Aim: Create your own wallet using Metamask for crypto transactions

Objective:

- To understand MetaMask and Ganache Wallets
- To understand how the transaction works on Ethereum.

Requirement:

- Web browser
- MetaMask extension
- Ganache Tool

Ganache

Ganache is a powerful tool for Ethereum developers, enabling them to efficiently build, test, and debug their applications in a local environment, saving time and resources while ensuring code quality before deployment to a live network.

Ganache is a personal blockchain for Ethereum development. In simpler terms, it's a tool that allows you to run a simulated Ethereum blockchain right on your own computer, specifically for building, testing, and debugging decentralized applications (dApps) and smart contracts.

Key features of Ganache:

- **Local blockchain:** It creates a private, local Ethereum blockchain for development.
- **Multiple accounts:** You can create and manage multiple accounts within the wallet.
- **Faucet:** It has a built-in faucet to generate test Ether for your accounts.
- **RPC server:** Provides an RPC server for interacting with the local blockchain.
- **Customization:** Allows you to customize the blockchain's settings, such as block time and gas limit.

Gas Limit:

- **Definition:** The maximum amount of computational work a transaction is allowed to perform on the Ethereum network.
- **Purpose:** Prevents infinite loops or malicious code from consuming excessive resources.
- **Units:** Measured in units of gas.

- **Setting:** You can set the gas limit for each transaction in Ganache.

Gas Price:

- **Definition:** The amount of Ether you're willing to pay for each unit of gas consumed by a transaction.
- **Purpose:** Determines how quickly your transaction will be processed. Higher gas prices typically result in faster confirmation times.
- **Units:** Measured in Gwei (one billionth of an Ether).
- **Setting:** You can set the gas price for each transaction in Ganache.

Relationship between Gas Limit and Gas Price:

- **Total Cost:** The total cost of a transaction is calculated by multiplying the gas used by the gas price.
- **Insufficient Gas:** If a transaction runs out of gas before completing, it will be reverted, and you'll lose the gas fee.

In essence, Ganache is a valuable tool for developers to experiment with Ethereum smart contracts and applications without risking real funds.

Creating and Testing Ganache Wallet with Metamask for Transactions

To create a local Ethereum blockchain using Ganache and test it with MetaMask for transactions, follow these steps:

Step 1: Install Ganache

1. **Download Ganache**:
    - Go to the Ganache official website (https://archive.trufflesuite.com/ganache/) and download the installer for your operating system (Windows, macOS, or Linux).
2. **Install Ganache**:
    - Run the installer and follow the instructions to install Ganache on your computer.

Step 2: Run Ganache

1. **Open Ganache**:

    - Launch Ganache from your applications menu.
2. **Create a New Workspace**:
    - Click on **"Quickstart Ethereum"** to create a new local blockchain network.
    - Ganache will automatically create a local Ethereum blockchain with few test accounts preloaded with 100 ETH each.
3. **View Blockchain Information**:
    - Note down the **RPC Server URL** (usually `http://127.0.0.1:7545`), which you will need to connect MetaMask to this local network.

Step 3: Install MetaMask

1. **Download MetaMask**:
    - Go to the MetaMask website and install the MetaMask browser extension for Chrome, Firefox, Edge, or Brave.
2. **Set Up MetaMask**:
    - Open the MetaMask extension.
    - Click "Get Started" and either import an existing wallet or create a new one. If creating a new wallet, follow the setup instructions and securely store your seed phrase.

Step 4: Connect MetaMask to Ganache

1. **Open MetaMask**:
    - Click on the MetaMask extension icon in your browser.
2. **Add a New Network**:
    - Click on the network dropdown (usually says "Ethereum Mainnet") at the top of the MetaMask window.
    - Click on **"Add Network"**.
3. **Configure the Ganache Network**:
    - Fill in the network details as follows:
        - **Network Name**: `Ganache Localhost`
        - **New RPC URL**: `http://127.0.0.1:7545` (this is the default RPC Server URL from Ganache)
        - **Chain ID**: `1337` (or `5777`, depending on Ganache's default settings; check the Ganache UI for the correct Chain ID)
        - **Currency Symbol**: `ETH`
    - Click **"Save"** to add the network.

Step 5: Import a Ganache Account to MetaMask

1. **Copy a Private Key from Ganache**:
    - In the Ganache UI, you will see a list of accounts with addresses and private keys.
    - Click on the key icon next to any account to copy its private key.
2. **Import the Account into MetaMask**:
    - In MetaMask, click on the circular account icon in the top right corner.
    - Scroll down and select "Add Account".
    - Under “Import a wallet or account”, choose "Private Key" as the import type.
    - **Paste the private key** you copied from Ganache into the designated field.
    - Click "Import".

Step 6: Test Transactions with MetaMask

1. **Initiate a Transaction**:
    - With MetaMask connected to your Ganache network, select one of the imported accounts.
    - Click **"Send"** in MetaMask, and enter the recipient address (you can use another Ganache account's address) and the amount of ETH to send.
    - Confirm the transaction details and click **"Next"**, then **"Confirm"**.
2. **Check Transaction Status in Ganache**:
    - Open the Ganache application and go to the **"Transactions"** tab to view the transaction details.
    - You should see the transaction listed, including details such as the sender, receiver, gas used, and block number.

Step 7: Verify the Transaction

1. **Check MetaMask**:
    - In MetaMask, navigate to the "Activity" tab for the account to see the transaction history.
2. **Verify the Balance Change**:
    - Confirm that the ETH balance has been updated according to the transaction.

By following the above steps, we have successfully created a local Ethereum blockchain with Ganache and tested transactions using MetaMask.

0x8e5B41DE8ff3C222ca925132C3C778aFaD2EB05D

Talha@123