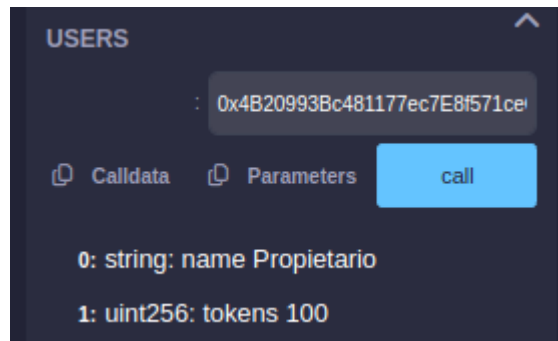


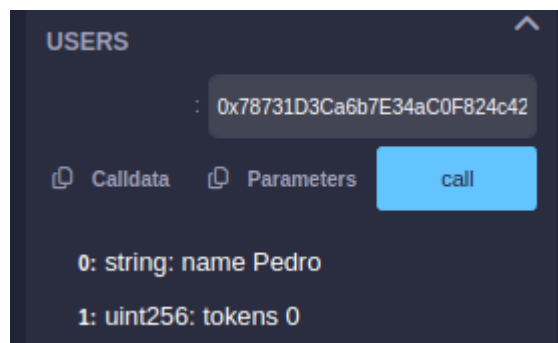
EJERCICIO 1

Creamos una cuenta propietario:



The screenshot shows a dark-themed interface titled 'USERS'. At the top, there is a text input field containing the address '0x4B20993Bc481177ec7E8f571ce'. Below this, there are three buttons: 'Calldata', 'Parameters', and a blue 'call' button. At the bottom, the parameters for the transaction are listed: '0: string: name Propietario' and '1: uint256: tokens 100'.

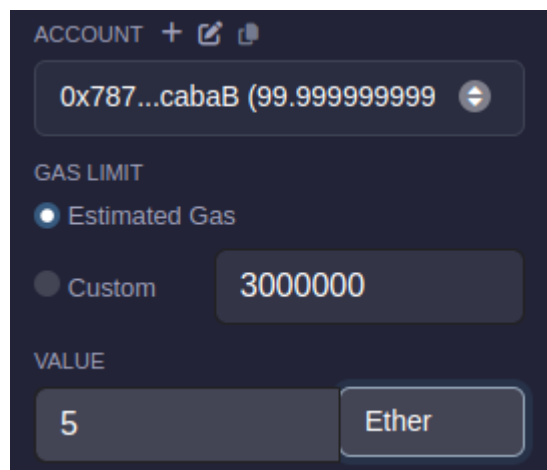
y un usuario Pedro



The screenshot shows a dark-themed interface titled 'USERS'. At the top, there is a text input field containing the address '0x78731D3Ca6b7E34aC0F824c42'. Below this, there are three buttons: 'Calldata', 'Parameters', and a blue 'call' button. At the bottom, the parameters for the transaction are listed: '0: string: name Pedro' and '1: uint256: tokens 0'.

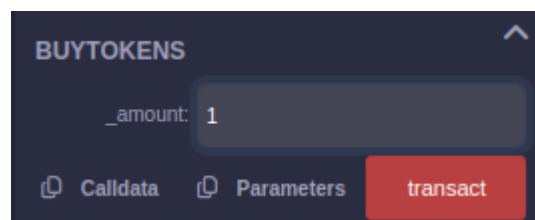
Comprar tokens

seleccionamos la cuenta de pedro y ponemos 1 token = 5 ether



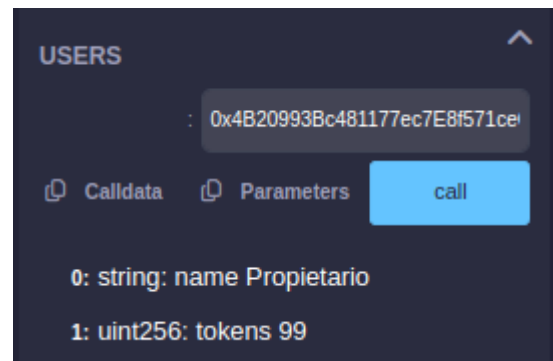
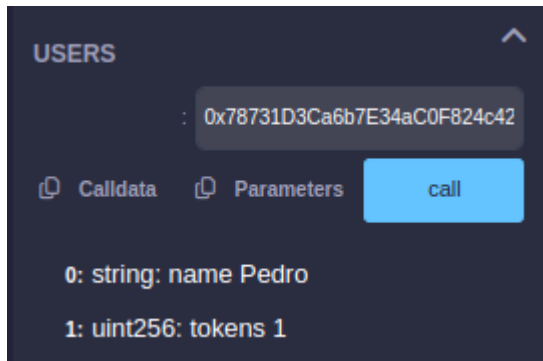
The screenshot shows a dark-themed interface titled 'ACCOUNT'. At the top, there is a dropdown menu showing '0x787...cabaB (99.999999999)'. Below this, there are two radio buttons: 'Estimated Gas' (selected) and 'Custom'. Next to the 'Custom' radio button is a text input field containing '3000000'. At the bottom, there is a 'VALUE' section with a text input field containing '5' and a button labeled 'Ether'.

Comparamos 1 token

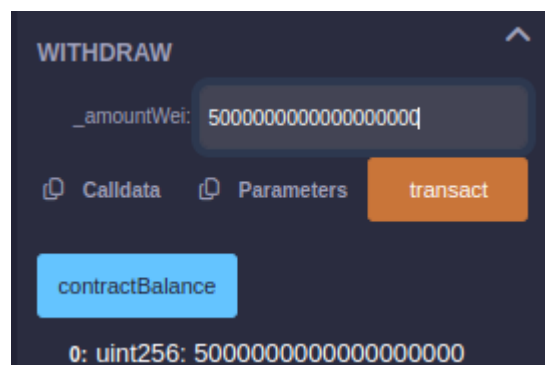


The screenshot shows a dark-themed interface titled 'BUYTOKENS'. At the top, there is a text input field labeled '_amount:' containing the value '1'. Below this, there are three buttons: 'Calldata', 'Parameters', and a red 'transact' button.

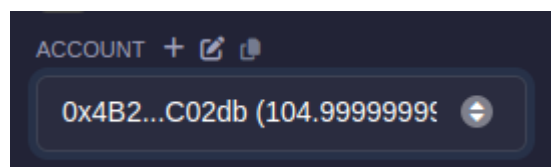
y comprobamos que Pedro tiene el token que acaba de comprar y el propietario perdió uno.



Pulsamos en contractBalance para ver el ether (5) y se lo mandamos al propietario



Comprobamos que al propietario le llego el ether



EJERCICIO 2

Qué es MetaMask

MetaMask es una **cartera digital (wallet)** que te permite:

Guardar y enviar criptomonedas (como Ether).

Conectarte a aplicaciones Web3 (por ejemplo, Remix o dApps).

Usar redes de prueba (testnets) para practicar sin dinero real.

1 Descargamos la extensión

2 Creamos la wallet

Contraseña de MetaMask

Utiliza esto para recuperar la billetera en todos los dispositivos. **MetaMask can't reset it.**

Crear nueva contraseña

La contraseña debe tener al menos 8 caracteres

Confirmar contraseña

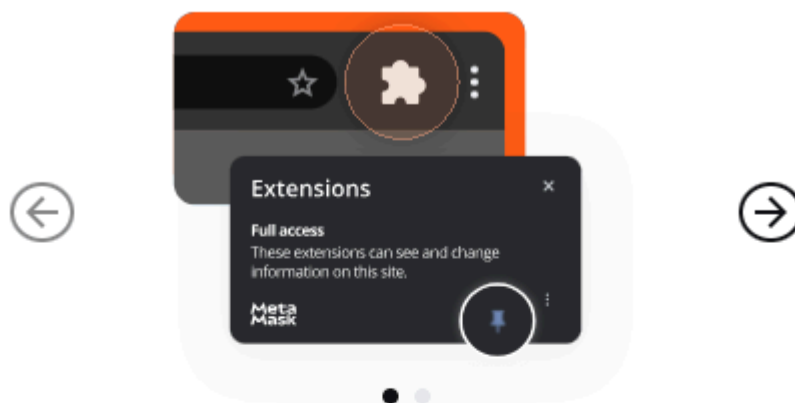
☐ Get product updates, tips, and news including by email. We may use your interactions to improve what we share.

Crear contraseña

Escanea el código QR y descarga la aplicación



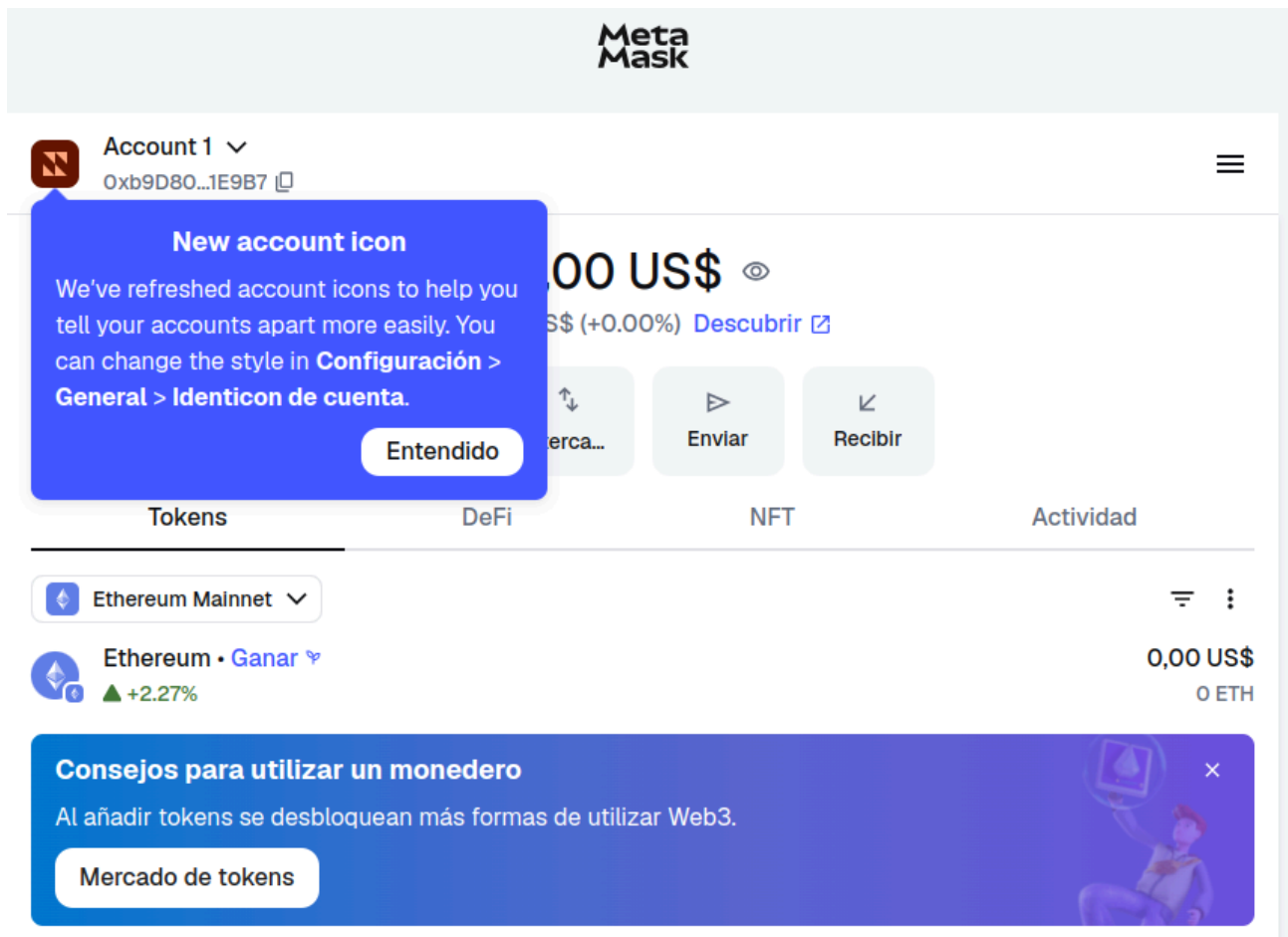
Lleva MetaMask contigo a todas partes con la



¡Instalación completa!

Ancla MetaMask en tu navegador para que sea accesible y las confirmaciones de las transacciones se vean con facilidad.

Hecho



Ya tenemos **MetaMask** instalado.

3 Activar la red de prueba Sepolia

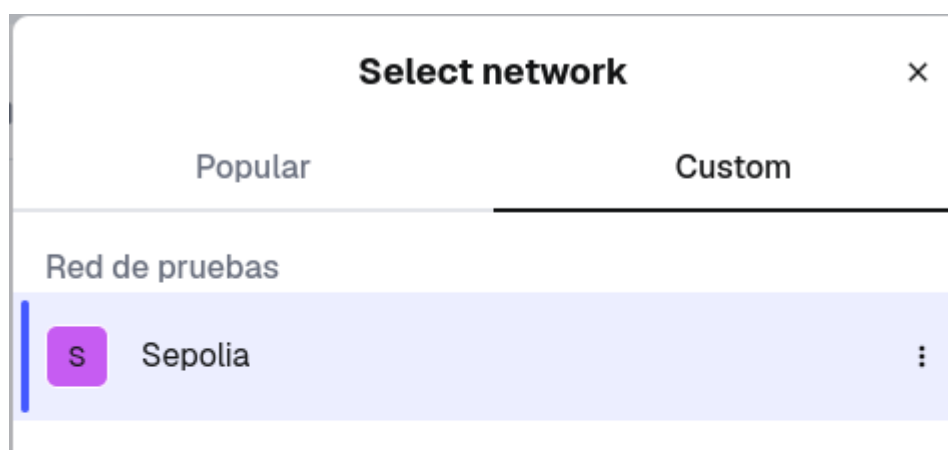
Configuración → avanzado → activamos mostrar redes de prueba:

Mostrar redes de prueba

Seleccione esta opción para mostrar las redes de prueba en la lista de redes

☒ ACTIVADO

En la **ventana de selección de red** de MetaMask seleccionamos la red de pruebas **sepolia**



Aquí es donde elegimos en qué blockchain o testnet vamos a trabajar.

4 Obtener ETH de prueba (faucet)

Los **faucets** son páginas que envían pequeñas cantidades de ETH falso (de prueba) gratis.

cloud.google.com/application/web3/faucet/ethereum/sepolia

Cloud Web3

Get your first drip on Celo's new Sepolia testnet today! Open faucet →

Ethereum Sepolia Faucet BETA

Get free Sepolia ETH sent directly to your wallet. Brought to you by [Google Cloud for Web3](#)

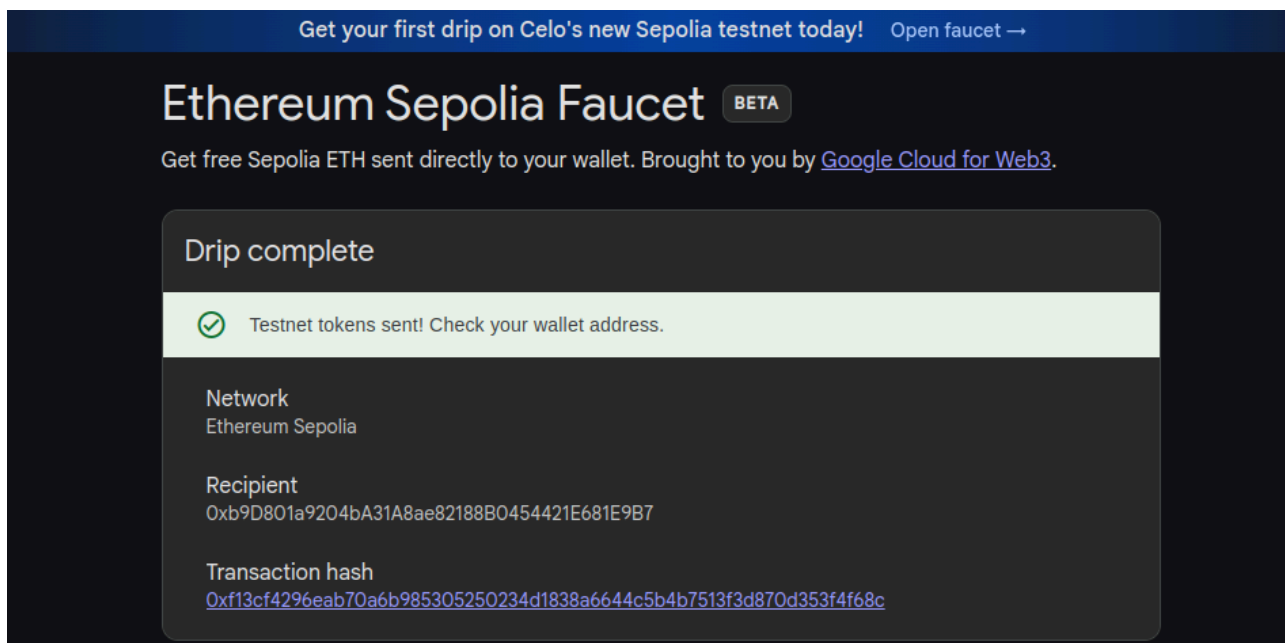
Select network*
Ethereum Sepolia

*required

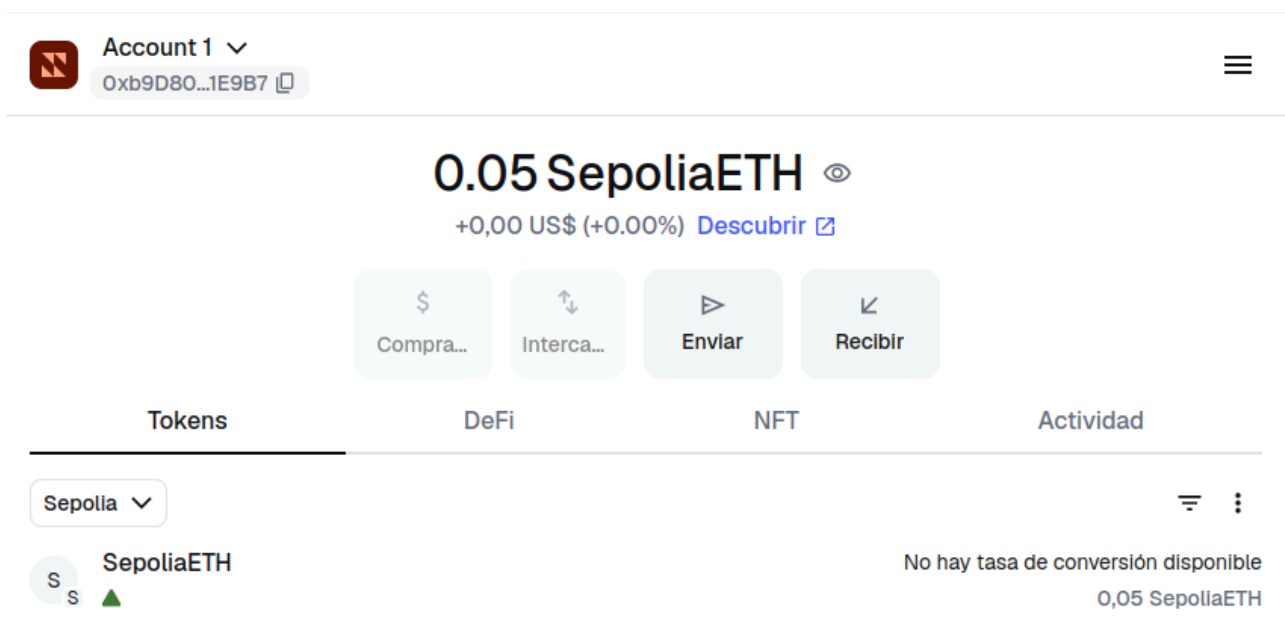
Wallet address or ENS name*
0xb9D801a9204bA31A8ae82188B0454421E681E9B7

Enter the account address or ENS name where you want to receive tokens

Receive 0.05 Sepolia ETH



Y vemos como ha incrementado a 0.05



Le damos a la opción de ver los detalles de la transacción:

Overview ETH BALANCE 0.05 ETH	More Info TRANSACTIONS SENT Latest: N/A First: N/A FUNDED BY 0x52f1984C...2E7aCEedD 5 mins ago	Multichain Info N/A
--	---	-------------------------------

Transactions
Token Transfers (ERC-20)

Latest 1 from a total of 1 transactions
 Download Page Data

Transaction Hash	Method	Block	Age	From	To	Amount	Txn Fee
0xf13cf4296ea...	Transfer	9445775	5 mins ago	0x52f1984C...2E7aCEedD	0xb9D801a9...1E681E9B7	0.05 ETH	0.00000002

MetaMask

<

Revisar

≡

S

0,01 SepoliaETH

De

Account 1

>

Para

Account 2

Red

Sepolia

Tarifa de red

0 S SepoliaETH

Velocidad


Mercado ~12 s

Cancelar

Confirmar


Vemos la confirmación después de unos segundos de que la operación se ha realizado





Account 1 
0xb9D80...1E9B7 





0.04 SepoliaETH 

+0,00 US\$ (+0.00%) [Descubrir](#) 



Compra...


Interca...




Enviar


Recibir

TokensDeFiNFTActividad

Sepolia 

Oct 19, 2025

Enviado
Confirmado


-0.01 SepoliaETH
-0.01 SepoliaETH

Overview

ETH BALANCE
0.039968499999748 ETH

More Info


TRANSACTIONS SENT
Latest: 2 mins ago ↗ First: 2 mins ago ↗

FUNDED BY
0x52f1984C...2E7aCEedD  | 30 mins ago













Multichain Info

N/A

TransactionsToken Transfers (ERC-20)

Latest 2 from a total of 2 transactions 

 Download Page Data 

	Transaction Hash	Method 	Block	Age	From	To	Amount	Txn Fee
	0x75f3ffd159e... 	Transfer	9445908	1 min ago	0xb9D801a9...1E681E9B7 	 0xA28aF3f4...8660e1bE7 	0.01 ETH	0.0000315
	0xf13cf4296ea... 	Transfer	9445775	29 mins ago	0x52f1984C...2E7aCEedD 	 0xb9D801a9...1E681E9B7 	0.05 ETH	0.00000002

EJERCICIO 3

Introducimos en REMIX los valores del constructor que hay en el contrato
constructor(string memory _titular, uint256 _saldoInicial)

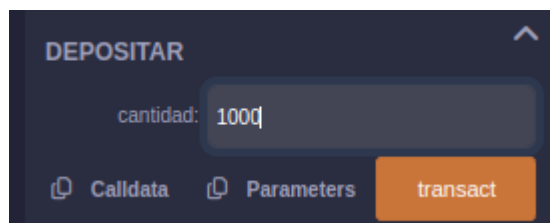


Deployamos y consultamos el titular y el saldo del mismo:



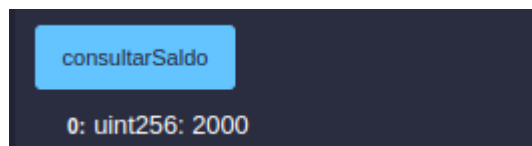
Depositar dinero

Añadimos otros 1000 a la cuenta.



The screenshot shows a dark-themed interface titled "DEPOSITAR". It features a text input field labeled "cantidad:" with the value "1000" entered. Below the input field are three buttons: "Calldata", "Parameters", and "transact". The "transact" button is highlighted in orange.

Y consultamos el saldo:



The screenshot shows a dark-themed interface with a single button labeled "consultarSaldo". Below the button, the text "0: uint256: 2000" is displayed.

Retirar dinero (solo el propietario)

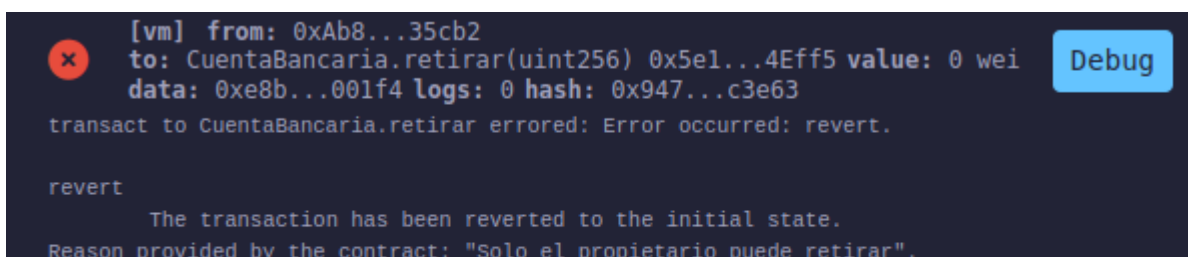
Retiramos 500 y consultamos el saldo



The screenshot shows a dark-themed interface titled "RETIRAR". It features a text input field labeled "cantidad:" with the value "500" entered. Below the input field are three buttons: "Calldata", "Parameters", and "transact". The "transact" button is highlighted in orange. Below these buttons is a button labeled "consultarSaldo". At the bottom, the text "0: uint256: 1500" is displayed.

Prueba desde otra cuenta

Cambiamos a otra cuenta intentamos retirar dinero y nos indica que no es el propietario de la cuenta y no puede retirar dinero



The screenshot shows a console log with a red "x" icon. The log contains the following text:

```
[vm] from: 0xAb8...35cb2  
to: CuentaBancaria.retirar(uint256) 0x5e1...4Eff5 value: 0 wei  
data: 0xe8b...001f4 logs: 0 hash: 0x947...c3e63  
transact to CuentaBancaria.retirar errored: Error occurred: revert.  
  
revert  
The transaction has been reverted to the initial state.  
Reason provided by the contract: "Solo el propietario puede retirar".
```

A "Debug" button is visible on the right side of the log.

EJERCICIO 4

Elegimos el contrato ERC20

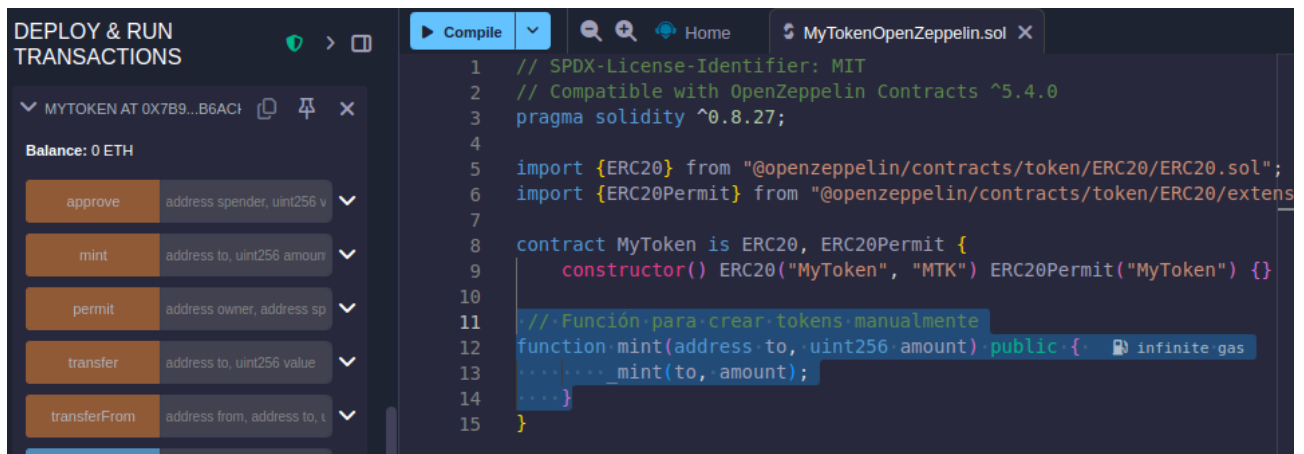
The screenshot displays the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' sidebar is open, showing a list of deployed contracts. The 'MYTOKEN AT 0X9D8...A5692' contract is selected, with a balance of 0 ETH. Below this, various functions are listed, including 'approve', 'permit', 'transfer', 'transferFrom', 'allowance', 'balanceOf', 'decimals', 'DOMAIN_SE...', 'eip712Domain', 'name', 'nonces', 'symbol', and 'totalSupply'. The main editor area shows the Solidity code for the 'MyTokenOpenZeppelin.sol' contract. The code is as follows:

```
1 // SPDX-License-Identifier: MIT
2 // Compatible with OpenZeppelin Contracts ^5.4.0
3 pragma solidity ^0.8.27;
4
5 import {ERC20} from "@openzeppelin/contracts/token/ERC20/ERC20.sol";
6 import {ERC20Permit} from "@openzeppelin/contracts/token/ERC20/extensions/ERC20Permit.sol";
7
8 contract MyToken is ERC20, ERC20Permit {
9     constructor() ERC20("MyToken", "MTK") ERC20Permit("MyToken") {}
10 }
```

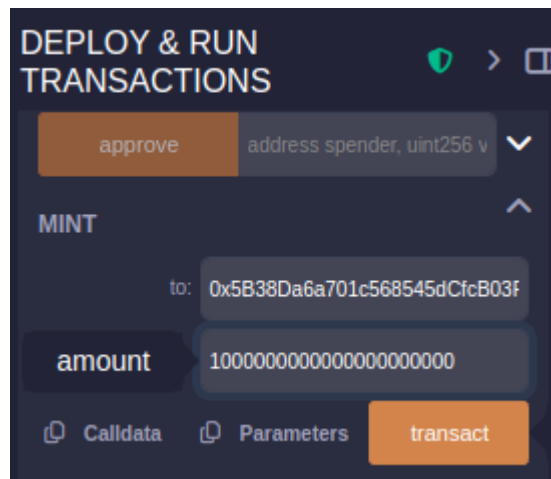
At the bottom, the 'Explain contract' panel is visible, showing a transaction log entry: '[vm] from: 0x5B3...eddC4 to: MyToken.(constructor) value: 0 wei data: 0x610...e0033 logs: 0 hash: 0x9b9...c5de7'. The 'AI copilot' button is also present.

Lo copiamos en REMIX lo compilamos y lo deployamos

El contrato original de OpenZeppelin **no crea tokens por sí solo**, no tiene función pública de mint(), por lo que la creamos



Acuñamos tokens (El número $10 \cdot 10^{20}$ significa “1000 tokens enteros”, porque los ERC-20 usan 18 decimales.)



y lo comprobamos mirando el balance que tiene esa dirección.

