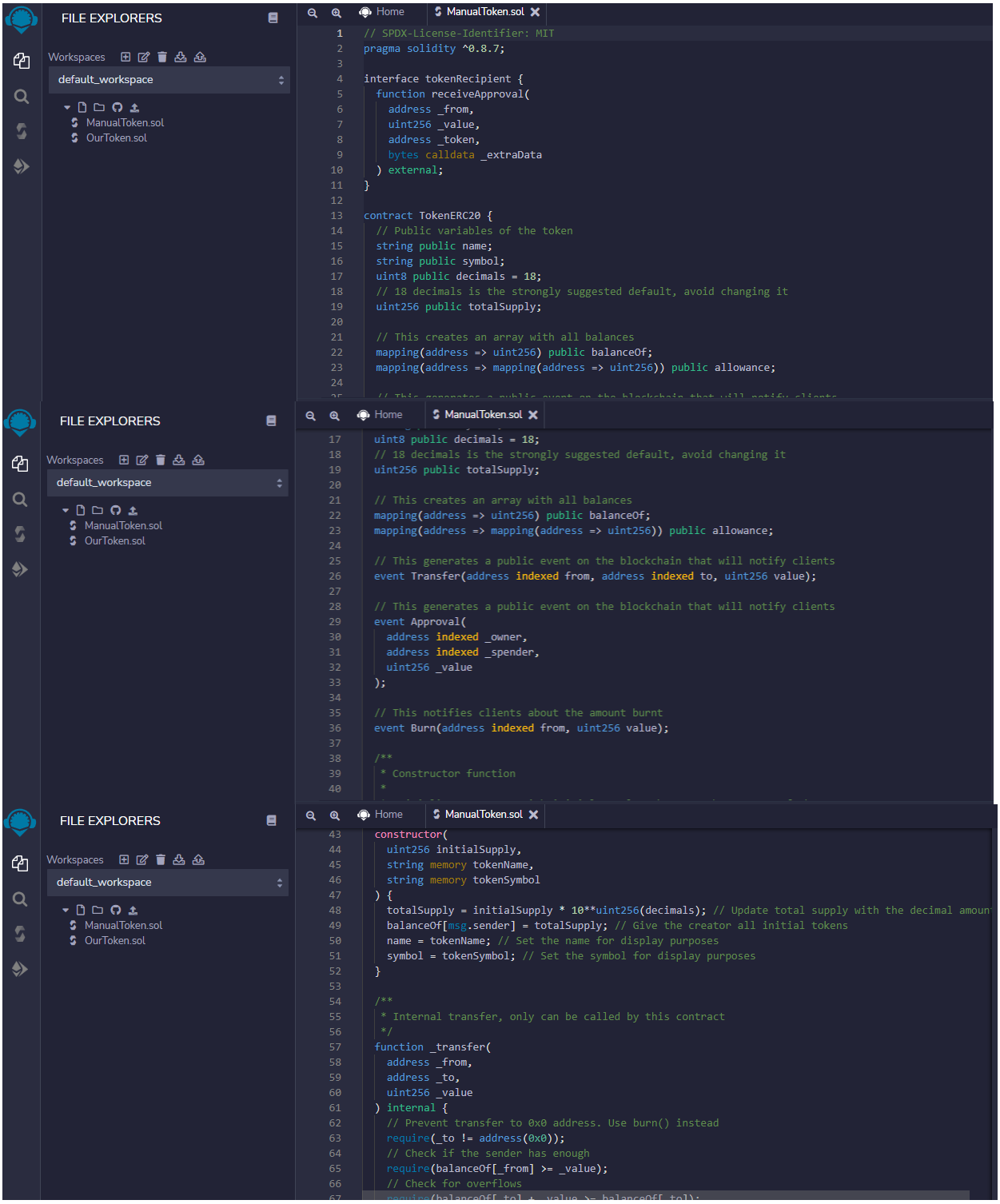
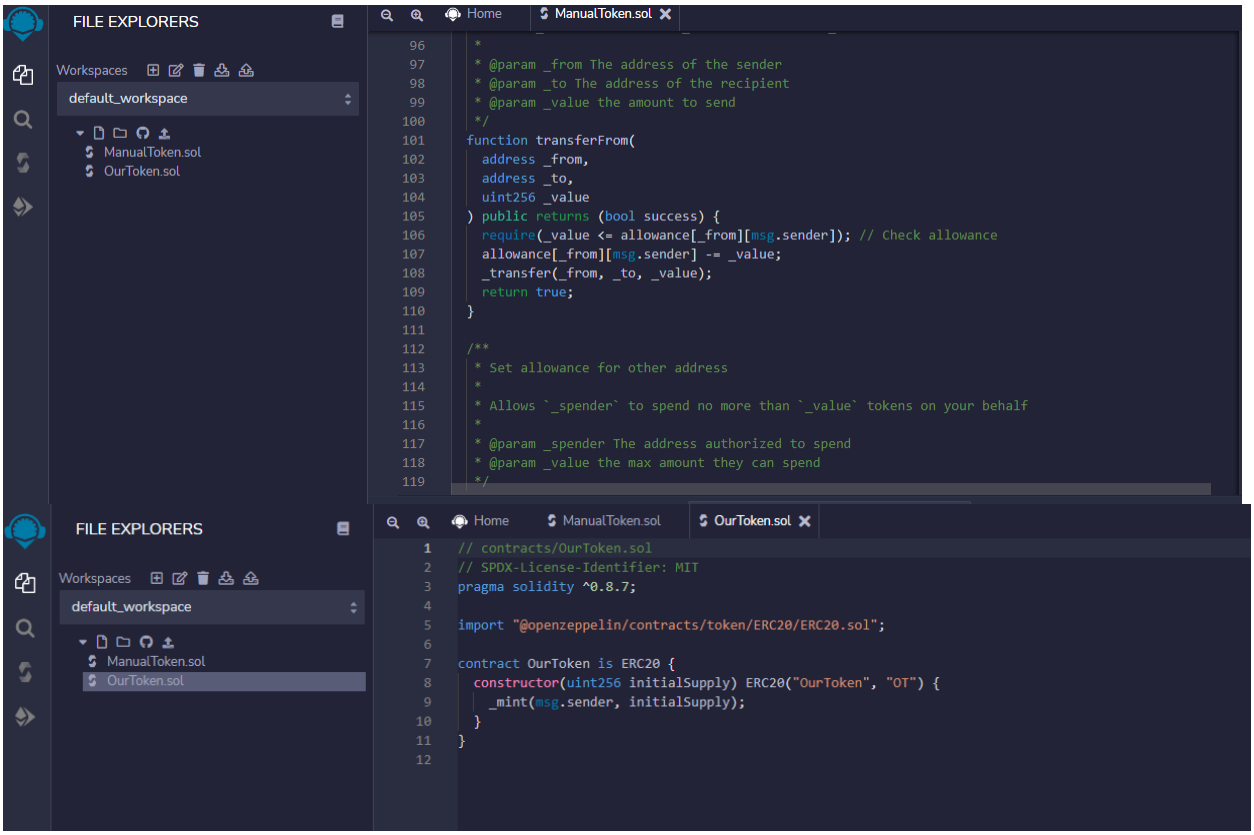
Hardhat ERC20s




FILE EXPLORERS
Workspaces
default_workspace
ManualToken.sol
OurToken.sol
Home
ManualToken.sol
96 *
97 * @param _from The address of the sender
98 * @param _to The address of the recipient
99 * @param _value the amount to send
100 */
101 function transferFrom(
102 address _from,
103 address _to,
104 uint256 _value
105) public returns (bool success) {
106 require(_value <= allowance[_from][msg.sender]); // Check allowance
107 allowance[_from][msg.sender] -= _value;
108 _transfer(_from, _to, _value);
109 return true;
110 }
111
112 /**
113 * Set allowance for other address
114 *
115 * Allows `_spender` to spend no more than `_value` tokens on your behalf
116 *
117 * @param _spender The address authorized to spend
118 * @param _value the max amount they can spend
119 */

FILE EXPLORERS
Workspaces
default_workspace
ManualToken.sol
OurToken.sol
Home
ManualToken.sol
OurToken.sol
1 // contracts/OurToken.sol
2 // SPDX-License-Identifier: MIT
3 pragma solidity ^0.8.7;
4
5 import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
6
7 contract OurToken is ERC20 {
8 constructor(uint256 initialSupply) ERC20("OurToken", "OT") {
9 _mint(msg.sender, initialSupply);
10 }
11 }
12