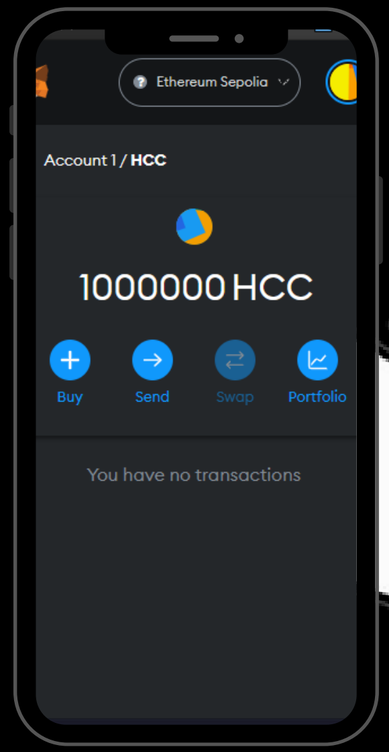
Learn to make your own crypto currency using solidity!

Guide making your own club Reward System



Easy Guide to Beginners and Pros alike!

Agenda

Before we will learn how to create your own token, let's get prepared.

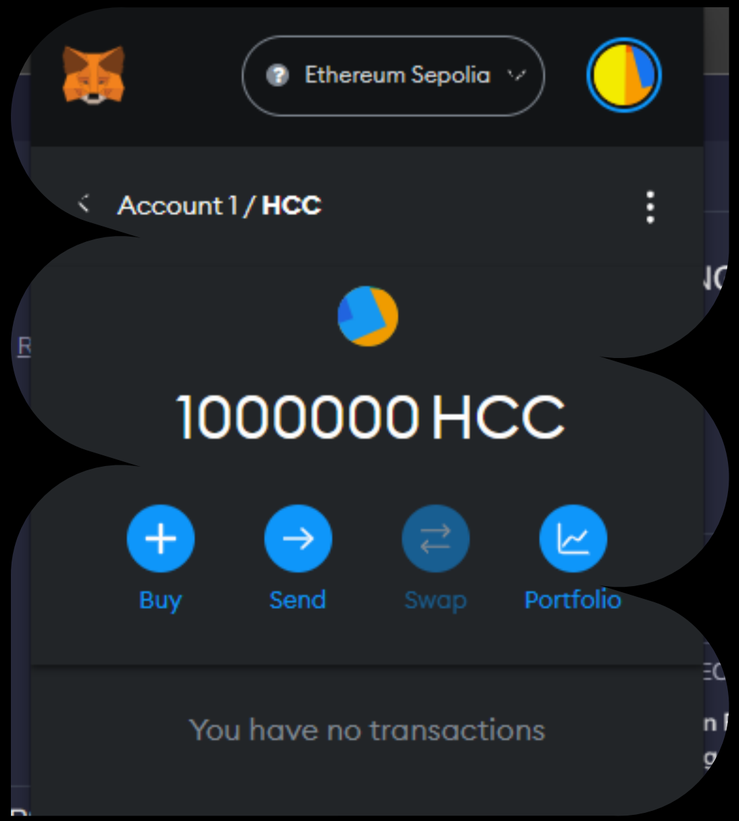
01

To create a crypto token (duh!)

02

Understand the basics of Solidity language

Hey There, Hacker!



This is an example of a token I made!

In this jam, you'll be making your **very own cryptocurrency which can be shared among your friends or club mates!**

This task isnt really time consuming and you'll also learn something new on the way!

Why do you need one?

Use this token between your friends and especially your club mates

This jam has been designed with clubs in mind, with that beening said; why not make your own token as a form of gratitude or completion for your club? Let it be a form of gamification of tasks to make things better. Or you can use this as the perfect opportunity to learn something new and make something rewarding!

The Outline: Steps to our Goal

Follow these five steps and let's finish this!

1. Creating A Endpoint With Quicknode
2. Using Faucets for Gas Fees
3. The Coding and Creation
4. Deployment & Verifying your coins
5. Sharing your coins!

Wait!

Before you start, we need to make an metamask account!

Metamask is like an virtual wallet where you will be storing your coins!

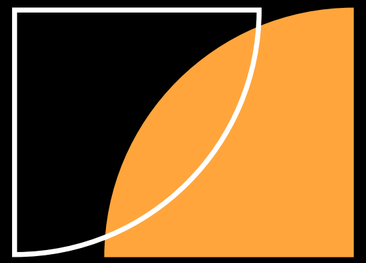
Create Your Accounts

Downloading the browser extension

1. Launch your browser, and search for Metamask Extension or Metamask.io.
2. Click Metamask, and download it on your browser.

Creating the Account

1. Once you download the Metamask extension, click “Get Started” to create an account.
2. On the two options provided, choose “Create Wallet”.
3. Enter a strong password for your account.
4. You will be shown a random 12-word secure phrase, make sure to write it down or save it using a secure password manager.
5. Click the favicon (the circular account image) at the top right of your wallet.
6. Click 'Create Account'.
7. Enter your preferred name and then hit 'Create' to confirm



Now Let's Get Started by making an endpoint in an website called Quicknode

1. Navigate to quicknode.com and create an account

Create Your Endpoint

What to fill in the metamask form

- Network Name: Ethereum Sepolia
- New RPC URL: [Paste the copied HTTP link]
- Chain ID: 11155111
- Currency Symbol: ETH
- Block Explorer URL: https://sepolia.etherscan.io

After making your account,

1. Click "Endpoints"
2. Click "Create Endpoint"
3. Click "Continue"
4. Click on "Ethereum"
5. Click on "Sepolia"
6. Choose the free version
7. Wait for the process to complete

After making your endpoint,

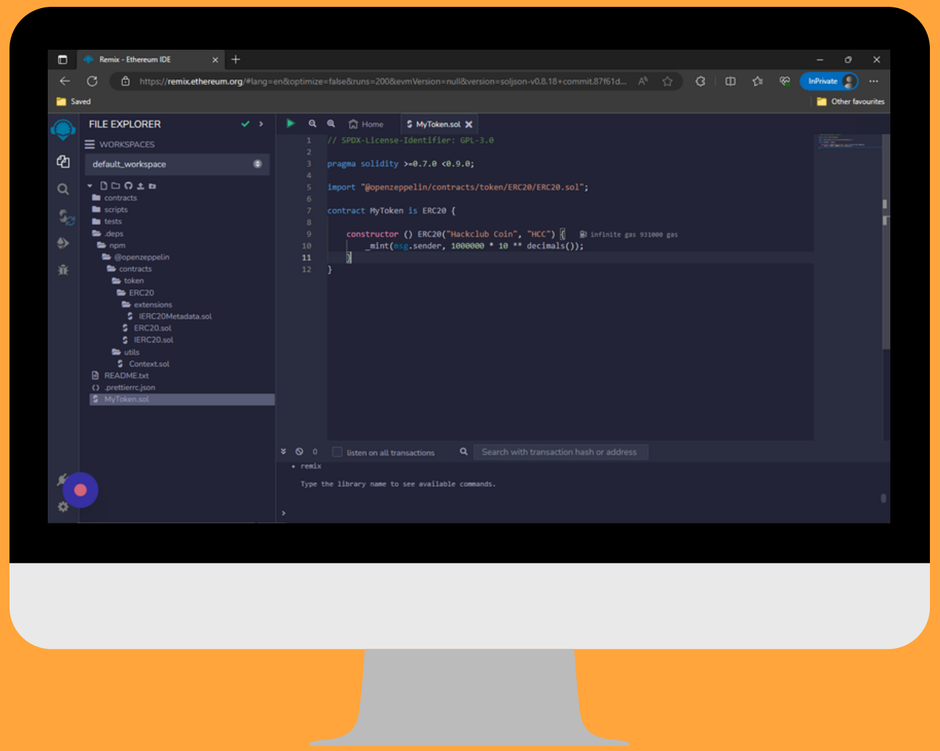
1. Copy the HTTP link provided at the top
2. Open MetaMask and click on the profile button
3. Click "Add Network"
4. Scroll to the bottom and click "Add a Network Manually"
5. Fill out the form as follows:
6. Click "Save" or "Add" to add the network
7. Press the text showing "Account 1" to copy the MetaMask address

Head on to this site:

sepoliafaucet.com

1. Paste the copied MetaMask address into the provided field on the Sepolia Faucet website
2. Follow the instructions on the website to claim the free tokens for token creation fees

The Code prep.



Creating the code and file.

1. Go to remix.ethereum.org in your web browser (Now you're a real crypto artist! Grab your virtual paintbrush and start creating your masterpiece.)
2. Create a new file called MyToken.sol and add the following code from the next slide

Deployment

1. Click on the third icon in the sidebar to access the Solidity compiler
2. Click on "Compile MyToken.sol" and wait for the compilation to finish
3. Click on the fourth icon in the sidebar to access the Deploy tool (Deploy, deploy, and away! It's like sending your token into the crypto universe. Ready for takeoff!)
4. Click on the environment dropdown and choose "Injected provider - Metamask" (You may need to refresh the page and the name may not specifically mention MetaMask)
5. Click on "Deploy" (Waiting for confirmation? Don't worry, it's not like waiting for a pizza delivery, It wont take long!)
6. Wait for the MetaMask popup to appear and accept any prompts that show up (Buckle up, folks! You're about to embark on a digital transaction journey. Fasten your seatbelts and prepare for takeoff!)

The Code & creation

Copy from the jam to get it more accurate

Example (Default) Code

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.7.0 <0.9.0;

 import"@openzeppelin/contracts/token/ERC20/ERC20.sol";

contract MyToken is ERC20 {

        constructor () ERC20("Your Coin Name", "Your coin symbol (3-4 letters only)") { _mint(msg.sender, 1000000 * 10 ** decimals());
 } }
```

Code explaination:

the first line is a comment which states the license of the code

the second line specifies the version of solidity the code is compatible with

the third line imports code from an open source library called openzeppelin

the forth line defines a new contract called MyToken from the ERC20 contract t

he next few lines defines the personality of the coin like its name, symbol, amount and decimal points. psss.. BTW dont remove the " " from around your token name and symbol.

Verification

1. Scroll down in the Deploy and Run Transactions tool to find the "Deployed Contracts" section
2. Click on the copy icon to copy the deployed contract's address
3. Go to sepolia.etherscan.io in your web browser
4. Paste the copied contract address into the search bar on Etherscan and verify its existence (Let's play detective! Paste the contract address and see if our virtual creation has left its digital footprint on the blockchain.)

Importing

1. Go back to MetaMask (Time to check on your tokens like a dragon guarding its treasure!)
2. Click on "Import Tokens" at the bottom of the page
3. Paste the deployed contract's address (same one used in sepolia.etherscan.io) and wait for the form to auto-complete

1. Continue and wait for the token to be imported
2. You will receive your coins in MetaMask (Cha-ching! It's raining digital coins in your MetaMask wallet. Time to celebrate!)

Verify

(This is where you check if you did things right!)

Personalizing options!

Try experimenting and testing different blockchains like matic!

Try making more coins for different occasions

Try messing around the settings!
It wont hurt to try new things!

Thank you maker.

That's right! By the end of this jam you have prolly made something of purpose making you, a maker.