# 458.https://stackoverflow.com/questions/70897461/exclusive-nft-per-smart-contract-in-solidity

**T:**Exclusive NFT per smart contract in SOLIDITY

**Q:**I want to mint only one exclusive NFT per smart contract. All tutorials and books shows how to make collectables through inheritance of ERC721. So the simple code is:  
  
pragma solidity ^0.8.0;import "@openzeppelin/contracts/token/ERC721/extensions/ERC721URIStorage.sol";import "@openzeppelin/contracts/utils/Counters.sol";import "hardhat/console.sol"; contract SimpleCollectible is ERC721 { uint256 public tokenCounter; constructor () public ERC721 ("Dogie", "DOG"){ tokenCounter = 0; } function createCollectible(string memory tokenURI) public returns (uint256) { uint256 newItemId = tokenCounter; \_safeMint(msg.sender, newItemId); \_setTokenURI(newItemId, tokenURI); tokenCounter = tokenCounter + 1; return newItemId; } }  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
How restrict minting to 1 piece of art?

**C1:**You can create a storage variable as a boolean, and once someone mints it, set it to true, then you can check if said variable is true for the next minting attempts with a require.

1 **Answer**

**A1:**Add a condition to tokenCounter. You could use the following:  
  
function createCollectible(string memory tokenURI) public returns(uint256) { require(tokenCounter == 0, "error msg here"); uint256 newItemId = tokenCounter; \_safeMint(msg.sender, newItemId); \_setTokenURI(newItemId, tokenURI); tokenCounter = tokenCounter + 1; return newItemId; }  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
In fact, there are tons of ways of achieving the same result. Maybe you could use a simple boolean to save a little bit on gas.  
  
I hope you fin this useful :)