# 619.https://stackoverflow.com/questions/69193720/erc721-nft-creating-a-function-to-buy-sell-nfts-that-have-been-preminted-by-the

**T:**ERC721 NFT creating a function to buy/sell NFTs that have been preminted by the contract owner, security question

**Q:**I came across this answer while researching creating a function for someone to buy an NFT:  
  
https://stackoverflow.com/a/67384225/1414721  
  
The relevant lines are  
  
 IERC20 tokenContract = IERC20(tokenAddress); require(tokenContract.transferFrom(msg.sender, address(this), price), "buy: payment failed");  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
I don't want people to be able to buy my NFTs in anything other than Ethereum, although the author here says:  
  
prevent an edge case where if gas runs out during execution, the buyer could end up with their NFT for free  
  
This caught my eye and from reading the code it seems the check that is being done here is to prevent this edge case.  
  
What I'm not sure of is how the applies when the currency the NFT is being bought in is Ethereum.  
  
I have adjusted my buy function to look like  
  
 function buy(uint256 \_tokenId) external payable { uint256 price = tokenIdToPrice[\_tokenId]; require(price > 0, 'This token is not for sale'); require(msg.value == price, 'Incorrect value'); address seller = ownerOf(\_tokenId); IERC20 tokenContract = IERC20(address(0)); require(tokenContract.transferFrom(msg.sender, address(this), price), "buy: payment failed"); payable(seller).transfer(msg.value); \_transfer(seller, msg.sender, \_tokenId); tokenIdToPrice[\_tokenId] = 0; emit NftBought(seller, msg.sender, msg.value); }  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
which I believe a) incorporates the token Contract from being the Ethereum token (IERC20(address(0)) - I understand address(0) is the Ethereum token address?) and b  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
require(tokenContract.transferFrom(msg.sender, address(this), price), "buy: payment failed");  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
makes sure the gas limit edge case mentioned is handled.  
  
Is this correct, Googling this was quite hard.  
  
One last question (unrelated, I hope that's OK) - when a market displays all the NFTs available for a collection, i assume that the way they are doing that is the contract has a function that returns the NFT IDs and the Token URIs? Is that correct, or does OpenZeppelin provide this functionality and I don't have to concern myself with adding this function?

1 **Answer**

**A1:**i think the gas edge case was solved by changing the order of operations. first pay then transfer the token.