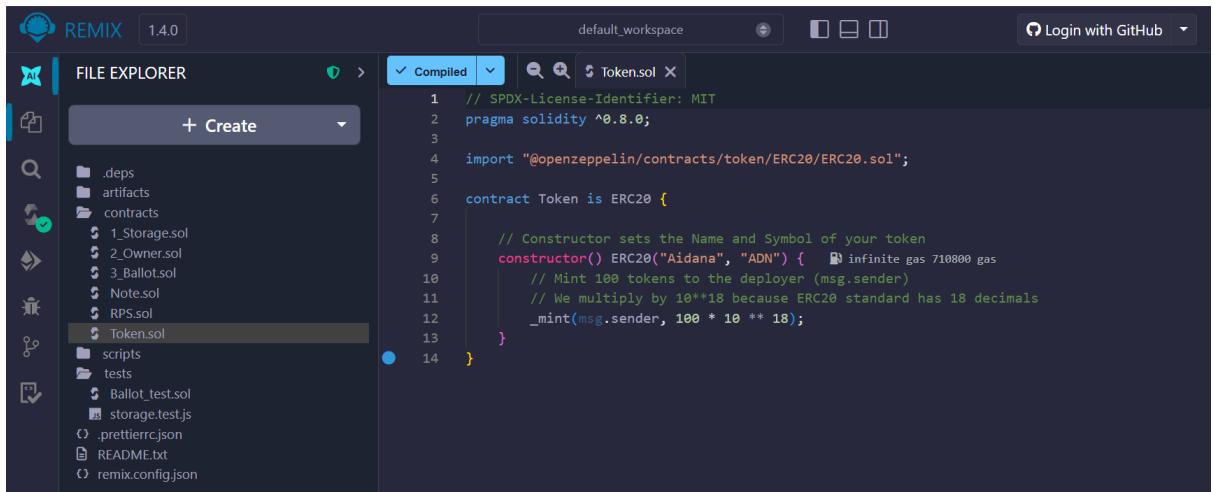


Assignment 11

Part 1: Create Your Own Cryptocurrency (ERC20)

- I created a "Token.sol" file in Remix IDE, pasted the provided code, compiled it, then deployed it and confirmed the transaction in MetaMask.
The contract was deployed successfully.

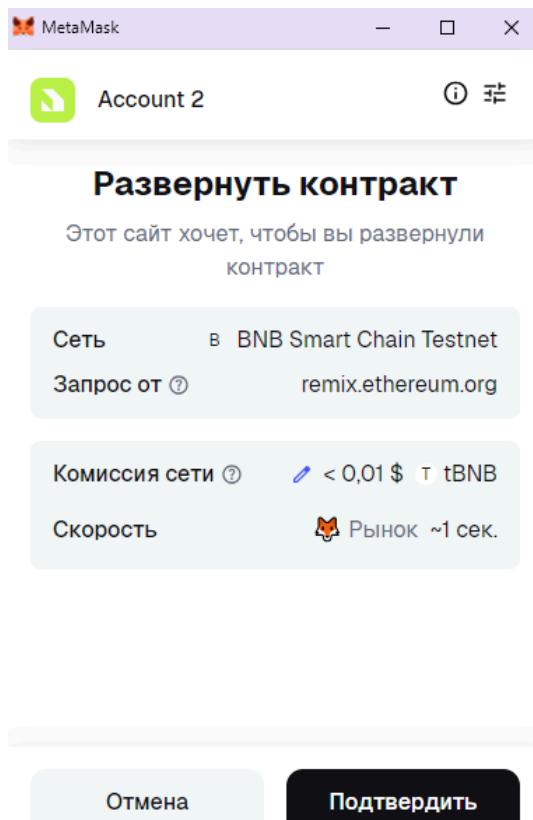


The screenshot shows the Remix IDE interface. On the left is the File Explorer sidebar with various project files listed. The main workspace shows the "Token.sol" file being edited. The code is an ERC20 token implementation:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

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

contract Token is ERC20 {
    // Constructor sets the Name and Symbol of your token
    constructor() ERC20("Aidana", "ADN") {
        _mint(msg.sender, 100 * 10 ** 18);
    }
}
```



Dec 11, 2025



Развертывание контракта
Подтверждено

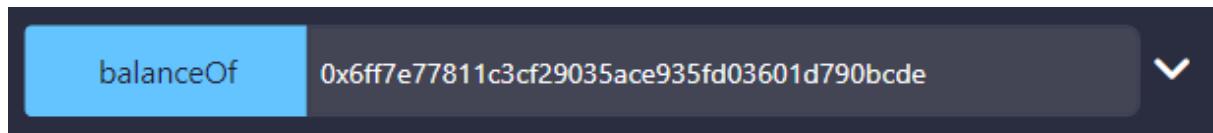
-0 TBNB
-0 TBNB

<https://testnet.bscscan.com/tx/0x3cb912d9ce217cde2131f295c95ec8b31823e3065514334fe7b33e4ff39500ab>

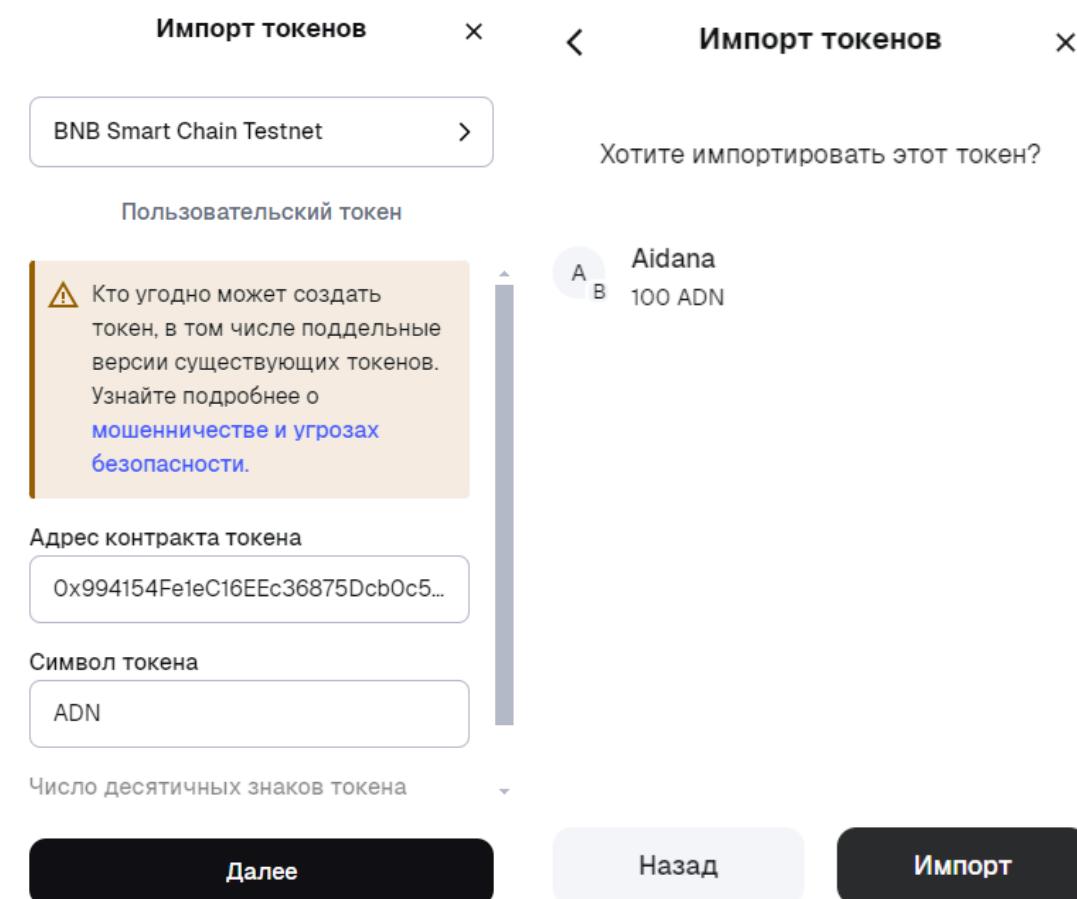
Contract Address: 0x994154Fe1eC16EEc36875Dcb0c5AF06b8BCCa031

2. To check that my wallet received 100 tokens, I used the **balanceOf** function.

MetaMask wallet address → 0x6ff7e77811c3cf29035ace935fd03601d790bcde



0: uint256: 10000000000000000000000000000000

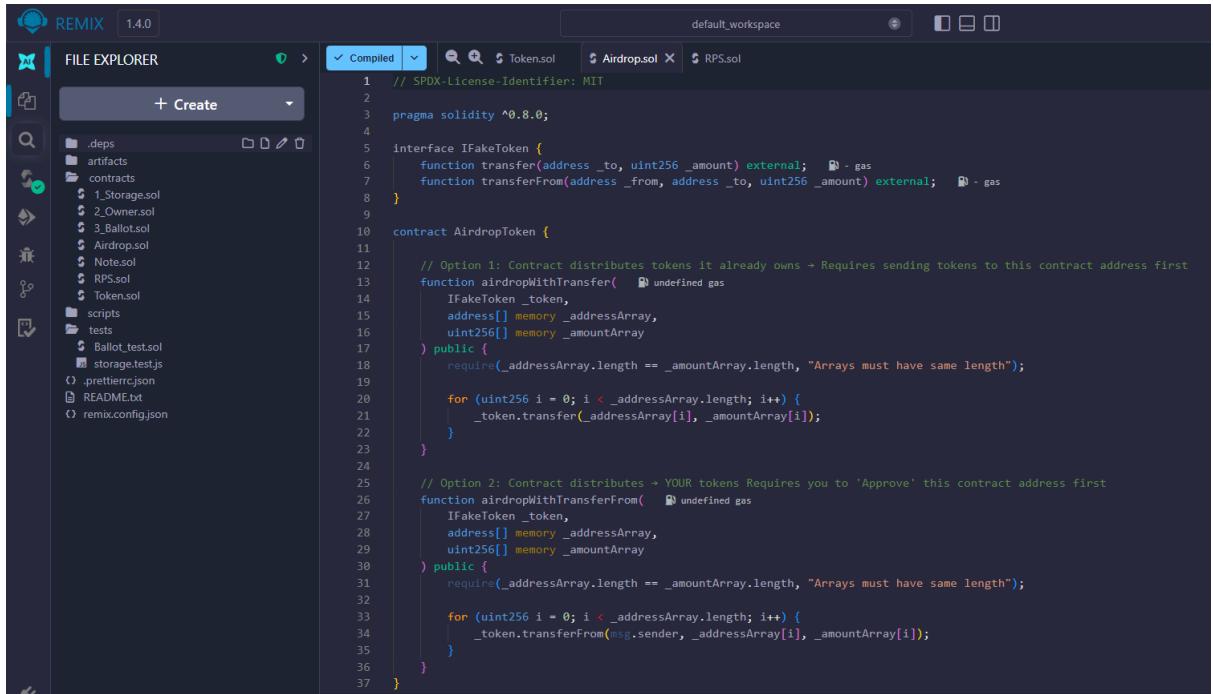


I also imported my token (ADN) into MetaMask to verify that my wallet had received 100 tokens.



Part 2: Token Distribution (Airdrop System)

- I created a "Airdrop.sol" file, pasted the provided code into it, compiled it, then deployed it and confirmed it in MetaMask.



The screenshot shows the REMIX IDE interface with the following details:

- FILE EXPLORER:** Shows files like .deps, artifacts, contracts (1.Storage.sol, 2_Owners.sol, 3_Ballot.sol, Airdrop.sol, Note.sol, RPS.sol, Token.sol), scripts, tests, and storage.test.js.
- CODE EDITOR:** Displays the `Airdrop.sol` file content:// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

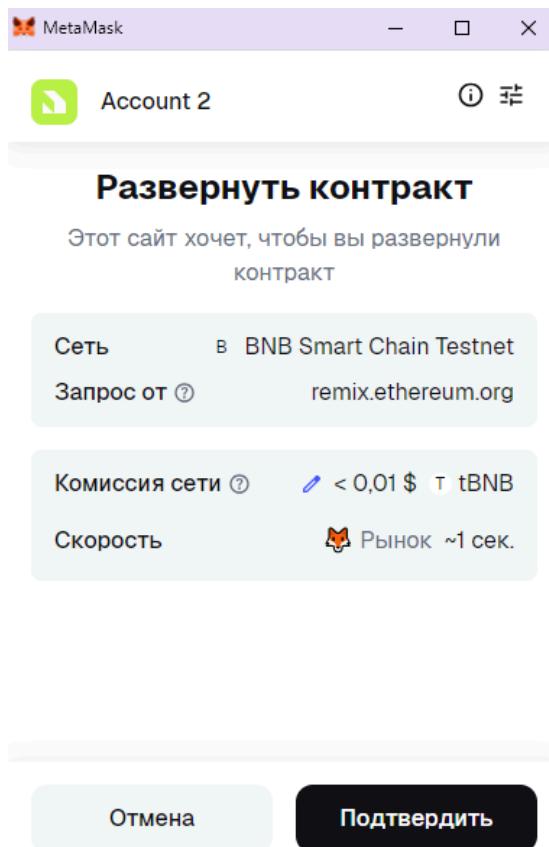
interface IFakeToken {
 function transfer(address _to, uint256 _amount) external; // - gas
 function transferFrom(address _from, address _to, uint256 _amount) external; // - gas
}

contract AirdropToken {
 // Option 1: Contract distributes tokens it already owns → Requires sending tokens to this contract address first
 function airdropWithTransfer(IFakeToken _token,
 address[] memory _addressArray,
 uint256[] memory _amountArray
) public {
 require(_addressArray.length == _amountArray.length, "Arrays must have same length");

 for (uint256 i = 0; i < _addressArray.length; i++) {
 _token.transfer(_addressArray[i], _amountArray[i]);
 }
 }

 // Option 2: Contract distributes → YOUR tokens Requires you to 'Approve' this contract address first
 function airdropWithTransferFrom(IFakeToken _token,
 address[] memory _addressArray,
 uint256[] memory _amountArray
) public {
 require(_addressArray.length == _amountArray.length, "Arrays must have same length");

 for (uint256 i = 0; i < _addressArray.length; i++) {
 _token.transferFrom(msg.sender, _addressArray[i], _amountArray[i]);
 }
 }
}



Dec 11, 2025



Развертывание контракта
Подтверждено

-0 TBNB
-0 TBNB

<https://testnet.bscscan.com/tx/0xb4b13eb29d57b041c69481a2e93a177a83e61976bcfb5ede778326ed1c614764>

Contract Address: 0xaAAFEF2b87e27deD1fD8Ad9B6037CcAfD26C6024

The screenshot shows a list of transactions. The first item is 'TOKEN AT 0X994...CA031 (BLOCKCHAIN)' with three icons (copy, details, delete). The second item is 'AIRDROPTOKEN AT 0XAAA...C6024' with three icons (copy, details, delete).

2. To approve the contract, I go back to my deployed "Token" contract from Part 1 and find the **approve** function.
 - **spender:** 0xaAAFEF2b87e27deD1fD8Ad9B6037CcAfD26C6024 ("Airdrop" contract address)
 - **value:** 10000000000000000000000000000000 (maximum allowance to spend my tokens)

Then I clicked "Transact" to submit the approval.

The screenshot shows the 'APPROVE' transaction page. It has two input fields: 'spender:' with the value '0xaAAFEF2b87e27deD1fD8Ad9B6037CcAfD26C6024' and 'value:' with the value '10000000000000000000000000000000'. Below the fields are buttons for 'Calldata' and 'Parameters', and an orange 'transact' button.



Account 2



Запрос на ограничение расходов

Этот сайт хочет получить разрешение на вывод
ваших токенов

Прогнозируемые изменения

Вы даете кому-то разрешение на трату этой
суммы с вашего счета.

Лимит расходования

1000 0x99415...Ca031

Расходующее лицо 0xaAAFE...C6024

Сеть BNB Smart Chain Testnet

Запрос от remix.ethereum.org

Комиссия сети < 0,01 \$ tBNB

Скорость Рынок ~1 сек.

Отмена

Подтвердить

Dec 11, 2025



Одобрить лимит расходов в размере ADN

Подтверждено

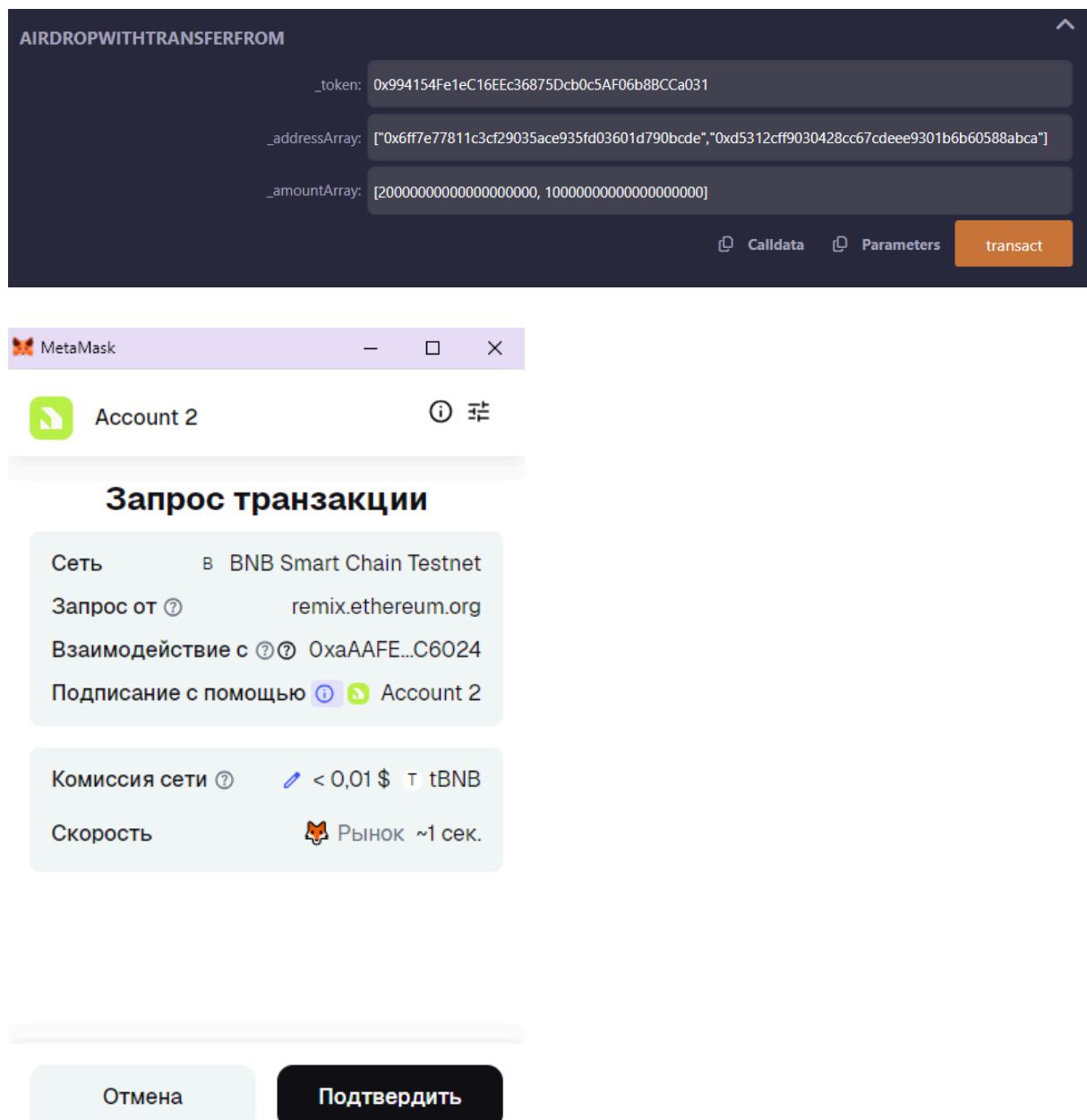
[block:77780829 txIndex:0] from: 0x6ff...0BcdE to: Token.approve(address,uint256) 0x994...Ca031 value: 0 wei data: 0x095...00000 logs: 1
hash: 0x4b0...f2dbf

<https://testnet.bscscan.com/tx/0xaa0e02f9bcb0858f9cba0a2c9687f505ccd595596389704ddf6f72236fe4f1f4>

3. To execute the "Airdrop" contract , I go to the "Airdrop" contract and use the **airdropWithTransferFrom** function.

- **_token:** 0x994154Fe1eC16EEc36875Dcb0c5AF06b8BCCa031 ("Token" Address)
- **_addressArray:**
["0x6ff7e77811c3cf29035ace935fd03601d790bcde", "0xd5312cff9030428cc67cdeee9301b6b60588abca"] → my MetaMask wallet addresses
- **_amountArray:** [200000000000000000000000, 1000000000000000000000] → (20 and 10 tokens)

Then I clicked "Transact".



Dec 11, 2025



Взаимодействие по контракту
Подтверждено

-0 TBNB
-0 TBNB

<https://testnet.bscscan.com/tx/0x91dba155f2c3a104fea414369132c3c45a3b7ee88fe9b4a862d20899f93e1a0e>

My token (ADN) balance went down from 100 ADN to 90 ADN (-10 token)

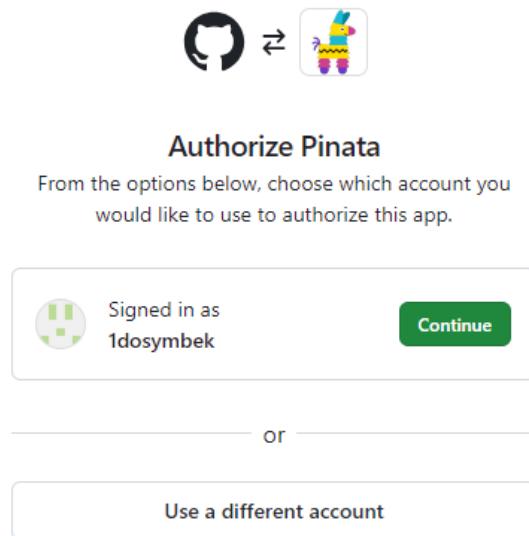
A B ADN

Нет доступного обменного курса
90 ADN

Part 3: NFT Collection with Metadata (ERC721 + IPFS)

Step A: Preparing IPFS Metadata (Crucial Step)

1. Pinata registration via GitHub.

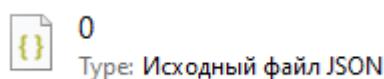


2. I clicked "Add file", picked a JPEG image, and uploaded it.

NAME	CID	SIZE	CREATION DATE	FILE ID	Actions
5321413009315924774.jpg	bafybeic3nscabf566bmhku7vllnnxrhxgdcs3pd6xfykvfddl2bc7xcqqy	373.68 KB	12/11/2025		

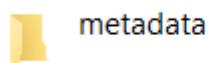
3. CID: **bafybeic3nscabf566bmhku7vllnnxrhxgdcs3pd6xfykvfddl2bc7xcqqy**

4. I created a file called "0.json" and pasted the code into it and added the image CID I copied from Pinata.



```
assignment 11 > metadata > 0.json > ...
1  {
2    "name": "Aidana NFT #0",
3    "description": "My first practical NFT work",
4    "image": "ipfs://bafybeic3nscabf566bmhku7v1lnnxrhxgdcs3pd6xfykvfddl2bc7xcqqy"
5 }
```

5. I put "0.json" into a folder named "metadata".



Name	Date modified	Type	Size
0	12/11/2025 5:49 AM	Исходный файл J...	1 KB

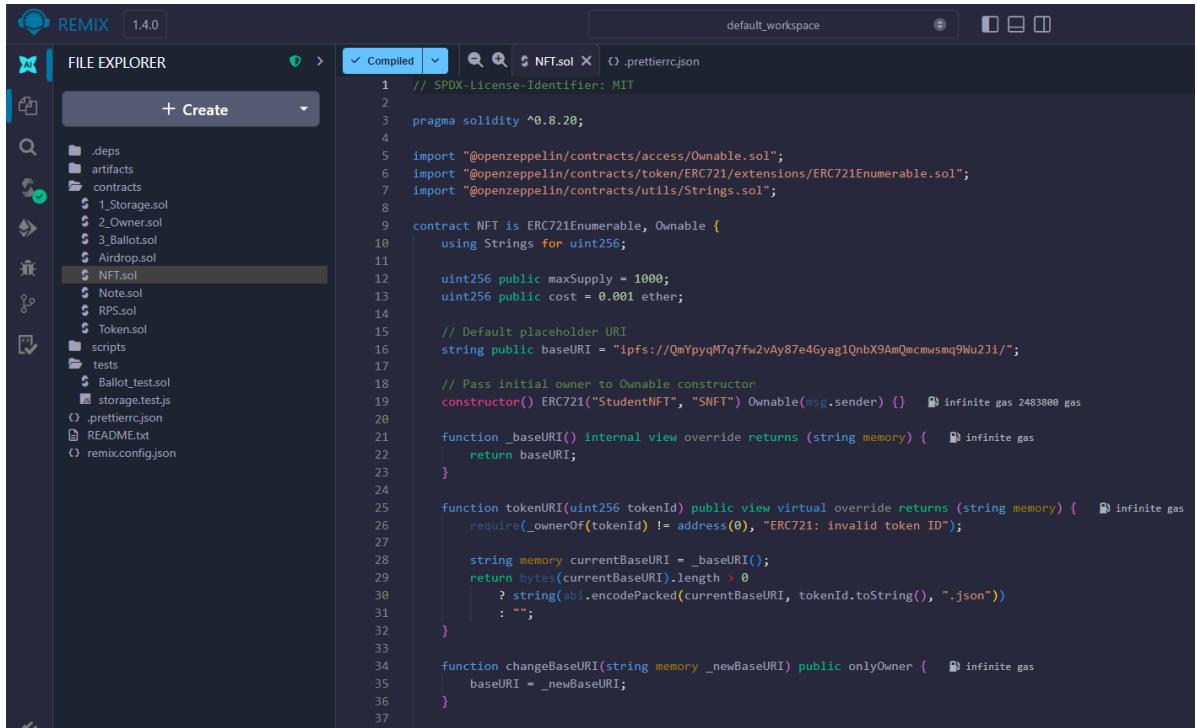
6. Next, I uploaded "metadata" to Pinata and copied its CID.

NAME	CID	SIZE	CREATION DATE	FILE ID	⋮
metadata	bafyb...fsura	223 B	12/11/2025	Copy	⋮
5321413009315924774.jpg	bafyb...xcqqy	373.68 KB	12/11/2025	Copy	⋮

CID: **bafybeigffjj3t5fpq6unnb55f4f3n7zmslpfocmob3jeesebbqyanfsura**

Step B: Smart Contract Deployment

- I created "NFT.sol", pasted the provided code into it, compiled and deployed it, and also confirmed it in MetaMask.



```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.20;

import "@openzeppelin/contracts/access/Ownable.sol";
import "@openzeppelin/contracts/token/ERC721/extensions/ERC721Enumerable.sol";
import "@openzeppelin/contracts/utils/Strings.sol";

contract NFT is ERC721Enumerable, Ownable {
    using Strings for uint256;

    uint256 public maxSupply = 1000;
    uint256 public cost = 0.001 ether;

    // Default placeholder URI
    string public baseURI = "ipfs://QmYpyqM7q7fw2vAy87e4Gyag1QnbX9AmQmcwsmq9Wu2Ji/";

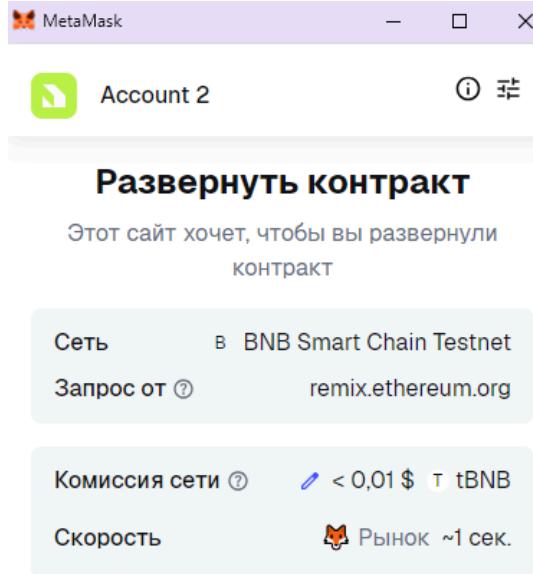
    // Pass initial owner to Ownable constructor
    constructor() ERC721("StudentNFT", "SNFT") Ownable(msg.sender) {}

    function _baseURI() internal view override returns (string memory) {
        return baseURI;
    }

    function tokenURI(uint256 tokenId) public view virtual override returns (string memory) {
        require(_ownerOf(tokenId) != address(0), "ERC721: invalid token ID");

        string memory currentBaseURI = _baseURI();
        return bytes(currentBaseURI).length > 0
            ? string(abi.encodePacked(currentBaseURI, tokenId.toString(), ".json"))
            : "";
    }

    function changeBaseURI(string memory _newBaseURI) public onlyOwner {
        baseURI = _newBaseURI;
    }
}
```



Отмена

Подтвердить

Dec 11, 2025



Развертывание контракта

Подтверждено

-0 tBnB

-0 tBnB

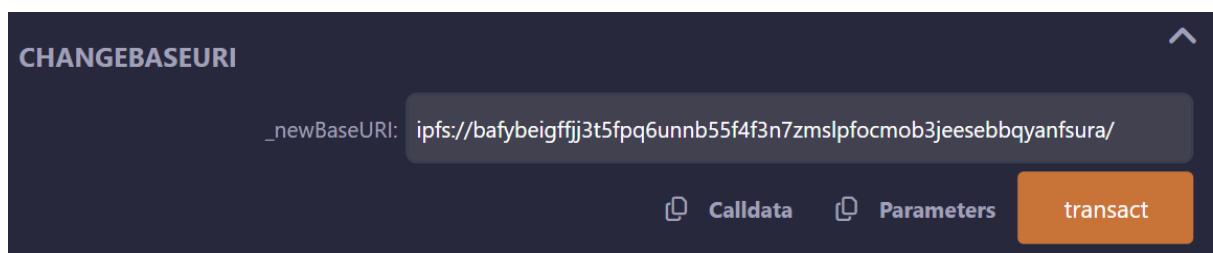
<https://testnet.bscscan.com/tx/0x6e6569aa7bbe236f4ff135714e76d802cc03f9b466c6a8f87b025ca624669cd4>

NFT Contact Address: 0x20ff997ce78ab748588EFbC51255C9525496D0F7

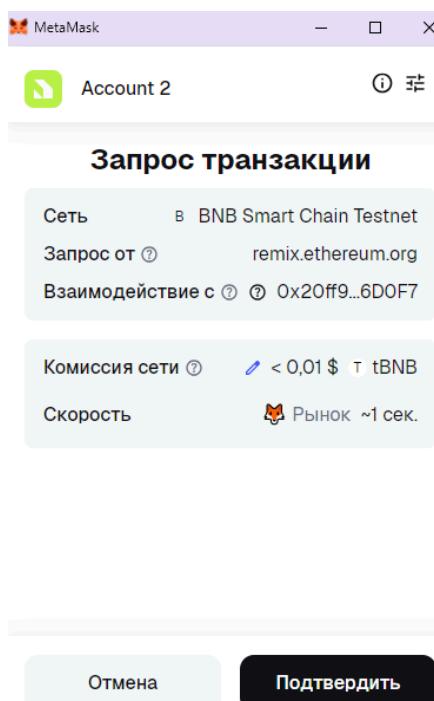
Step C: Linking and Minting

1. changeBaseURI function →

<ipfs://bafybeigffjj3t5fpq6unnb55f4f3n7zmslpfocmob3jeesebbqyanfsura/>



Then, I clicked "Transact" and confirmed it in MetaMask.



Dec 11, 2025



Взаимодействие по контракту

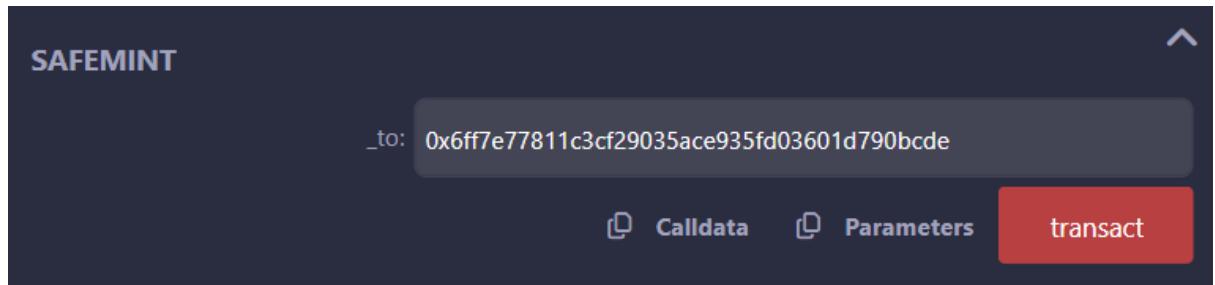
Подтверждено

-0 tBnB

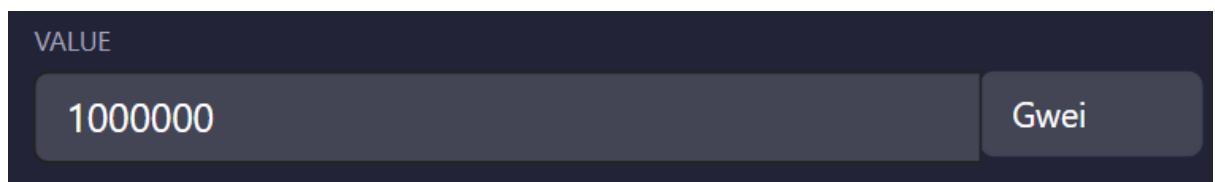
-0 tBnB

<https://testnet.bscscan.com/tx/0xe688b79c859409517c5896aa5db7526d9af66fa0c7bdbefd831cd1f8ea946cc1>

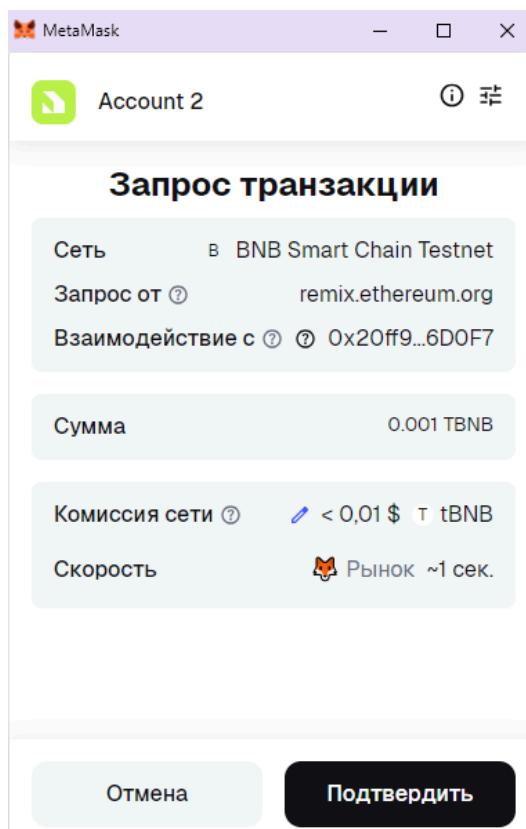
2. **safeMint** function → 0x6ff7e77811c3cf29035ace935fd03601d790bcde
(MetaMask Account Address)



the "value" field → 1000000 Gwei = 0.001 Ether → "Transact".



I confirmed the transaction request in MetaMask.



Dec 11, 2025



Взаимодействие по контракту
Подтверждено

-0.001 TBNB
-0.001 TBNB

<https://testnet.bscscan.com/tx/0x97b0991775502db8b2acf579f30be44024b169c5a46769c186fe863066a81342>

3. **tokenURI** function with "ID 0" returned:

The screenshot shows a dark-themed interface for interacting with a smart contract. At the top, it says "TOKENURI". Below that, there is a text input field containing "tokenId: '0'". To the right of the input field are three buttons: "Calldata", "Parameters", and a large blue "call" button. The "call" button is highlighted with a light blue background.

string:

ipfs://bafybeigffjj3t5fpq6unnb55f4f3n7zmslpfocmob3jeesebbqyanfsura/0.json

This screenshot is similar to the one above, showing the "TOKENURI" interface. It has the same layout with "tokenId: '0'" in the input field and a "call" button. Below the interface, the returned value is displayed: "0: string: ipfs://bafybeigffjj3t5fpq6unnb55f4f3n7zmslpfocmob3jeesebbqyanfsura/0.json".

<https://gateway.pinata.cloud/ipfs/bafybeigffjj3t5fpq6unnb55f4f3n7zmslpfocmob3jeesebbqyanfsura/0.json>

NFT displaying in MetaMask:

Account 2 ▾

5

0,00 \$
+0,00 \$ (+0,00 %)

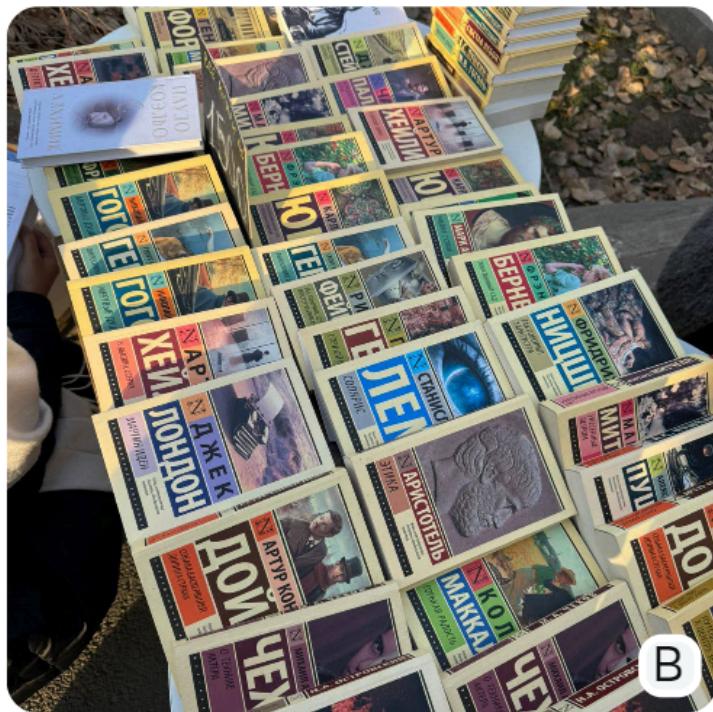
\$ Купить ↑↓ Обмен... ▷ Отправ... ↵ Получи...

Токены DeFi NFT Деятельность

BNB Smart Chain Testnet ▾



Aidana NFT #0



Aidana NFT #0



B

Aidana NFT #0

My first practical NFT work

Адрес контракта

0x20ff9...6D0F7 [🔗](#)

ID токена

0

Стандарт токена

ERC721

Отказ от ответственности: MetaMask извлекает медиафайл из исходного URL. Этот URL иногда изменяется торговой площадкой, на которой был выполнен майнтинг NFT.