

Desarrollo Blockchain Ethereum con Solidity

Módulo 4 - Implementar un token ERC20

Implementar un token ERC20

OpenZeppelin

- Empresa fundada en 2015 en Buenos Aires.
- Ofrecen productos y servicios orientados al desarrollo seguro de contratos inteligentes.
- Poseen una serie de contratos pre-diseñados que son confiables para su uso.
- Entre esos contratos poseen implementaciones para ERC20 y ERC721.



Instalar OpenZeppelin

1. Abrir una terminal y posicionarse en la carpeta de un proyecto de Truffle donde vamos a utilizar los contratos.
2. Ejecutar el comando: **npm install @openzeppelin/contracts**
Esto instala una serie de contratos, incluyendo el de los tokens.



Extender un contrato ERC20

1. En el proyecto de truffle crear un contrato nuevo.
2. Importar en el contrato el ERC20 de open zeppelin que está en la carpeta **node_modules/...**
3. Extender el contrato nuevo de ERC20.

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity >=0.7.0 <0.9.0;

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

contract ERC20_Nuevo is ERC20 {
}
```

Supply inicial

Para dar un suministro inicial de tokens, hay que modificar el constructor extendiéndolo del constructor original del contrato de OpenZeppelin.

En el constructor se debe llamar a la función **mint** que genera un balance de tokens para la cuenta que le indiquemos en el monto que le indiquemos.

```
contract ERC20_Nuevo is ERC20 {  
    constructor(string memory name, string memory symbol, string memory mes) ERC20(name, symbol) {  
        _mint(msg.sender, 1000);  
    }  
}
```

Implementar



Una vez finalizado se puede compilar e implementar el contrato y ya tenemos listo nuestro token para su utilización.

Opcionalmente podemos extender el código agregando atributos y funciones.

¡Muchas gracias!

¡Sigamos trabajando!