



DEPLOY & RUN TRANSACTIONS ✓ >

decreaseA...

address spender, uint256 subtr

▼

mint

address account, uint256 amou

▼

increaseAL...

address spender, uint256 addre

▼

renounce...

▼

transfer

address recipient, uint256 amou

▼

transferFr...

address sender, address recipie

▼

allowance

address owner, address spende

▼

balanceOf

address account

▼

decimals

▼

isMinter

address account

▼

name

▼

symbol

▼

totalSupply

▼

Low level interactions

i

CALLDATA

Transact

▼ KASEICOINCROWDSALE AT 0X7EE

✕

Balance: 0 ETH

buyTokens

address beneficiary

▼

rate

token

wallet

weiRaised

Low level interactions

i

CALLDATA

Transact

▼ KASEICOINCROWDSALEDEPLOYEI

✕

Balance: 0 ETH

kasei_cro...

0: address: 0xb0EDE5A9Fc4C6c81C613D9660235A1AE4542fED7

kasei_toke...

0: address: 0xF0640a7741B30c451848E56710D7282d954E6345

Low level interactions

i

CALLDATA

Transact

▶

🔍

🔍

Home

KaseiCoin.sol

KaseiCoinCrowdsale.sol ✕

↔

```
1 pragma solidity ^0.5.0;
2
3 import "./KaseiCoin.sol";
4 import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/
5 import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/
6
7
8 // Have the KaseiCoinCrowdsale contract inherit the following OpenZeppelin:
9 // * Crowdsale
10 // * MintedCrowdsale
11 contract KaseiCoinCrowdsale is Crowdsale, MintedCrowdsale{ // UPDATE THE CONTRACT S
12
13     // Provide parameters for all of the features of your crowdsale, such as the `r
14     constructor(
15         uint256 rate,
16         address payable wallet,
17         KaseiCoin token
18     ) public Crowdsale(rate, wallet, token) {
19         // constructor can stay empty
20     }
21 }
22
23
24 contract KaseiCoinCrowdsaleDeployer {
25     // Create an `address public` variable called `kasei_token_address`.
26     address public kasei_token_address;
27     // Create an `address public` variable called `kasei_crowdsale_address`.
28     address public kasei_crowdsale_address;
29
30     // Add the constructor.
31     constructor(
32         string memory name,
33         string memory symbol,
34         address payable wallet
35     )
36
37     public {
38         // Create a new instance of the KaseiCoin contract.
39         // Assign the token contract's address to the `kasei_token_address` variabl
40
41         // Create a new instance of the `KaseiCoinCrowdsale` contract
42         KaseiCoin token = new KaseiCoin(name, symbol, 0);
43         kasei_token_address = address(token);
```

▼

🔇

0

☐ listen on all transactions

🔍

Search with transaction hash or address

-

data: 0x000...c13c/ logs: 0 hash: 0x000...e410e

creation of KaseiCoinCrowdsaleDeployer pending...

✓

[block:8 txIndex:0] from: 0x57E...c3bCf to: value: 0 wei

Debug ▼

>