# 9.https://stackoverflow.com/questions/70913808/transfer-erc721-from-owner-to-buyer

**T:**Transfer ERC721 from owner to buyer

**Q:**I'm trying to let other addresses execute the buy function, but it throws me the error ERC721: approve caller is not the owner nor approved for allthis is the code  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
function testbuy() public payable{ require(ownerOf(1) == \_owner, "Already bought"); approve(msg.sender, 1); safeTransferFrom(\_owner, msg.sender, 1);}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
How could I make other address buy the NFT from the owner address? I'm having trouble understanding approve and setApproveForAll.. Thanks :)

2 **Answer**

**A1:**Based on the error message, I'm assuming that you're using the Openzeppelin implementation of ERC721.  
  
Since your contract derives from the OZ implementation, you can use their internal functions and don't need to go through the approve() and safeTransferFrom() process. If you wanted to go this way, you would need to invoke the approve() function from the \_owner address directly - not by the buying user through the testbuy() function, as this was the logical error in your code.  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
Specifically, you can use the \_transfer() function (source):  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
function testbuy() public payable{ require(ownerOf(1) == \_owner, "Already bought"); \_transfer(\_owner, msg.sender, 1);}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]

**A2:**You can find your error OpenZeppelin ERC721 here:  
  
function approve(address to, uint256 tokenId) public virtual override { address owner = ERC721.ownerOf(tokenId); require(to != owner, "ERC721: approval to current owner"); require( \_msgSender() == owner || isApprovedForAll(owner, \_msgSender()), "ERC721: approve caller is not owner nor approved for all" ); \_approve(to, tokenId);}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
Address which is calling testbuy() function has to be the owner of the token. In other case it cannot give the approval (or the token has to have approval for all), hence the approve function.