

Forum:[Hack Forums](#)

Thread: [Requesting Pastebin Link Grabber](#)

Goodies - June 18, 2014, 6:58 pm

I am doing a project and I'd really like to see if there is a pastebin link grabber already made in python. If not, I'll end up writing one anyways, but I'd really like to see if there is one, or at least a basic idea of one, already.

Mine would just request the home page and grab the "Public Paste" section, save them in a tuple/array/dictionary/whatever python calls them, check for duplicates, and process from there, then repeat. It wouldn't take too long, but time is money.

Thanks, guys!

Corpinator - June 18, 2014, 7:44 pm

I think im going to try to make this for some practice

EDIT: I did find one on the internet though i will pm you it

fork-on-loop - June 19, 2014, 5:24 pm

Something like this?

```
Code import BeautifulSoup
```

```
import urllib2
```

```
import time
```

```
import Queue
```

```
import threading
```

```
import sys
```

```
import datetime
```

```
import random
```

```
import os
```

```
pasteseen = set()
```

```
pastes = Queue.Queue()
```

```
def downloader():
```

```
    while True:
```

```
        paste = pastes.get()
```

```
        delay = 5 # random.uniform(1, 3)
```

```
        fn = "pastebins/%s-%s.txt" % (paste, datetime.datetime.today().strftime("%Y-%m-%d"))
```

```
        content = urllib2.urlopen("http://pastebin.com/raw.php?i=" + paste).read()
```

```
        if "requesting a little bit too much" in content:
```

```
            print "Throttling... requeueing %s" % paste
```

```
            pastes.put(paste)
```

```
            time.sleep(5)
```

```
        else:
```

```
            try:
```

```
                f = open(fn, "wt")
```

```
                f.write(content)
```

```
                f.close()
```

```
            except:
```

continue

```
sys.stdout.write("Downloaded %s, waiting %f sec\n" % (paste, delay))
time.sleep(delay)
pastes.task_done()
```

def scraper():

```
    scrapecount = 0
    while scrapecount < 10:
        html = urllib2.urlopen("http://www.pastebin.com").read()
        soup = BeautifulSoup.BeautifulSoup(html)
        ul = soup.find("ul", "right_menu")
        for li in ul.findAll("li"):
            href = li.a["href"]
            if href in pastesseen:
                sys.stdout.write("%s already seen\n" % href)
            else:
                href = href[1:] # chop off leading /
                pastes.put(href)
                pastesseen.add(href)
                #sys.stdout.write("%s queued for download\n" % href)
        delay = 12 # random.uniform(6,10)
        time.sleep(delay)
        scrapecount += 1
```

def scraper_two():

```
    scrapecount = 0
    while scrapecount < 10:
        html = urllib2.urlopen("http://pastebin.com/archive").read()
        soup = BeautifulSoup.BeautifulSoup(html)
        ul = soup.find("table", "maintable")
        for li in ul.findAll("td"):
            try:
                href = li.a["href"]
                if href in pastesseen:
                    sys.stdout.write("%s already seen\n" % href)
                else:
                    href = href[1:]
                    pastes.put(href)
                    pastesseen.add(href)
            except:
                continue
        delay = 12
        time.sleep(delay)
        scrapecount += 1
```

num_workers = 1

```
for i in range(num_workers):
    t = threading.Thread(target=downloader)
    t.setDaemon(True)
    t.start()
```

if not os.path.exists("pastebins"):

```
os.mkdir("pastebins")
```

```
s = threading.Thread(target=scrapper_two)
```

```
s.start()
```

```
s.join()
```