ICSSC Workshop: Build a Reddit Bot in Python

https://github.com/billycastelli/WikiBot-Workshop

Workshop by Will Castelli

What is Reddit?

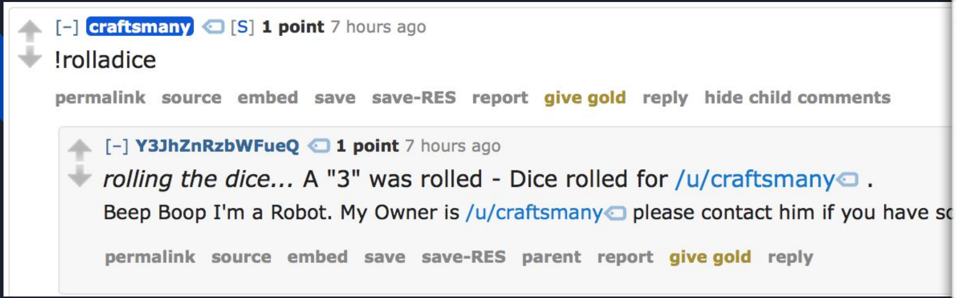
- Community based website comprised of “subreddits” dedicated to a particular topic.
- Submissions voted to the “frontpage” of the site.
- Millions of submissions, millions of comments (good for us!).
- Check out /r/learnpython for some more information about Python.

What is a bot?

- A bot is an application that runs a script, autonomously.
- “Bots perform tasks that are both simple and structurally repetitive, at a much higher rate than would be possible for a human alone”.
- Can you imagine looking through every comment posted on Reddit for a specific keyword?

Why build bots?

- Can be useful for other Redditors.
- Building a bot is an interesting way to show off your Python skills.
- Companies use bots to get information about users.
- Can you think of any Reddit bots you’ve seen?



[–] craftsmany [S] **1 point** 7 hours ago

!rolladice

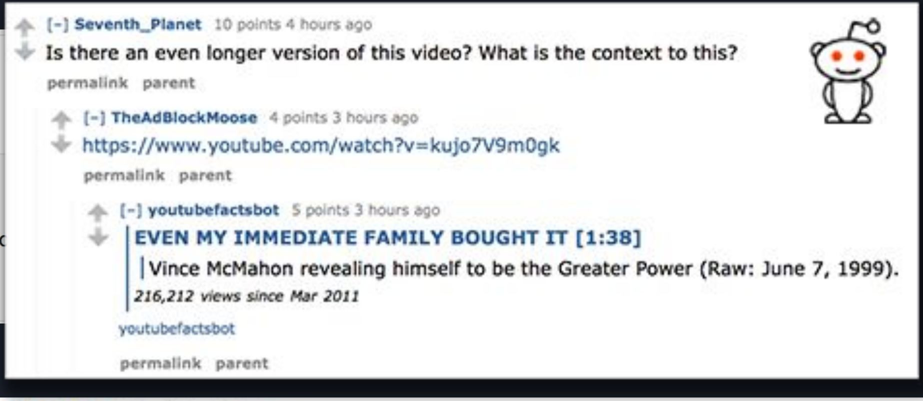
permalink source embed save save-RES report give gold reply hide child comments

[–] Y3JhZnRzbWFueQ **1 point** 7 hours ago

rolling the dice... A "3" was rolled - Dice rolled for /u/craftsmany .

Beep Boop I'm a Robot. My Owner is /u/craftsmany please contact him if you have sc

permalink source embed save save-RES parent report give gold reply



[–] Seventh_Planet 10 points 4 hours ago

Is there an even longer version of this video? What is the context to this?

permalink parent

[–] TheAdBlockMoose 4 points 3 hours ago

https://www.youtube.com/watch?v=kujo7V9m0gk

permalink parent

[–] youtubefactsbot 5 points 3 hours ago

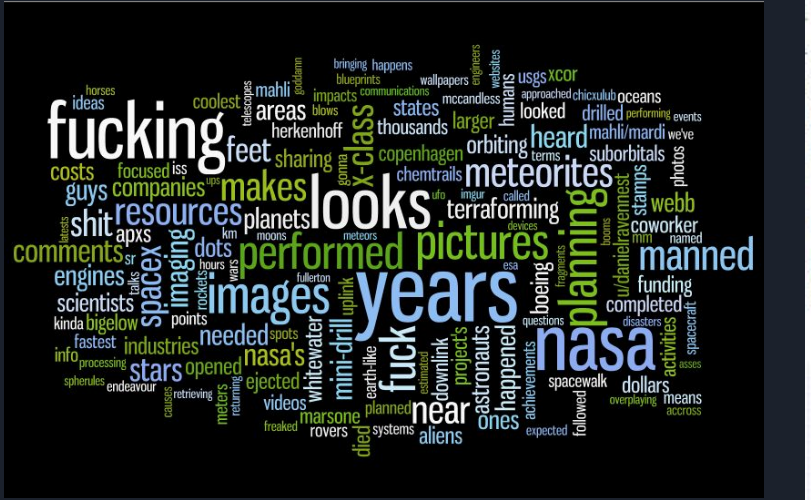
EVEN MY IMMEDIATE FAMILY BOUGHT IT [1:38]

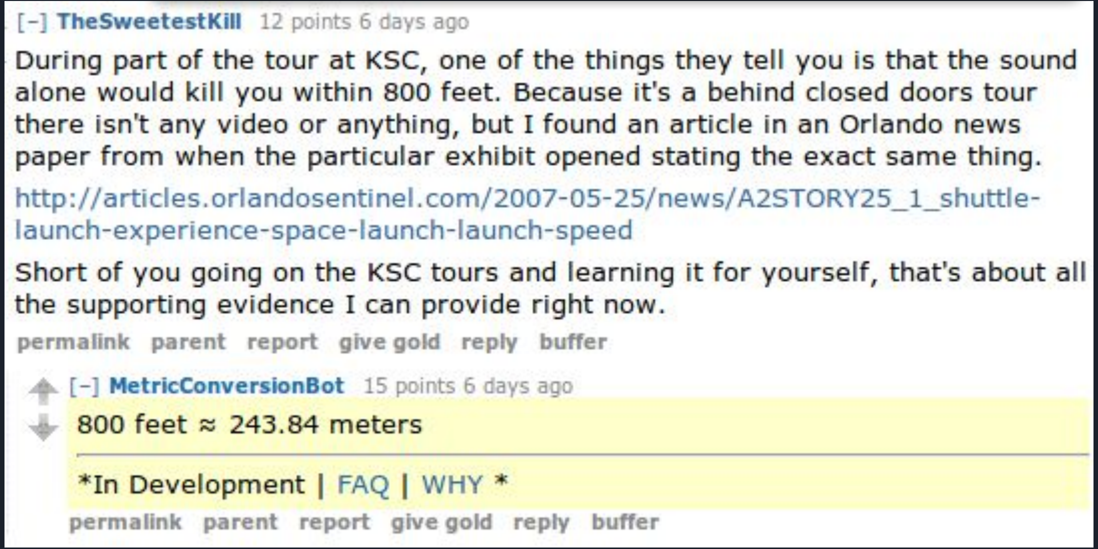
Vince McMahon revealing himself to be the Greater Power (Raw: June 7, 1999).

216,212 views since Mar 2011

youtubefactsbot

permalink parent





[–] TheSweetestKill 12 points 6 days ago

During part of the tour at KSC, one of the things they tell you is that the sound alone would kill you within 800 feet. Because it's a behind closed doors tour there isn't any video or anything, but I found an article in an Orlando news paper from when the particular exhibit opened stating the exact same thing.

http://articles.orlandosentinel.com/2007-05-25/news/A2STORY25_1_shuttle-launch-experience-space-launch-launch-speed

Short of you going on the KSC tours and learning it for yourself, that's about all the supporting evidence I can provide right now.

permalink parent report give gold reply buffer

[–] MetricConversionBot 15 points 6 days ago

800 feet ≈ 243.84 meters

*In Development | FAQ | WHY *

permalink parent report give gold reply buffer

Rules for bots (“bottiquette”)

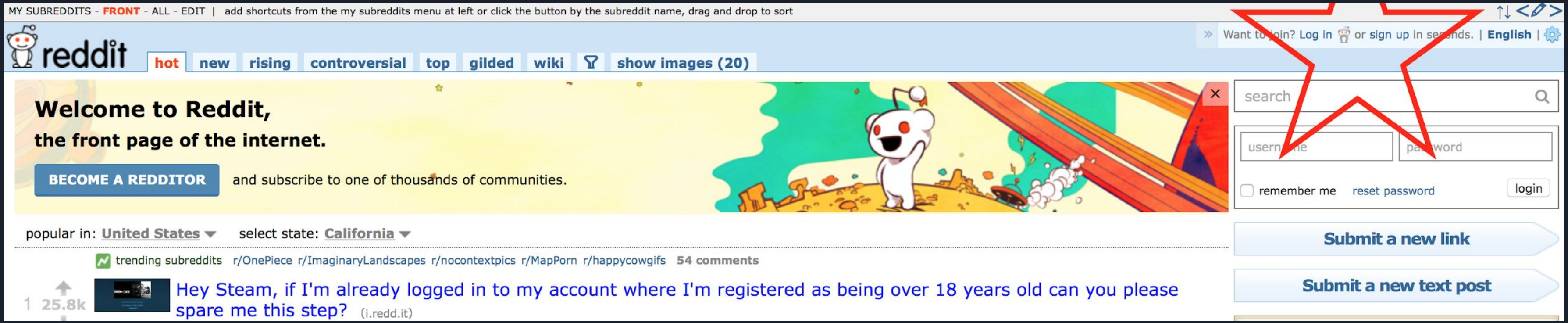
- Use common sense (don’t use a bot on /r/suicidewatch, /r/depression...)
- Don’t respond to common words or phrases.
- Don’t harass people/groups of people…
- Don’t upvote/downvote with your bot.
- You will get banned if you don’t follow the rules.
- Read full rules: https://www.reddit.com/wiki/bottiquette

Our goal tonight...

- Create Reddit a Reddit bot that replies to comments with a Wikipedia summary.
- Our bot will…
  1. Respond to comments with a keyword.
  2. Respond to comments with a wikipedia.org link.

For starters....

- Lets create new Reddit accounts! (You can use your main account, but don’t blame me if you get banned for doing something wrong…)
- Remember your username and password.



MY SUBREDDITS - FRONT - ALL - EDIT | add shortcuts from the my subreddits menu at left or click the button by the subreddit name, drag and drop to sort

Want to join? Log in or sign up in seconds. | English |

Welcome to Reddit,

the front page of the internet.

BECOME A REDDITOR and subscribe to one of thousands of communities.

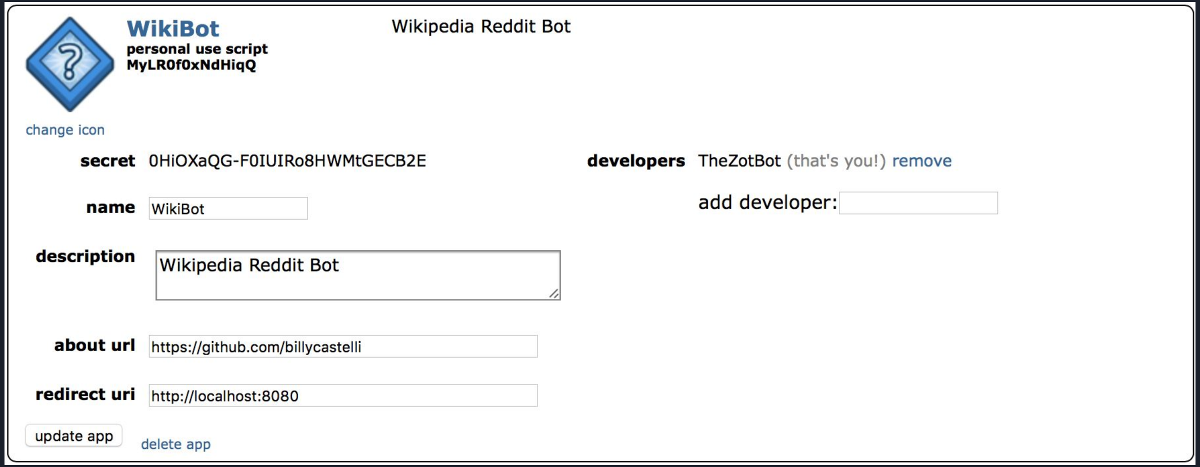
popular in: United States select state: California

trending subreddits r/OnePiece r/ImaginaryLandscapes r/nocontextpics r/MapPorn r/happycowgifs 54 comments

1 25.8k Hey Steam, if I'm already logged in to my account where I'm registered as being over 18 years old can you please spare me this step? (i.redd.it)

Registering our bot with Reddit

- Go to https://reddit.com/prefs/apps/
- Create a SCRIPT.
- Input a name, description, and an about url.
- Scripts don't use the redirect uri, so just put http://localhost:8080 and forget about it.



WikiBot
personal use script
MyLR0f0xNdHiqQ

Wikipedia Reddit Bot

secret 0HiOXaQG-F0IUIRo8HWMtGECB2E

developers TheZotBot (that's you!) remove

add developer:

name WikiBot

description Wikipedia Reddit Bot

about url https://github.com/billycastelli

redirect uri http://localhost:8080

KEEP THIS PAGE OPEN!
We need it soon...

About Praw

- PRAW (Python Reddit API Wrapper) gives us an **easy** way to use the Reddit API for our bots.
- Abides by all the Reddit API rules so we don’t have to worry about breaking them.

- Read the docs here: https://praw.readthedocs.io/en/latest/

About Wikipedia (a python package)

- Wikipedia is a Python package that gives us easy access to Wikipedia information.
- Much easier than parsing the HTML ourselves (I don’t think we have time for that)
- More on this later

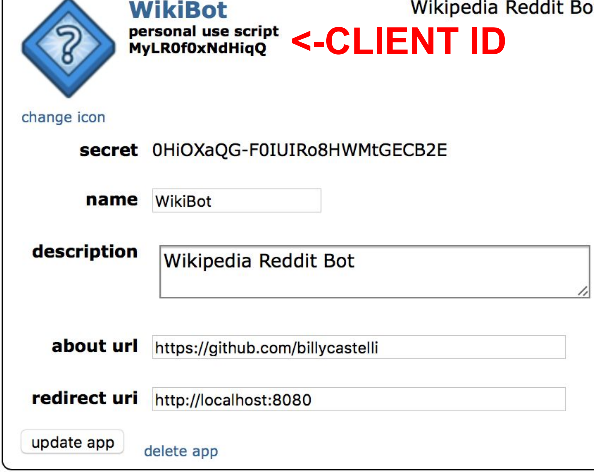
- Read the docs: https://wikipedia.readthedocs.io/en/latest/code.html#api

Let’s begin...

- Open a Terminal.
- `pip3 install praw`
- `pip3 install wikipedia`

Open your favorite text editor

- Let’s start with the basics.
- First, we need connect to Reddit.
- We need to obtain a “Reddit Instance”
- Remember the app registration page? Go back and get your client_id and client_secret id.
- User agent should be a unique string in the form of `<platform>:<app ID>:<version string> (by /u/<Reddit username>)`



WikiBot
personal use script
MyLR0f0xNdHiqQ <-CLIENT ID

Wikipedia Reddit Bo

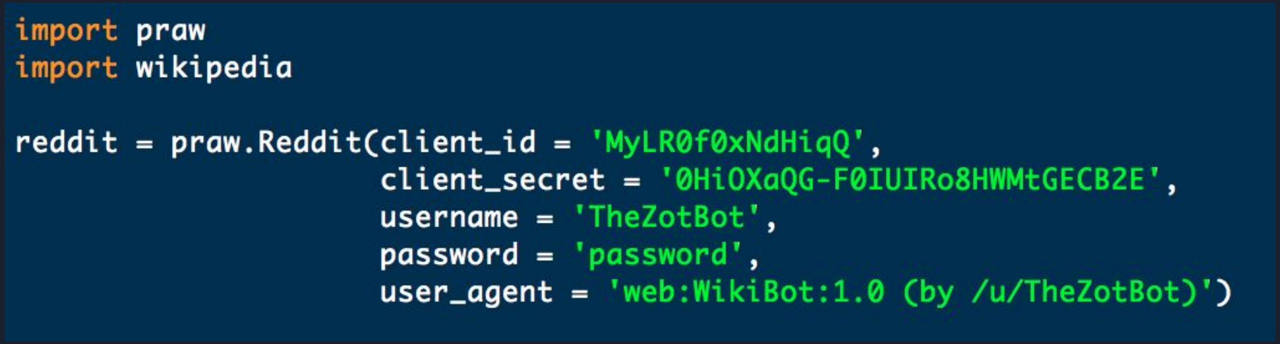
secret 0HiOXaQG-F0IUIRo8HWMtGECB2E

name WikiBot

description Wikipedia Reddit Bot

about url https://github.com/billycastelli

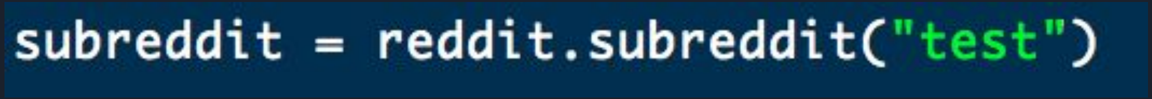
redirect uri http://localhost:8080



```python
import praw
import wikipedia

reddit = praw.Reddit(client_id = 'MyLR0f0xNdHiqQ',
                     client_secret = '0HiOXaQG-F0IUIRo8HWMtGECB2E',
                     username = 'TheZotBot',
                     password = 'password',
                     user_agent = 'web:WikiBot:1.0 (by /u/TheZotBot)')
```

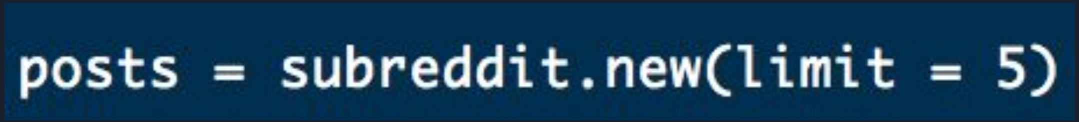
Still setting up...

- Create a subreddit instance (you can enter any subreddit...)



```
subreddit = reddit.subreddit("test")
```

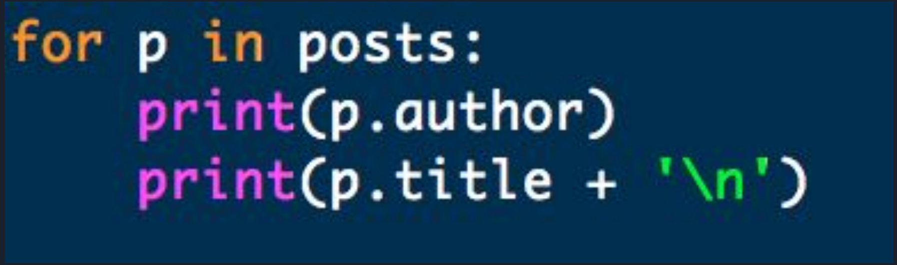
- And to get an iterable object containing posts (specify new, hot, or top, and how many posts)...



```
posts = subreddit.new(limit = 5)
```

Starting to get information

- Lets see if it works. Lets print out some information about these posts.



```
for p in posts:
    print(p.author)
    print(p.title + '\n')
```

- Hopefully you see the information for your posts. You can go to the subreddit in a browser to verify this information is correct if you can’t believe your eyes.

FYI....

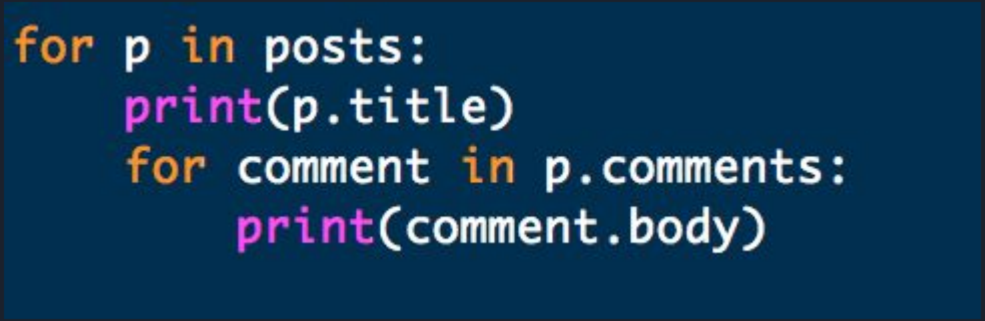
- We can do ALOT with posts. We can interact by replying or upvoting. We can get information about the author, subreddit, etc.
- To see all attributes and methods of a post object, you can run

```
print(dir(p))
```

- You can only read the docs as well.

Obtaining comments

- Comments are arguably the most interesting piece of data we can get from Reddit.
- We can view the comments from the posts we just accessed (limit = 1 to test)

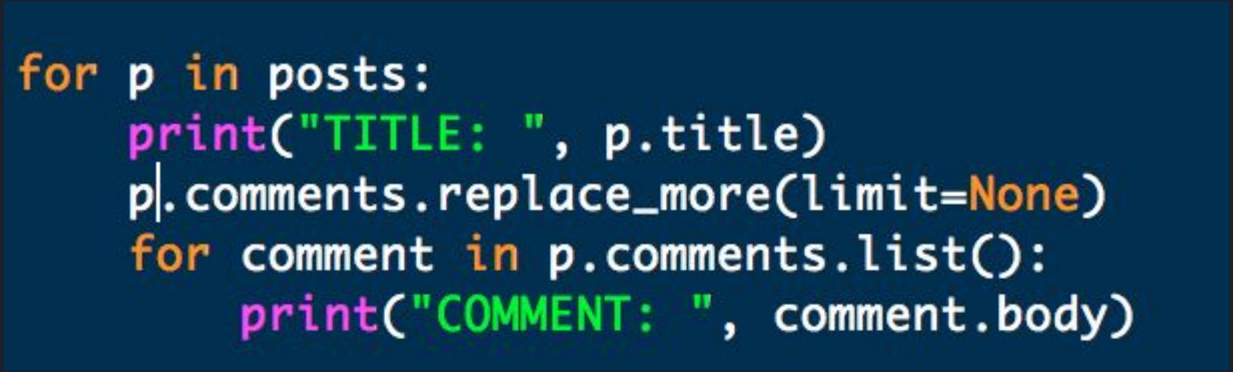


```
for p in posts:
    print(p.title)
    for comment in p.comments:
        print(comment.body)
```

Notice anything strange?

Getting (pretty much) all comments

- We want to get all the comments, not just parent comments.
- This is a recursive problem….
- Luckily, praw provides a method .list() that will give us all the comments without having to write our own recursive method.
- Combine with .replace_more() to obtain pretty much all comments on a post.



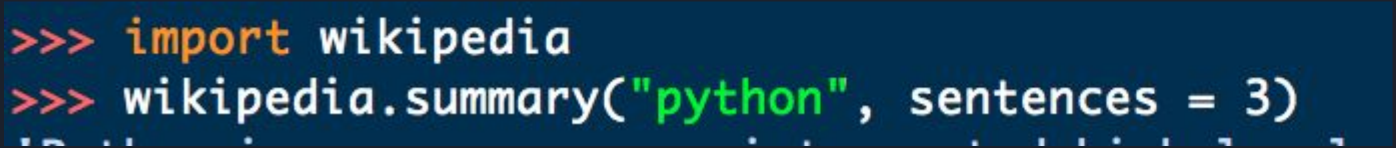
```
for p in posts:
    print("TITLE: ", p.title)
    p.comments.replace_more(limit=None)
    for comment in p.comments.list():
        print("COMMENT: ", comment.body)
```

Possible errors...

What can we do with this information?

A side note…
Errors may occur with the summary method!
For now, best practice would be wrapping it in a try/except.

- Now, we have enough information to create a basic bot.
- Remember our goal? Creating a wikipedia bot.
- Test out how Wikipedia works in the shell…

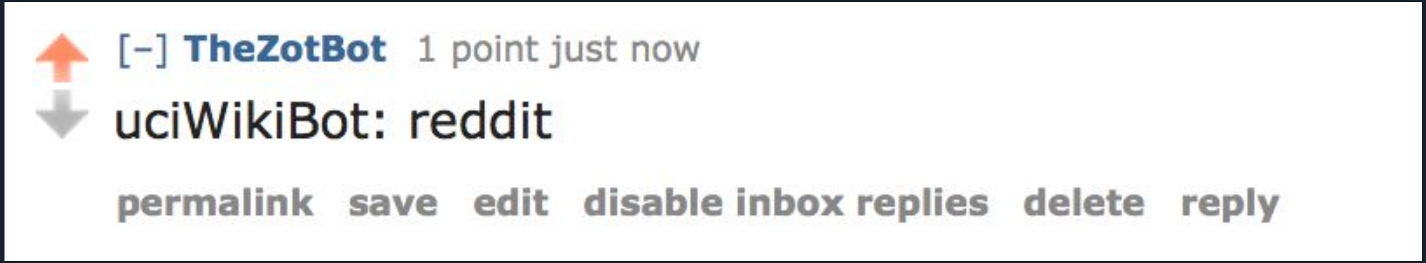


```
>>> import wikipedia
>>> wikipedia.summary("python", sentences = 3)
```

- You should get a string with the first 3 sentences of the wikipedia summary.
- This Wikipedia python library can do a lot more (we are just using it for summary).

Calling the bot ourselves

- Let’s start by calling our bot with a custom keyword.
- Go on /r/test and make a comment….
- Maybe don’t use “uciWikiBot”, pick your own trigger word.

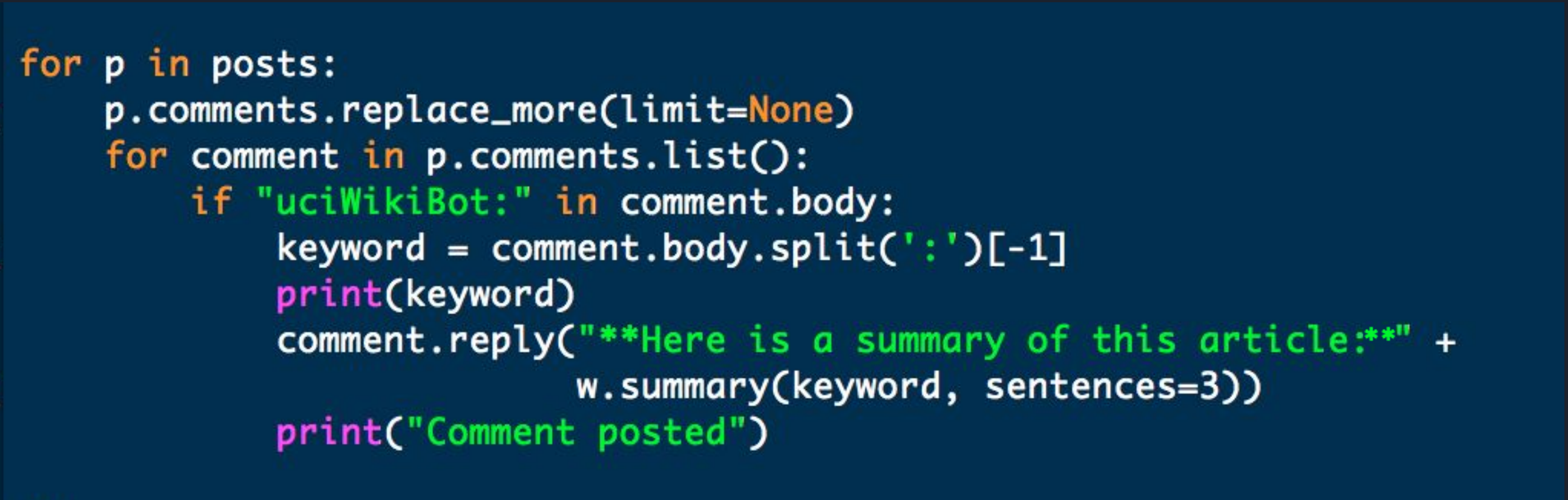


[–] TheZotBot 1 point just now

uciWikiBot: reddit

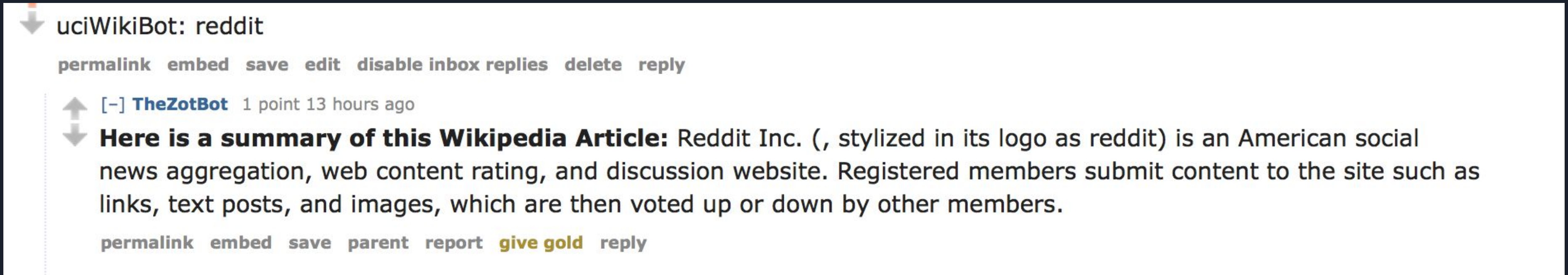
Replying to our comments

- Get the keyword from the body (string manipulation...)
- `comment.reply` with the summary and a message
- Did your bot reply to the comment? Make sure the subreddit and posts you are looking at are within range...



```python
for p in posts:
    p.comments.replace_more(limit=None)
    for comment in p.comments.list():
        if "uciWikiBot:" in comment.body:
            keyword = comment.body.split(':')[-1]
            print(keyword)
            comment.reply("**Here is a summary of this article:**" +
                          w.summary(keyword, sentences=3))
            print("Comment posted")
```

Hopefully, you see something like this



uciWikiBot: reddit

permalink embed save edit disable inbox replies delete reply

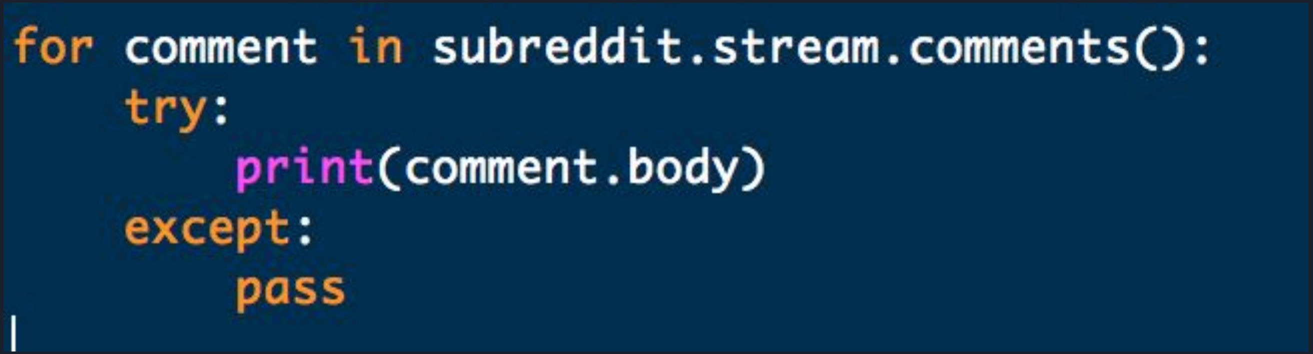
[–] TheZotBot 1 point 13 hours ago

Here is a summary of this Wikipedia Article: Reddit Inc. (, stylized in its logo as reddit) is an American social news aggregation, web content rating, and discussion website. Registered members submit content to the site such as links, text posts, and images, which are then voted up or down by other members.

permalink embed save parent report give gold reply

What about live comments?

- So far, we are looking at past comments.
- We can look at live comments with streams (this is how most active bots work).
- Let’s look at live comments on a subreddit (maybe switch to an active subreddit, like /r/politics)

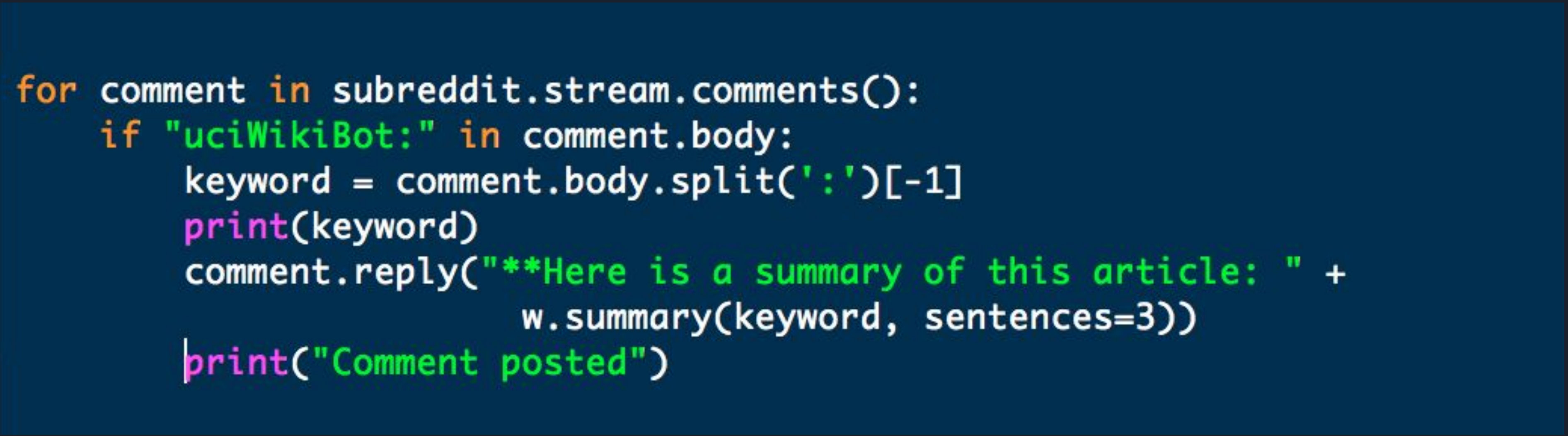


```
for comment in subreddit.stream.comments():
    try:
        print(comment.body)
    except:
        pass
```

A lot of comments...possibility characters will crash the program.

We can respond to our live comments too

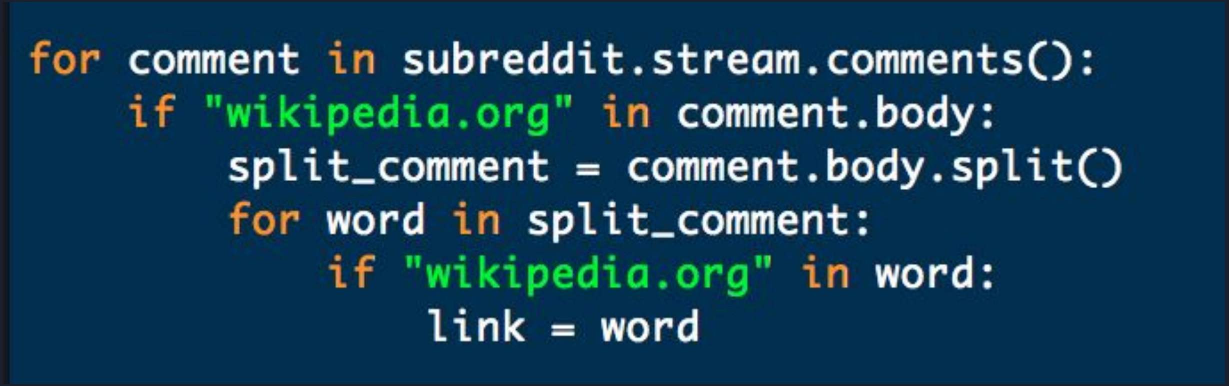
- You can comment out the previous for loop.
- Let’s loop through live comments on a subreddit, and see if we can make a reply.



```
for comment in subreddit.stream.comments():
    if "uciWikiBot:" in comment.body:
        keyword = comment.body.split(':')[-1]
        print(keyword)
        comment.reply("**Here is a summary of this article: " +
                        w.summary(keyword, sentences=3))
        print("Comment posted")
```

A more interesting challenge

- Now that we know how to make a bot that replies to a comment, lets reply to wikipedia.org links.
- Find comments with wikipedia in it, and get the link from that comment.
- (There is probably a better way to accomplish this)



```
for comment in subreddit.stream.comments():
    if "wikipedia.org" in comment.body:
        split_comment = comment.body.split()
        for word in split_comment:
            if "wikipedia.org" in word:
                link = word
```

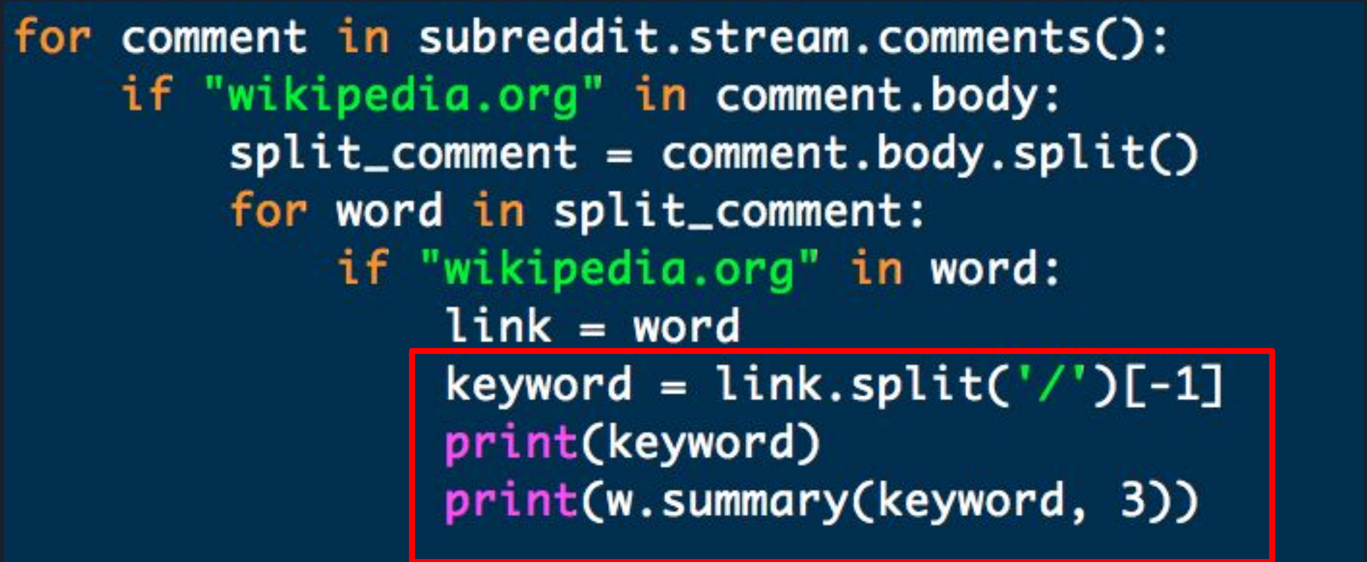
The format of a wikipedia URL

```
https://en.m.wikipedia.org/wiki/Pyramid_scheme

https://en.wikipedia.org/wiki/Cringe_comedy

https://en.wikipedia.org/wiki/Palestinian_tunnel_warfare_in_the_Gaza_Strip

https://en.wikipedia.org/wiki/Aeroflot_Flight_6502

https://en.wikipedia.org/wiki/Aeroflot_Flight_3352
```

Think about how we can get the keywords from one of these URLs.

Formatting wikipedia URL

- *Generally*, a wikipedia link has the format… https://en.wikipedia.org/wiki/KEYWORD
- Splitting by the “/” will (usually) give the keyword as the last item in the list.

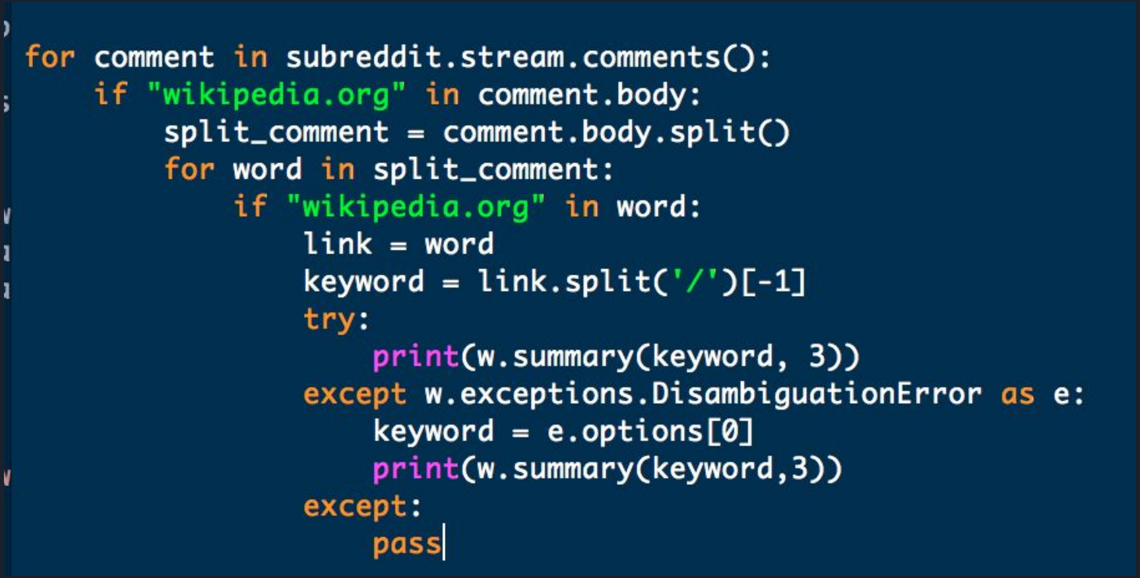


```
for comment in subreddit.stream.comments():
    if "wikipedia.org" in comment.body:
        split_comment = comment.body.split()
        for word in split_comment:
            if "wikipedia.org" in word:
                link = word
                keyword = link.split('/')[-1]
                print(keyword)
                print(w.summary(keyword, 3))
```

This won’t cover every Wikipedia URL. We can keep stripping the URL for a more precise URL, but for the sake of time this should be fine for most cases.

Possible errors

- You may have encountered an error while testing Wikipedia.
- For example, some keywords will have many pages associated with it.



```
for comment in subreddit.stream.comments():
    if "wikipedia.org" in comment.body:
        split_comment = comment.body.split()
        for word in split_comment:
            if "wikipedia.org" in word:
                link = word
                keyword = link.split('/')[-1]
                try:
                    print(w.summary(keyword, 3))
                except w.exceptions.DisambiguationError as e:
                    keyword = e.options[0]
                    print(w.summary(keyword,3))
                except:
                    pass
```

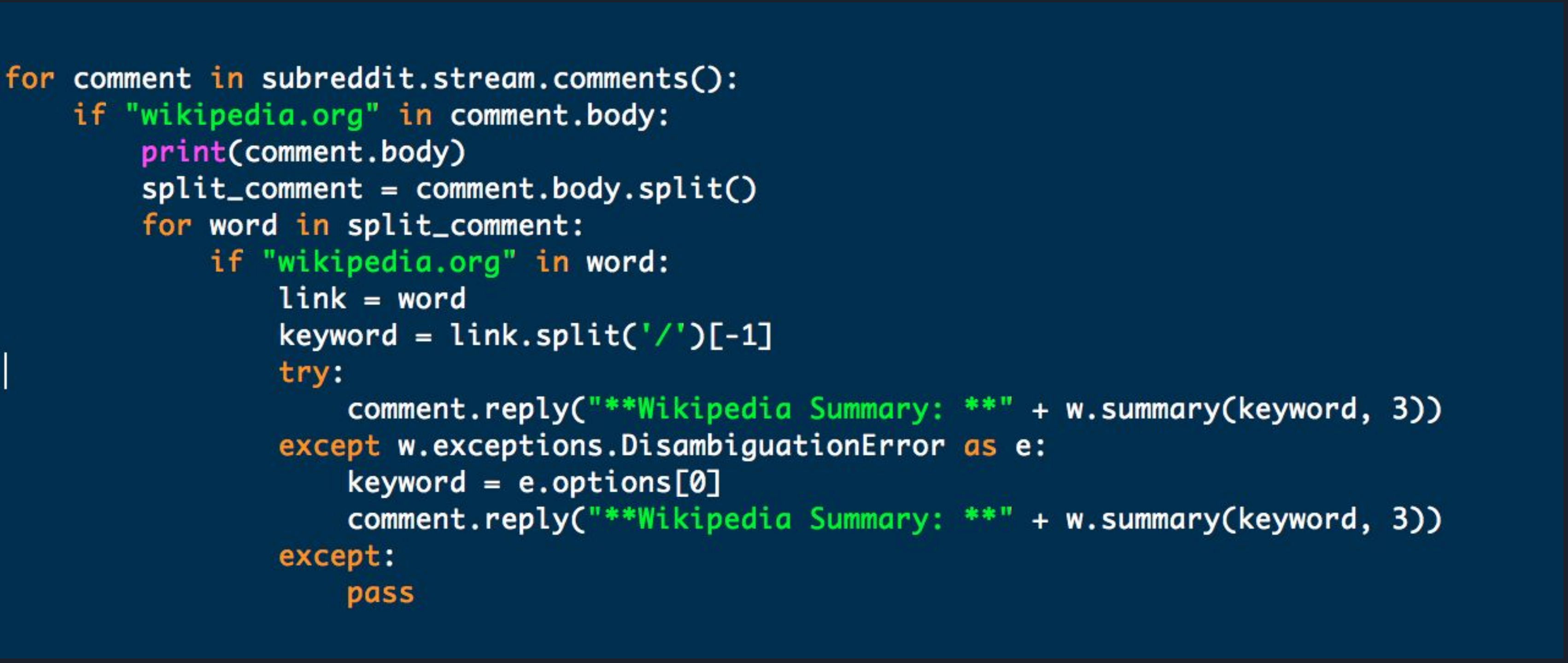
If we have a DisambiguationError, we pick the first article that it could refer to.

If we have some other error, we will ignore it.

Add this to our “uciWikiBot” code too.

Replying to links

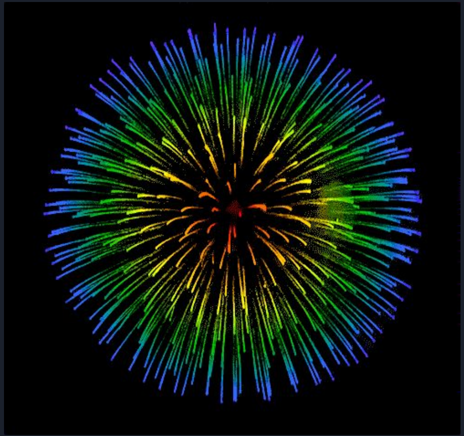
- If you were able to print out Wikipedia summaries, lets try replying to them.



```
for comment in subreddit.stream.comments():
    if "wikipedia.org" in comment.body:
        print(comment.body)
        split_comment = comment.body.split()
        for word in split_comment:
            if "wikipedia.org" in word:
                link = word
                keyword = link.split('/')[-1]
                try:
                    comment.reply("**Wikipedia Summary: **" + w.summary(keyword, 3))
                except w.exceptions.DisambiguationError as e:
                    keyword = e.options[0]
                    comment.reply("**Wikipedia Summary: **" + w.summary(keyword, 3))
                except:
                    pass
```

We did it?

- Now, you should have a basic bot that can scan past and live comments.
- We implemented an outside library to provide Wikipedia information



What’s next?

- That is the extent of our bot’s features.
- Feel free to improve upon your bot, or add any other features you like.
- Keeping your bot open in the background isn’t ideal (consider putting it on a server to run indefinitely…)
- Thanks for coming join ICSSC