution for each step
- (Extra) Use TDD methodology