Requirements:

* Global List Variable Defined at the top of the script called PATHS = [].  Inside of this list I will provide paths that I want scanned.  Ie: ['/home/media/tv/', '/home/media/movies', '/home/recorded/tv']
* Also create a global variable under PATHS called GARBAGE\_FOLDER which is where these files will be moved to
* **You're going to need recursion here, don't get too caught up in what it is because it's a very advanced thing .... here's all I want you to do with it,**
  + define a function called find\_files\_to\_remove(path) that takes a path and does an os.listdir on the items inside of it.
  + Do a for loop over the items that os.listdir returns and check to see if they are a folder using os.isfile or os.isfolder whichever you want.
  + If it's a file run another function that you create called check\_file(file) which takes the file as a parameter (we'll get to what it'll do next).
  + If it's a folder however I want you to call files\_to\_remove(path) on it. So that from inside of the function you are calling itself.  What that will do is end up walking all of the sub-folders in a directory.
* Inside of the check\_file function
  + first check to see if the file has a .nfo extension, if it does mark it for removal,
  + if it doesn't check to see if the word 'sample' is in the filename (you're going to want to perform .lower() on the string so that you can bypass any case-sensitivity issues before you do this check) ...
  + if the word sample is in the name then check to see if its filesize is less than 50 MB's ...
    - << found workaround. I feel like there’s a better way to do this. **REVISIT** >>
  + if it is mark it for removal by appending the file path to to a to\_remove list.
* Once you finish looking at all of the files in a given directory then go about moving them one-by-one ... there's a function in OS that'll let you move them.

Check your destination file to see if a file of that name already exists

use an os function for it

os.isfile or something suffices

If it does exist, start a for loop in range of say 50

So first

if (os.exists(destination)):

for append\_num in range(0,50):

Then on every loop see if that file exists with a join of its name and a dash with append\_num

ex:  
  
for append\_num in range(0,50):

if os.exists("{}-{}".format(destination,append\_num)

It'll try that for every number up until 50

And hell if there are 50 things that all havet he exact same name, 'eff it - overwrite the first one in the list with copy -0

Make sense?

os.path.isfile    rather