

COINGT ERC20 Token Contract

Overview

The COINGT token is an ERC20-compliant smart contract deployed on the Ethereum blockchain. It is designed to serve as a secure, transparent, and efficient digital asset. The contract adheres to industry standards and has been well-documented to meet regulatory and operational requirements.

This document outlines the contract's features, functionalities, and transaction history, providing clarity for regulators and stakeholders.

Key Features

- **ERC20 Standard Compliance:** Implements all mandatory functions (`transfer`, `approve`, `transferFrom`, `balanceOf`, etc.) and events (`Transfer`, `Approval`), ensuring compatibility with Ethereum wallets and applications.
- **Transparent Documentation:** The contract is fully commented, explaining the functionality and purpose of each component for easy auditing and regulatory review.
- **Minting and Burning:** Includes owner-controlled minting and burning mechanisms to manage the token supply effectively.
- **Secure Access Control:** Utilizes the `Ownable` module to restrict sensitive operations like minting and burning to the contract owner.
- **Event Logging:** Emits ERC20 standard events (`Transfer`, `Approval`) for transparency and traceability on the blockchain.

Technical Details

Token Metadata

- **Token Name:** COINGT
- **Token Symbol:** COINGT
- **Decimals:** 0 (integer-based token for simplicity)
- **Initial Supply:** 195,770 tokens minted to the deployer upon deployment
- **Contract Address:** [0x7a1Eb5e5F2e1587D2DC3350Ac51E57f46dc28C32](#)

Functionalities

1 • Basic ERC20 Functions:

- `transfer(address recipient, uint256 amount)`: Transfers tokens to a recipient.
- `approve(address spender, uint256 amount)`: Approves a spender to use a specific amount of tokens.
- `transferFrom(address sender, address recipient, uint256 amount)`: Transfers tokens using an allowance mechanism.
- `balanceOf(address account)`: Returns the token balance of a specific account.
- `totalSupply()`: Returns the total supply of tokens.

2 • Owner-Only Functions:

- `mint(address account, uint256 amount)`: Mints new tokens and assigns them to a specific account.
- `burn(address account, uint256 amount)`: Burns tokens, reducing the total supply.
- `transferOwnership(address newOwner)`: Transfers contract ownership to a new address.
- `renounceOwnership()`: Renounces ownership, leaving the contract without an owner.

3 • Allowance Management:

- `increaseAllowance(address spender, uint256 addedValue)`: Increases a spender's allowance.
- `decreaseAllowance(address spender, uint256 subtractedValue)`: Decreases a spender's allowance.

Transaction History

Burning Event

As part of the token's lifecycle management, a **burning event** was conducted to reduce the token supply. This demonstrates responsible supply control and a commitment to maintaining value.

- **Transaction Hash:** [0x57d1577ddda25acbe9d6ef525bf106b7b7fc79093160f77b220380c129c98c75](#)
- **Tokens Burned:** 42,546
- **Date:** November 22/2024.

Security Features

- **No Overflow/Underflow Risks:** Built-in Solidity 0.8.x protections eliminate the need for external libraries like `SafeMath`.
- **Zero Address Checks:** Ensures no tokens are transferred, minted, or burned to/from the zero address.
- **Event Logging:** Tracks all token transfers and approvals for transparency.
- **Well-Documented Code:** Inline comments improve clarity and provide a detailed understanding of the contract's purpose and functionality.

Deployment and Verification

- **Blockchain:** Ethereum Mainnet
- **Contract Address:** [0x7a1Eb5e5F2e1587D2DC3350Ac51E57f46dc28C32](#)
- **Source Code:** Verified and available on Etherscan.

For any questions or further clarifications, please contact the team here: info@coingt.com