Jul 24, 2023
Megan Peters

# How to save image, PDF, or text to Apple Notes **from an Android or Windows device** semi-automatically using AppleScript and a cloud sync service of your choosing

## Requirements:

- A Mac running Apple Notes
- The internet
- A non-Apple device you want to save stuff from

## Overview:

Are you sick of not being able to save images, PDFs, or other attachments to Apple Notes directly from a non-Apple device? Now you can!

This process will allow you to save any PDF, text file, or image directly to Apple Notes from an Android or Windows device. I use it for exporting from Samsung Notes, but you could use it for anything really. Each new PDF or image added to the folder will be turned into a new note in the Notes folder of Apple Notes. (You could edit the script to have a different destination within your Notes file system if you wish.) If you have iCloud sync turned on, they will then also sync across all your Apple devices. PDFs and images won't be editable, but they will be within the rest of your workflow. Text files can be opened and then copy-pasted into Apple Notes. There's probably an automatic way to do that too, but I really wanted my handwriting on my Galaxy S23 Ultra to be the main target of this workflow, so I haven't worked out the automatic text import yet (but I'm sure it's possible with a little digging). You could also just type in the Apple Notes web interface for text so that seems less critical for me.

**Note:** Automatic creation of Apple Notes will only work "nearly instantly" if you have a Mac that is constantly awake and connected to the internet. Otherwise, it will happen when you wake up your Mac so that the sync can be triggered.

**Known weird behavior:** The newly-created notes will have a weird, big space before the attached file. There are some posts on the internet written about this. I have not been able to get it to go away so I just deal with it.

Ready? Let's get started.

## Let's do it:

**Step 1: Prepare the "folder actions" script folder so you have write access**

1. Navigate to Macintosh HD / Library / Scripts / Folder Action Scripts
2. “Get info” on this folder, or hit Apple-i
3. Set permissions so that “everybody” has read/write access

**Step 2: Make sure you have sync turned on for whatever sync service you’re using (Google Drive, Dropbox, etc)**

**Step 3: Decide which folder within your synced folders you want to use as a “dropbox”.**

1. Get the full path of that folder. For example:
   Macintosh HD/Users/YOURACCT/My Drive/NotesTest/
2. Make a new sub-folder inside this target dropbox folder called ‘processed’. This directory will house the files you drop in the main folder AFTER they have been turned into Apple notes, so they don’t get processed multiple times.

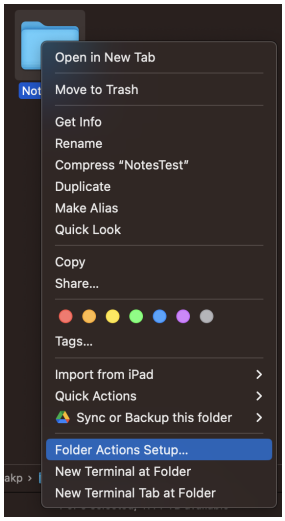
**Step 4: Open Script Editor on your Mac. Copy-paste the following into a new script. Or save the .scpt file in this github folder. Make sure you save the new script to Macintosh HD / Library / Scripts / Folder Action Scripts.**
You will need to make 1 change before saving your script: You need to set the destination folder for ‘processed’ files to the sub-directory inside YOUR main watched folder. To do this, you will need to format the path replacing the slashes with colons. For example, with the path above, you’ll need to change Macintosh HD:Users:YOURACCT:My Drive:NotesTest:processed: to the path to your processed folder in your own directory structure. You then need to paste this into the appropriate spot in the script, where it says “set dst to folder (“PATH”).

```
on adding folder items to theAttachedFolder after receiving theNewItems

    --display notification "i'm triggered" -- uncomment this if you want to know whether the trigger is working
    --delay 1 -- uncomment this if you want to know whether the trigger is working

    tell application "Finder"
      set theName to name of theAttachedFolder
      set dst to folder ("Macintosh HD:Users:YOURACCT:Desktop:NotesTest:processed:")

      -- Count the new items
      set theCount to length of theNewItems

      -- Loop through the newly detected items
      repeat with anItem in theNewItems

            set thePDF to anItem as text
            set |note name| to name of file thePDF

            -- Process the current item
            tell application "Notes"
                  --set |new note| to make new note with properties {name:|note name|, body:|note body|}
```

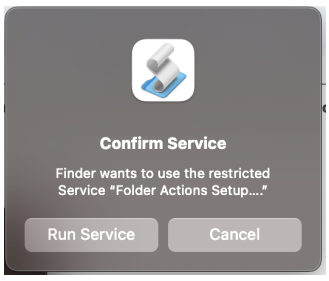
```
                set |new note| to make new note with properties {name:|note
name|}
                make new attachment at end of attachments of |new note| with data
(file thePDF)
                show |new note|
            end tell

            -- Move the current item to another folder so it's not processed again in the
future
            move anItem to dst

        end repeat
    end tell

end adding folder items to
```

**Step 5: Add the script as a folder action script.**

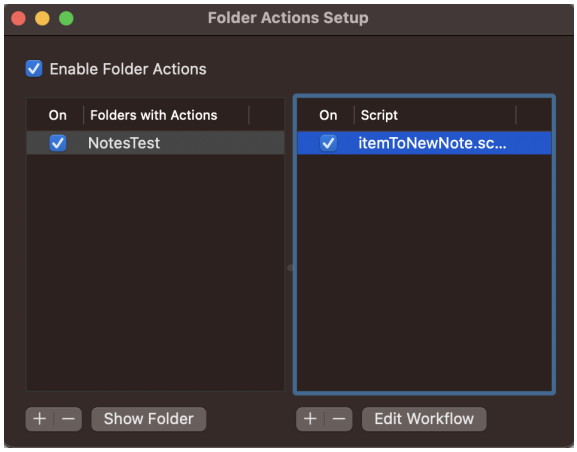
1. Right-click on the target directory in your sync folder (not the ‘processed’ sub-folder, the main target directory).
2. Choose “Folder Actions Setup” from the contextual menu:




3. Click “Run Service” on this pop-up:




4. Click the “plus” on the right-hand window, and choose your new script from the dialog. Make sure it is checked, and also check “Enable Folder Actions”.




**Step 6: Test the setup.**
On your Mac, move any random PDF, image, or text file into your target folder. If any permissions dialogs show up, click “yes, allow”.

**Step 7: Send an item from your Android or Windows device to Apple notes.**
Now turn to your non-Apple device. Save a PDF, image, or text file of your choosing to the cloud folder on that device. For example, on my Samsung Galaxy I save a PDF of a Samsung Note to the Google Drive folder that is being watched via the above workflow. As soon as my Mac syncs that file from the cloud, the “watched folder action” is triggered and a new note shows up in Apple Notes!