

Forum:[Hack Forums](#)

Thread: [Writing a simple python Viewbot](#)

Variez - June 12, 2020, 8:05 pm

Introduction:

Hey, so I've always personally loved bots. Bots are fascinating in the way they interact with webpages, the intricate bypassing, and more importantly the money behind them ? . This tutorial will give a brief look into: web requests, python, multi-threading, and web-based botting.

Part One [What is a Bot?]:

Depending on who you ask this definition may vary however I consider a bot to be something that automates a task. For this we will be using python. For setup I highly recommend using sublime text and installing yourself python3.7 however any text editor will do. Once you've got these installed fire up command prompt and type the following command:

Code pip install requestsThis is important so we can get our libraries installed, if we don't get this installed then we can't use any of its modules.

Part Two [Writing our Bot]:

We are going to want to import the three libraries which we are using for this bot, in this case we will be using the random library , threading library, and also the requests library.

After we have that imported lets get onto writing our viewbot itself:

Code headers = {'user-agent': "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.116 Safari/537.36"}One of the things which we will want to define is our user agent. When we click on a url we our sending it specific information in the header itself. This information tells the webpage what to load, however with botting scripts we use this make it look like our bot is a real person.

next we want to create our function for loading our page, notice how I pass both the argument url,proxy these will be variables which I use in my program itself and MUST be passed whenever you call our function. Another line in this code is:

Code s = requests.session()This is defining our snippet of code so we can call it just by using s.

Now onto the fun stuff :) :

So if you click f11 and then click head over to the network tab then hit f5 you can reload the page and see whats actually being loaded. So in this example we see 2 pieces of really important information when loading the page itself:

1. Its a get request
2. The Request URL

The are both incredibly important when writing your code

So the code above is doing something super important first one being we are sending a get request this is written as:

Code s.get()We already defined what s is, which means we don't have to write more code. Next we are passing our url for the site (2) which has not been defined yet since its passed by the function itself so this is blank till we call it in later code. Another thing we are passing is

Code headers=headersWe are now using the fake user agent which I used above, and sending this

in the actual get headers itself, this makes it look like our bots just a normal user. The last thing in our actual code is passing our proxy which looks like this:

```
pr
Code oxies=proxy
```

This is so we are sending a new view request, this helps to quickly rack up views, however with some forums its not needed.

This code here is written by the wonderful Sheepy, I've just been reusing this code for getting my proxies mainly because of how lazy I am, regardless I'm going to use it in my example. So this code block is getting a random line from your text file which is called proxies.txt. After it grabs your proxy it uses regular expressions and re formats it so it looks something like "https : / / proxyhere" then it attempts to make a request to www[.]google[.]com and if it times out in under 3 seconds then it decides to not use this proxy. If you want to change the timeout time just change the value which happens to be 3 to the number of your choice.

```
Code def getproxy():
    list = open("proxies.txt", "r").readlines()
    while(True): #This code is written by sheepy so thanks to him <3, I am way to lazy to rewrite this
working code ;)
        try:
            proxy = {"https": "https: //{}".format(random.choice(list).rstrip())}
            if(len(proxy['https']) <= 1):
                continue
            ret = requests.get("URL HERE SINCE HF IS PREVENTING ME", proxies=proxy, timeout=3)
            return proxy
        except:
            passOk now on to the final block of code, this is where we actually use multithreading in our
program to quickly bot as many views as possible:
```

So we are sticking our code in a while loop just to keep it running forever since I assume your also like me, lazy. However if you wanted to calculate your views you could tell it just be in a for loop and set the range(100) to range(yourNumber) the for loop is creating multiple threads to view the page. Next in our code we are creating a thread thats target is view_page with the arguments you give, which is the url of your page and also the retrieval of the proxy itself. We end this all with start to tell our code to start view_page code with the following arguments.

Part Three [Conclusion]:

Bots are incredibly interesting in the ways they can automate tasks that no one is willing to do by hand. I personally love bots all of types, robots, web bots, sex bots, ect. Regardless its considered polite to check out the robots.txt and see what is allowed for interacting with the site itself. I hope you enjoyed this tutorial on creating a simple viewbot. If you enjoyed consider leaving a like, starring it or answering the poll :D

Part Five [Final Code]:

Since HF is clowning on this HQ post check this pastebin link:
[www\[.\]pastebin\[.\]com/tKFA51yM](http://www[.]pastebin[.]com/tKFA51yM)

Here is the finished code for those of you that are to lazy to write it out on your own, I highly encourage for people to write it by hand because otherwise your not retaining the information.

Is this used for Twitch? Great tutorial by the way! I am trying to learn Python and this will help.