
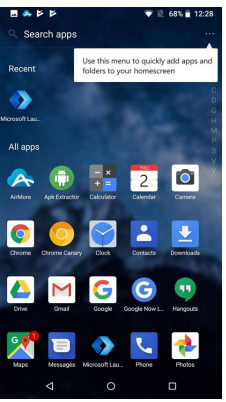


## Delete a file android

 I'm not robot  reCAPTCHA

[Continue](#)

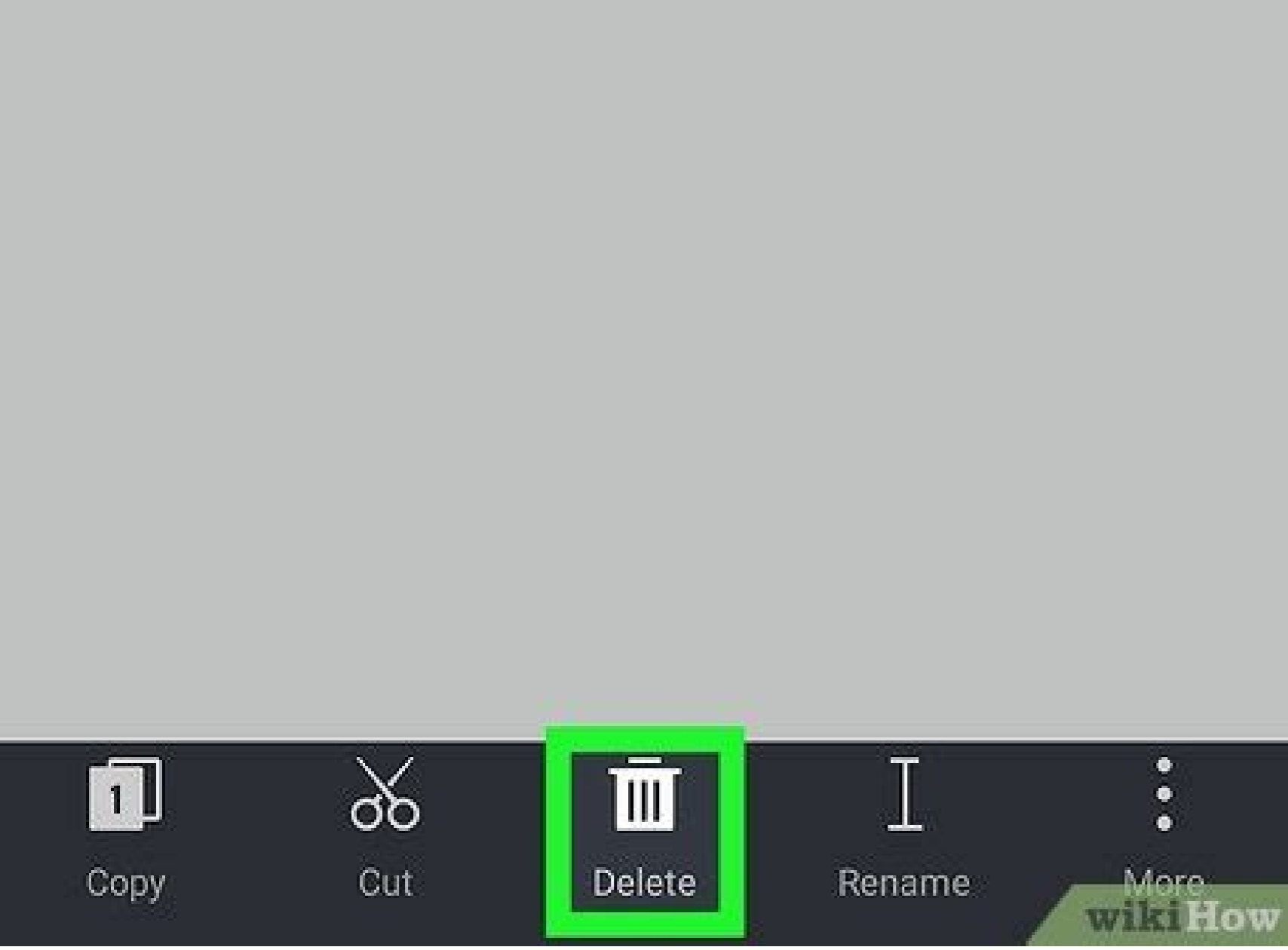




# Home

Decorate your home and visit friends'





I create an application in a gallery style that I need to add image lubrication functionality. I use the RNFS library, especially unlimited. When the feature is completed, the function does not remove the file from my phone. I create a new application with a great template to isolate the problem and add the following code lines: I transfer RNFS "React-Native-FS", {Sia undroid} Z 'react-nat App.TSX. Const usefunction = async () => {Const Wermision = Permissionandroid.Perissions.Write\_EXTERNAL\_storage; Const Hase Mission = Authorization and SisterID.Check (Approval); If (! Hassermission) {permission and sister.request (permission) wait; } Rnfs.unlink (/storarare/emulates/0/download/1.pdf). Then () => {console.log ('file deleted')}; useeffect () => {usefunction ()}; []}; App.TSX application component. The entire code is the permission to write in memory and then delete the file in my file system. If the road is not available, if the road is correct, I will receive the message "error: file: no file", "File deleted" and no error, but the file has not been removed from my device, i still find it with the Android file manager. Usually you will find files in the file application. If you cannot find a file application, it may be another application of the device manufacturer. Learn how to get help with your private device. IMPORTANT: Open the phone application finding and opening files. Learn where to find your applications. Downloaded files are displayed. You want to find more files, click the menu. You want to sort according to name, date, type or size, click the next ranking. "Sorting", if you are not modified or not ranking. You want to open the file, click on it. Delete files to open your phone files. Click the file. "Delete. If you do not see the eraser, click more. Remove, print, save, save the unit, and tap the file and hold the other sharing files. Click the stocks in the upper right corner. And other contents can download files like music. I create an application in the style of a gallery to which I need to add image removal function. I use the RNFS library, specifically the distribution function. When you complete the function, it does not throw an error, but does not even erase the file from my phone. To insulate the problem, I created a new application using the AWESOMEPROJECT and I added the following code lines: import RNF from "React-Native-FS"; import {PermissionsAndroid} with "React-Native", in the application. Const useeffectfunction = Async () => {Const Wermision = Permissions. Const Haslermission = Wait Permissadroid.check (Permission); if (! HASERMISSION) {wait for permissadroid.request (permission); } Rnfs.unlink (/storage/emuid/0/download/1.pdf). Then () => {console.log ('deleted file');}. log (s); }; useeffect () => {usefunction ()}; []}; In App.TSX in the component app. User user user user user user user user user user user user user user user user field calling user user user-"/> in Androidmanifest.xml anything to evaluate this iRKT code, asks for permission to write in memory, and then delete the file on my file system. If the path does not exist, I will see the message "[Error: fight r does not exist] ". If the path is correct, a "deleted file" and no errors will be displayed, but the file will not be deleted from my device, I will always find it using Android file manager. You can usually find files in your phone in your files. If you can't find the files, your device manufacturer may have a different application. Find out how to get help for your particular device. Important: Search and open the files to open the files on your phone. Find out where to find your applications. Your uploaded files are displayed. Press the menu to search for additional files. To sort by name, date, type, or size, click more. If you do not see "sort", click "Edit" or "Sort". Click it to open the file. Open File Removal on your phone. Click the file. Click Delete. If you do not see the Deletion icon, click Delete again. Share, print, follow files and other files, click longer file. Click to share. Perform additional actions such as B. Printing or adding Google disk to open a file. Press them. Search for other options in the upper right corner. Click more if necessary. Search for music, movies and other content. You can record files like music.Or books in different applications. To find this content, go to the app you downloaded it from. For example, "Learn how to find downloaded movies in the Google Play Movies & TV app. After connecting your phone to your computer next to the USB cable, move your files to your computer and you will find your files on your phone. Learn how to move files between your computer and phone. Get community help, get answers from community experts, I used android internal storage to save file for my app. Ex File RootFolder = context.getFilesDir(); AbxidFolder = new File(rootFolder, gealbumid()); I want to remove root folder, so I tried with rootfolder = context.getFilesDir(); AbxidFolder = new File(rootFolder, gealbumid()); AbxidFolder.delete() but it doesn't work as folder I read these answers without deleting but in my case it didn't work, it helped me to solve this problem. I don't understand where am I going Wong. Delete file from internal storage, how to delete internal storage file in android? Edit 2 ex ^ ( Hierarchy ca rtelle) User data 0 file package name 155775346846131 one more time I want to remove folder 155775346846131 after a large range of benchmark tests. We found that using RM -RF was much faster than using a file. Of course, if you have a small or simple catalog, it doesn't matter, but in our case we had a lot of gigabytes and meters heavily invested, which would take more than 10 minutes with files. DeleteDirectory and just 1 minute with RM -RF arrays here's our rough java implementation to do it: // Delete the given directory and all sub-regulars and files (i.e. recursively). // static public boolean selector -ekipator(file) throws OEXCEPTION, URRUPTEEException { if (file.exists()) { String deleteCommand = "rm -rf" + file.getSolutePath(); Time runtime = runtime.getRuntime(); Process Process = Runime.Exec(deleteCommand); Processes. whatform(); return the truth; } returns false; } Worth a try if you're dealing with large or complex catalogs. When did the deleted file actually come out from your device? Only when the last bit used for storage has been swapped. This isn't really a problem for most users, but if you're an international superppy or (slightly more likely) trade or sell old gear, it's important to clean it up to prevent strangers from studying more than you'd like. Shredroid is a free app that replaces free space even if it has itProblems with some Motorola devices (see below). Here's how to clean your Android device: Install Shredroid using the Android Market. Turn it on to set the (pretty basic) options. Select external and/or internal storage and then start as a background process. This is the only way to automate Shredroid, but it can be pig hogs, so manual demolition is probably best if you have a reliable system you can use regularly. It takes time and processing power, but it will be a long time for the data to run out. Motorola users should follow this programmer's FAQ to see when Shredroid will be ready for their device. Lukmanazis/shutterstock.com Completed archiving can make Android an old phone. An easy way to clean up space is to remove duplicate files, but that can be a pain. We show an easy way to do this. Storage on Android can be a bit cumbersome, so it's no surprise that file copies can be created without your prompt. A long list of files and manual copies are no fun. Fortunately, Google App Files can help. Related: how to slow down storage for android phone and android files some android devices is previously written. If you don't have one, you can download it from the Google Play Store. You first open the app when Google is asked to agree to its services and confidentiality. I agree, if accepted, continue. Give app access to files by canceling permissions. Now we can start with everything. First, switch to the "cleanup" tab in the bottom lane. Give the program a second or two to load everything. You should see the "Delete" tab somewhere on this screen. A touch, select files to import. You may need to enable the application to upload files a second time. Now you will see all the duplicate files listed next to the original marked file. You can tap all copies or manually select files to save the original and delete all copies. After selecting "Move the book button to the bottom of the screen. Tap the recycle files again to confirm. AS! The files will be thrown into the trash and permanently deleted after 30 days. There are many other ways to clean Google app files.It is a handy tool to clean Android. Parents: such a free space on the Android phone with Google apps that cannot be written (delete, edit...) to external storage in the specific address of the package. In the Android documentation it says: "The application should not write to secondary external storage devices except their own package-specific directory as allowed by synthesized permissions." However, there is one nasty problem (see code below). Tested on Samsung Galaxy S4, but this fix does not work on all devices. Also, I wouldn't count on this issue being available in future versions of Android. There is a great article explaining the external storage (4.4+) permission change. You can read more about the solution here. The source code for the build release comes from this website: Public class MediaFunctions (@targetapi(build.version codes.honeycomb) public DeteeViacontentProvider(context context, full name) { uri = getFileur(context, full name); if (uri == zero) { return false; } Try { ContentResolver Resolver = context.getContresolver(); // Change type to image, otherwise nothing will be folded contentValues ContentValues = new ContentValues(); int medi type=1;ContentValues.Put("media type","media type");Resolver.update(URI, contentValues, zero, zero).Return resolver.delete(uri, zero, zero) > 0; } to catch(throw e) { return false; } } @targetapi(build.version codes.honeycomb) static static uri getFileur(context context, full string name) { //note: see this class if your os version >= 11 uri =zero;cursor slider=zero;ContentResolver ContentResolver=zero;Try { contentResolver = context.getContentersolver(); If (contentResolver == zero) returns zero; Uri = meditore.files.getConturi("external"); String[] project = new String[2]; Screening[0] = " id"; Projection[1] = " data"; String SELECTION = " data =?"; // This prevents the insertion of the SQL String[] parameter select = new String[1]; Selection parameter [0] = full name; String sortOrder = " id"; Cursor = ContentResolver.Query(URI, projection, selection, selection parameters, deaf); if (cursor!=zero) { try { if (cursor. getCount() > 0) // File is available! {Cursor. movetoFirst(); int dataColumn = cursor. getColumnIndex("data"); String s = cursor. getString(dataColumn); If (!s. equals (full name) returns zero; int idColumn = cursor. getColumnIndex(" id"); Long id = cursor. getLong(idColumn); Uri = } Other // File is not in the media database! {Contentvalues Contentvalues = New Contentvalues (); Contentvalues. Set ("\_ data", full name); Uri = mediattore.files.getconturi ("external"); Uri = ContentResolver.inrsert (Uri, Contentvalues); } Catch (discard E) {Uri = zero; } Finally {cursor.close ()}; } Catch (one -Time E) {Uri = zero; } Return Uri; } }

