Using thirdweb

Create Contract

To create a new smart contract using thirdweb CLI, follow these steps:

1. In your CLI run the following command:
2. npx thirdweb create contract
3. Input your preferences for the command line prompts:
4.
    1. Give your project a name
5.
    1. Choose your preferred framework: Hardhat or Foundry
6.
    1. Name your smart contract
7.
    1. Choose the type of base contract: EmptyERC20
8.
    1. ,ERC721
9.
    1. , orERC1155
10.
    1. Add any desiredextensions
11. Once created, navigate to your project's directory and open in your preferred code editor.
12. If you open thecontracts
13. folder, you will find your smart contract; this is your smart contract written in Solidity.
14. The following is code for an ERC721Base contract without specified extensions. It implements all of the logic inside theERC721Base.sol
15. contract; which implements theERC721A
16. standard.
17. // SPDX-License-Identifier: MIT
18. pragma solidity ^0.8.0
19. ;
20. import
21. "@thirdweb-dev/contracts/base/ERC721Base.sol"
22. ;
23. contract Contract is ERC721Base
24. {
25. constructor
26. (
27. string memory _name,
28. string memory _symbol,
29. address _royaltyRecipient,
30. uint128 _royaltyBps
31.)
32. ERC721Base
33. (
34. _name, _symbol, _royaltyRecipient, _royaltyBps
35.)
36. {
37. }
38. }
39. This contract inherits the functionality of ERC721Base through the following steps:
40.
    - Importing the ERC721Base contract
41.
    - Inheriting the contract by declaring that our contract is an ERC721Base contract
42.
    - Implementing any required methods, such as the constructor.
43. After modifying your contract with your desired custom logic, you may deploy it to Celo usingDeploy
44. .

Alternatively, you can deploy a prebuilt contract for NFTs, tokens, or marketplace directly from the thirdweb Explore page:

1. Go to the thirdweb Explore pagehttps://thirdweb.com/explore
2. Choose the type of contract you want to deploy from the available options: NFTs, tokens, marketplace, and more.
3. Follow the on-screen prompts to configure and deploy your contract.

For more information on different contracts available on Explore, check outthirdweb's documentation.

Deploy Contract

Deploy allows you to deploy a smart contract to any EVM compatible network without configuring RPC URLs, exposing your private keys, writing scripts, and other additional setup such as verifying your contract.

1. To deploy your smart contract using deploy, navigate to the root directory of your project and execute the following command:
2. npx thirdweb deploy
3. Executing this command will trigger the following actions:
4.
    - Compiling all the contracts in the current directory.
5.
    - Providing the option to select which contract(s) you wish to deploy.
6.
    - Uploading your contract source code (ABI) to IPFS.
7. When it is completed, it will open a dashboard interface to finish filling out the parameters.
8.
    - _name
9.
    - : contract name
10.
    - _symbol
11.
    - : symbol or "ticker"
12.
    - _royaltyRecipient
13.
    - : wallet address to receive royalties from secondary sales
14.
    - _royaltyBps
15.
    - : basis points (bps) that will be given to the royalty recipient for each secondary sale, e.g. 500 = 5%
16. Select Celo as the network
17. Manage additional settings on your contract's dashboard as needed such as uploading NFTs, configuring permissions, and more.

For additional information on Deploy, please referencethirdweb's documentation .

If you have any further questions or encounter any issues during the process, please reach out to thirdweb support atsupport.thirdweb.com . Edit this page Previous Deploy on Celo Next Deploy with Remix