# 26.https://stackoverflow.com/questions/73480903/solidity-transferfrom-function-not-working-making-an-nft-marketplace-that-ac

**T:**Solidity TransferFrom() function not working / Making an nft marketplace that accepts custom tokens

**Q:**So the plan is to make a basic nft marketplace in which you can buy with eth and with a custom token, the eth part works but I haven´t been able to make the custom token part work.The token is already deployed to the mainnet, the nft marketplace will make use of it, so I forked the ethereum mainnet with hardhat so I could interact with it.This is the buy with eth function:  
  
function purchaseItemWithEth(uint \_itemId) external payable nonReentrant { uint \_totalPrice = getTotalPrice(\_itemId); Item storage item = items[\_itemId]; require(\_itemId > 0 && \_itemId <= itemCount, "item doesn't exist"); require(msg.value >= \_totalPrice, "not enough ether to cover item price and market fee"); require(!item.sold, "item already sold"); // pay seller and feeAccount item.seller.transfer(item.price); feeAccount.transfer(\_totalPrice - item.price); // update item to sold item.sold = true; // transfer nft to buyer item.nft.transferFrom(address(this), msg.sender, item.tokenId); // emit Bought event emit Bought( \_itemId, address(item.nft), item.tokenId, item.price, item.priceOsaku, item.seller, msg.sender ); }   
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
In theory, with the purchase with custom token function I would only need to change the token that is being sent, use the token instead of eth, so I made this:  
  
function purchaseItemWithToken(uint \_itemId) external payable nonReentrant { uint \_totalPrice = getTotalPriceWithToken(\_itemId; Item storage item = items[\_itemId]; require(\_itemId > 0 && \_itemId <= itemCount, "item doesn't exist"); //require(msg.value >= \_totalPrice, "not enough ether to cover item price and market fee"); require(!item.sold, "item already sold"); //pay seller and fee account depositTokens(10); sendTokens(item.seller, item.pricewithToken); sendTokens(feeAccount, (\_totalPrice - item.pricewithToken)); // update item to sold item.sold = true; // transfer nft to buyer item.nft.transferFrom(address(this), msg.sender, item.tokenId); // emit Bought event emit Bought( \_itemId, address(item.nft), item.tokenId, item.price, item.priceToken, item.seller, msg.sender ); } function getTotalPrice(uint \_itemId) view public returns(uint){ return((items[\_itemId].price\*(100 + feePercent))/100); } function getTotalPriceToken(uint \_itemId) view public returns(uint) { return((items[\_itemId].priceToken\*(100 + feePercent))/100); } function depositTokens(uint \_amount) public { ERC20(\_token).transferFrom(msg.sender, address(this), \_amount); } function sendTokens(address \_to, uint \_amount) public { ERC20(\_token).transfer(\_to, \_amount); } function getTokenSmartContractBalance() external view returns(uint) { return ERC20(\_token).balanceOf(address(this)); }  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
Buy it doesn't work.I tried approving the token before the transferFrom function is called like this:  
  
const name = await tokenContract.name() const approve = await tokenContract.connect(Buyer1Wallet).approve(publicKeyBuyer1, 1000000000) const approveMarketplace = await osakuContract.connect(Buyer1Wallet).approve('0xe8f7d98be6722d42f29b50500b0e318ef2be4fc8', 1000000000) console.log(approve) console.log(`${name} approved`)  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
But I get the error "not enough ether to cover item price and market fee" which is defined as one of the requires of the purchaseItem function.It mentions it even though before approving I bought 1 eth worth of tokens via uniswap with this:  
  
const apeBuyer1 = await uniswapContract.connect(Buyer1Wallet).swapExactETHForTokensSupportingFeeOnTransferTokens( 99, PAIR\_ADDRESS, publicKeyBuyer1, Date.now() + 1000 \* 60 \* 5, { value: ethers.utils.parseUnits(String(1), "ether"), gasLimit: 1000000, gasPrice: ethers.utils.parseUnits(String(50), "gwei"), } ) let receipt3 = await apeBuyer1.wait() console.log(`Buyer1 Bought: ${receipt3.transactionHash}`);  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
I don't know why it says it doesn't have enough balance when I a lot more than necessary for the function to run.  
  
All help is appreciated, thank you beforehand.

0 **Answer**