# 566.https://stackoverflow.com/questions/69956957/update-metadata-of-metaplex-nft

**T:**Update Metadata of Metaplex NFT

**Q:**I have some problem in updating Metaplex NFT Metadata.I used @metaplex/js and this is my code.  
  
import { programs } from '@metaplex/js';export const updateMetadataV1 = async () => { let { metadata : {Metadata, UpdateMetadata, MetadataDataData, Creator} } = programs; let signer = loadWalletKey(keyfile); let nftMintAccount = new PublicKey("EC8gGdtVFDoTf3vEGbLvPp7SVWta2xQrs99iWMbaFrdE"); let metadataAccount = await Metadata.getPDA(nftMintAccount); const metadat = await Metadata.load(solConnection, metadataAccount); let newUri = "https://arweave.net/my arweave address"; if (metadat.data.data.creators != null) { const creators = metadat.data.data.creators.map( (el) => new Creator({ ...el, }), ); let newMetadataData = new MetadataDataData({ name: metadat.data.data.name, symbol: metadat.data.data.symbol, uri: newUri, creators: [...creators], sellerFeeBasisPoints: metadat.data.data.sellerFeeBasisPoints, }) const updateTx = new UpdateMetadata( { feePayer: signer.publicKey }, { metadata: metadataAccount, updateAuthority: signer.publicKey, metadataData: newMetadataData, newUpdateAuthority: signer.publicKey, primarySaleHappened: metadat.data.primarySaleHappened, }, ); let result = await sendAndConfirmTransaction(solConnection, updateTx, [signer]); console.log("result =", result); }}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
The transaction result has no error, it means transaction success.I checked it on Solana Explorer.But the metadata doesn't change. What's the matter?

**C1:**Where can I find the solConnection definition?

**C2:**Where did you get the mint nft pubkey from?

**C3:**Are sure that the pubkey from the keypair file has the update authority of the nft metadata account? You can check it on explorer from the nft metadata account.

**C4:**Yeah, I am sure all about your suggestion, I found a solution from github.com/metaplex-foundation/metaplex/issues/…

**C5:**is it still an issue? Can you please show the transaction on slana explorer which is not showing an error?

2 **Answer**

**A1:**To Solve this issue this code can be used, This is a low-level code that uses the mpl-token-metadata npm package to call the on-chain updateMetadata Function.  
  
import { Wallet } from "@project-serum/anchor";import \* as anchor from "@project-serum/anchor";import {createUpdateMetadataAccountV2Instruction,DataV2,UpdateMetadataAccountV2InstructionArgs,UpdateMetadataAccountV2InstructionAccounts} from "@metaplex-foundation/mpl-token-metadata"const fs = require("fs");(async() => { // This is the Update Authority Secret Key const secretKey = fs.readFileSync( "/Users/pratiksaria/.config/solana/id.json", "utf8" ); const keypair = anchor.web3.Keypair.fromSecretKey( Buffer.from(JSON.parse(secretKey)) ); const endpoint = "https://metaplex.devnet.rpcpool.com/"; const connection = new anchor.web3.Connection(endpoint); const wallet = new Wallet(keypair); console.log("Connected Wallet", wallet.publicKey.toString()); const TOKEN\_METADATA\_PROGRAM\_ID = new anchor.web3.PublicKey( "metaqbxxUerdq28cj1RbAWkYQm3ybzjb6a8bt518x1s" ); // You have to enter your NFT Mint address Over Here const mintKey = new anchor.web3.PublicKey("5iSxT33FyHWsnb8NYSytY17TTXfkFn62FiCyFVFxYhqY"); const [metadatakey] = await anchor.web3.PublicKey.findProgramAddress( [ Buffer.from("metadata"), TOKEN\_METADATA\_PROGRAM\_ID.toBuffer(), mintKey.toBuffer(), ], TOKEN\_METADATA\_PROGRAM\_ID ); // BTW DeGods is my FAV collection although i cant afford one 🥲 const updated\_data: DataV2 = { name: "DeGods", symbol: "DG", uri: "https://metadata.degods.com/g/4924.json", sellerFeeBasisPoints: 1000, creators: [ { address: new anchor.web3.PublicKey( "CsEYyFxVtXxezfLTUWYwpj4ia5oCAsBKznJBWiNKLyxK" ), verified: false, share: 0, }, { address: wallet.publicKey, verified: false, share: 100, }, ], collection: null, uses: null, }; const accounts:UpdateMetadataAccountV2InstructionAccounts = { metadata: metadatakey, updateAuthority: wallet.publicKey, } const args:UpdateMetadataAccountV2InstructionArgs = { updateMetadataAccountArgsV2: { data: updated\_data, updateAuthority: wallet.publicKey, primarySaleHappened: true, isMutable: true, } } const updateMetadataAccount = createUpdateMetadataAccountV2Instruction( accounts, args ); const transaction = new anchor.web3.Transaction() transaction.add(updateMetadataAccount); const {blockhash} = await connection.getLatestBlockhash(); transaction.recentBlockhash = blockhash; transaction.feePayer = wallet.publicKey; const signedTx = await wallet.signTransaction(transaction); const txid = await connection.sendRawTransaction(signedTx.serialize()); console.log("Transaction ID --",txid);})()  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]

**A2:** let blockhashObj = await solConnection.getLatestBlockhash(); updateTx.recentBlockhash = blockhashObj.blockhash; updateTx.sign(alice); let endocdeTransction = updateTx.serialize({ requireAllSignatures: false, verifySignatures: false, }); var signature = await solConnection.sendRawTransaction(endocdeTransction, { skipPreflight: false, });