CROWDSALE

pragma solidity ^0.5.0;

import "./KaseiCoin.sol";

import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/contracts/crowdsale/Crowdsale.sol";

import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/contracts/crowdsale/emission/MintedCrowdsale.sol";

// Have the KaseiCoinCrowdsale contract inherit the following OpenZeppelin:

// \* Crowdsale

// \* MintedCrowdsale

contract KaseiCoinCrowdsale is Crowdsale, MintedCrowdsale { // UPDATE THE CONTRACT SIGNATURE TO ADD INHERITANCE

// Provide parameters for all of the features of your crowdsale, such as the `rate`, `wallet` for fundraising, and `token`.

constructor(

uint256 rate,

address payable wallet,

KaseiCoin token

) public Crowdsale(rate, wallet, token) {

// constructor can stay empty

}

}

contract KaseiCoinCrowdsaleDeployer {

// Create an `address public` variable called `kasei\_token\_address`.

address public kasei\_token\_address;

// Create an `address public` variable called `kasei\_crowdsale\_address`.

address public kasei\_crowdsale\_address;

// Add the constructor.

constructor(

string memory name,

string memory symbol,

address payable wallet

) public {

// Create a new instance of the KaseiCoin contract.

KaseiCoin token = new KaseiCoin(name, symbol, 0);

// Assign the token contract’s address to the `kasei\_token\_address` variable.

kasei\_token\_address = address(token);

// Create a new instance of the `KaseiCoinCrowdsale` contract

KaseiCoinCrowdsale kasei\_crowdsale =

new KaseiCoinCrowdsale(1, wallet, token);

// Aassign the `KaseiCoinCrowdsale` contract’s address to the `kasei\_crowdsale\_address` variable.

kasei\_crowdsale\_address = address(kasei\_crowdsale);

// Set the `KaseiCoinCrowdsale` contract as a minter

token.addMinter(kasei\_crowdsale\_address);

// Have the `KaseiCoinCrowdsaleDeployer` renounce its minter role.

token.renounceMinter();

}

}

Kasei Coin

pragma solidity ^0.5.0;

// Import the following contracts from the OpenZeppelin library:

// \* `ERC20`

// \* `ERC20Detailed`

// \* `ERC20Mintable`

import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/contracts/token/ERC20/ERC20.sol";

import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/contracts/token/ERC20/ERC20Detailed.sol";

import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/contracts/token/ERC20/ERC20Mintable.sol";

// Create a constructor for the KaseiCoin contract and have the contract inherit the libraries that you imported from OpenZeppelin.

contract KaseiCoin is ERC20, ERC20Detailed, ERC20Mintable {

constructor(

string memory name,

string memory symbol,

uint initial\_supply

)

ERC20Detailed(name,symbol,18)

public {

// constructor can stay empty

}

}