

Forum:[Hack Forums](#)

Thread: [Twitch Username Scraper](#)

iKQLY - August 25, 2014, 2:52 pm

Saw this being sold in the Premium Sellers Section. I really don't believe people should have to pay for something that takes only an hour or two, so I'll release my twitch scraper.

It is multi-threaded so it shouldn't lag when scraping

All you need to do is enter in a channel name. The listbox will then populate all viewers watching. You can then export to a text file.

If anyone wants to comment on the code and possibly suggest any way it could be optimized, feel free to post here.

Source Code:Spoiler [\(Click to View\)](#)
[Mediafire link](#)

Application Only:Spoiler [\(Click to View\)](#)
[Mediafire link](#)

Image: Spoiler [\(Click to View\)](#)

Credit to Tidas for theme:

<https://hackforums.net/showthread.php?tid=3569721&highlight=Twitch+Theme>

+Haze - August 25, 2014, 3:08 pm

What theme is that, looks nice IMHO.

iKQLY - August 25, 2014, 3:12 pm

(Aug 25, 2014 03:08 PM)+Haze Wrote: [\[CITE\]](#) What theme is that, looks nice IMHO. Thanks :) It's Twitch Theme, I think Tidas released this.

Here's link for theme

<https://hackforums.net/showthread.php?tid=3569721&highlight=Twitch+Theme>

Charles Carmichael - August 25, 2014, 3:30 pm

Whats the purpose of scraping mass twitch usernames :o

DBots - August 25, 2014, 6:11 pm

Okay I'll bite.

On your form load event, you have this:

Don't do that. That is not proper threading which is what I suppose you were trying to accomplish here. Once you remove that, you'll be on your way to properly threading this since you won't be suppressing any errors and you can more easily debug your application.

--

You instantiate a WebClient but then it is possible you may not use it and didn't even dispose of it.

Rather you check if there may be something stopping you from proceeding first.

--

Here, you acknowledge there may be an error and when there is, you still try to process the result, which there isn't one.

I would rather take the approach of making sure the string was downloaded before moving on to trying to process it.

--

You call this at the end but there is no need to.

+Haze - August 25, 2014, 7:27 pm

I agree with what dbots said. I see you're trying to use Multi threading. In that case use WebClient download string async method. It's so much better and faster when dealing with large amounts of data or doing things quickly.

iKQLY - August 25, 2014, 8:43 pm

(Aug 25, 2014 06:11 PM)DBots Wrote: [\[CITE\]](#) Okay I'll bite.

On your form load event, you have this:

Don't do that. That is not proper threading which is what I suppose you were trying to accomplish here. Once you remove that, you'll be on your way to properly threading this since you won't be suppressing any errors and you can more easily debug your application.

--

You instantiate a WebClient but then it is possible you may not use it and didn't even dispose of it.

Rather you check if there may be something stopping you from proceeding first.

--

Here, you acknowledge there may be an error and when there is, you still try to process the result, which there isn't one.

I would rather take the approach of making sure the string was downloaded before moving on to trying to process it.

--

You call this at the end but there is no need to.

Awesome, thanks! Just added in what you pointed out

(Aug 25, 2014 07:27 PM)+Haze Wrote: [\[CITE\]](#) I agree with what dbots said. I see you're trying to use Multi threading. In that case use WebClient download string async method. It's so much better and faster when dealing with large amounts of data or doing things quickly.

Thanks, never really bothered to look at async method. Just added it as well