

# Cannot print the terminal output in Streamlit?

[Charly Wagnier](#) 1 October 24, 2020, 8:09pm

Hi guys,

I'm trying to get a terminal output to be printed in Streamlit. I tried various codes including this one

```
import searchconsole
account = searchconsole.authenticate(client_config="GSCTatieLouCredentials
st.write(account)
```

but nothing seems to work - See screenshot below:



Any idea on how to print in Streamlit?

Thanks,  
Charly

2 Likes

[okld](#) 2 October 25, 2020, 12:17am

Hello [@Charly Wagnier](#) ,

One solution would be to redirect stdout/stderr to st.write (or anything you want).

Here's a quick example for stdout:

```
from contextlib import contextmanager, redirect_stdout
from io import StringIO
from time import sleep
import streamlit as st

@contextmanager
def st_capture(output_func):
    with StringIO() as stdout, redirect_stdout(stdout):
        old_write = stdout.write
```

```
def new_write(string):
    ret = old_write(string)
    output_func(stdout.getvalue())
    return ret

stdout.write = new_write
yield

output = st.empty()
with st_capture(output.code):
    print("Hello")
    sleep(1)
    print("World")
```

Hello  
World

Goodbye World

4 Likes

Charly Wagnier 3 October 25, 2020, 10:31am

Thank you Synode! 🙏

Trying your code I've had an error, along with the message:

Please report this bug at <https://github.com/streamlit/streamlit/issues>.

See error below:

```
(venv) C:\Users\Charly\Desktop\StreamForecast>streamlit run PrintToTerminal.py
2020-10-25 10:28:54.198 Thread 'MainThread': missing ReportContext
2020-10-25 10:28:54.200 Thread 'MainThread': missing ReportContext
2020-10-25 10:28:54.201 Thread 'MainThread': missing ReportContext
Traceback (most recent call last):
  File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\streamlit\runner.py", line 100, in on_started(self)
  File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\streamlit\runner.py", line 100, in _print_url(server.is_running_hello)
  File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\streamlit\runner.py", line 100, in click.secho("")
  File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\click\click.py", line 100, in return echo(message, file=file, nl=nl, err=err, color=color)
```

```
File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\cli
file.write(message)
File "C:\Users\Charly\Desktop\StreamForecast\PrintToTerminalSynode.py",
  output_func(stdout.getvalue())
File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\stre
  return dg._enqueue("markdown", code_proto) # type: ignore
File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\stre
  _enqueue_message(msg)
File "c:\users\charly\desktop\streamforecast\venv\lib\site-packages\stre
  raise NoSessionContext()
streamlit.errors.NoSessionContext
```

Is this expected?

Thanks,  
Charly

---

[okld](#) 4 October 25, 2020, 11:14am

Hmm, not quite 😊  
Can you share your code?

---

[Charly Wargnier](#) 5 October 25, 2020, 12:01pm

It was literally your code pasted verbatim in a Venv folder! 😊

---

[okld](#) 6 October 25, 2020, 2:29pm

Hmm, indeed. I've implemented that while streamlit was already running. I didn't run it from scratch though.  
I'll fix that a little bit later today

1 Like

---

[Charly Wargnier](#) 7 October 25, 2020, 4:20pm

Thanks Synode. I actually found a workaround for my issue, although I (and others I'm sure :)) would still be  
keen to see the above code working 🙌 😊

Charly

[Charly Wagnier](#) 8 October 26, 2020, 1:08am

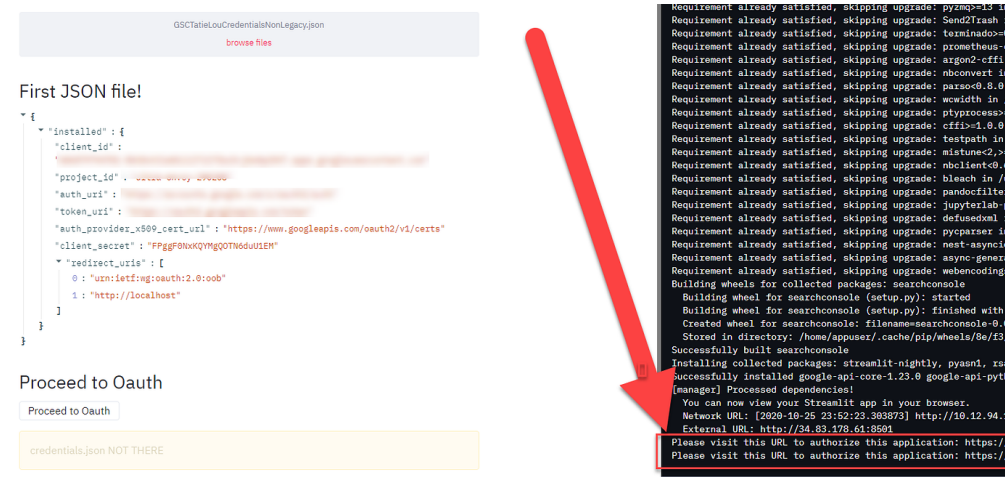
Hi [@okld](#),

It looks like I may have spoken too fast! 😊 The workaround I found earlier doesn't quite work, issue logged here:

### [OSError: \(\[Errno 98\] Address already in use\) in Streamlit Sharing](#)

Hi guys, I'm building an app which tries to authenticate a Google service via OAuth (credentials are in a JSON file). Once the 1st credentials are uploaded, the script is meant to redirect the user to Google's consent screen: [\[image\]](#) Instead, I have this error in Streamlit Sharing: `OSError: [Errno 98] Address already in use` Do you know if there's an explanation for this? It's worth noting that this app works perfectly well on my local machine. Here's the app: <https://share.streamlit.io/...>

I believe that being able to print the console's output to Streamlit would likely fix the issue. As you can see in the screenshot below, users would need to click on the URL in the console for the app to work:



Instead, I either get the Errno 98 issue (issue above) or nothing happens... whereas ideally users should be redirected to Google's consent screen.

Thanks,  
Charly

[okld](#) 9 October 26, 2020, 6:30pm

Alright, new version. Tell me if it works in your case

EDIT: removed output = `st.empty()`. Now you just have to put streamlit's function name as parameter.

```
from contextlib import contextmanager
from io import StringIO
from streamlit.report_thread import REPORT_CONTEXT_ATTR_NAME
```

```
from threading import current_thread
import streamlit as st
import sys
```

```
@contextmanager
```

```
def st_redirect(src, dst):
```

```
    placeholder = st.empty()
```

```
    output_func = getattr(placeholder, dst)
```

```
    with StringIO() as buffer:
```

```
        old_write = src.write
```

```
        def new_write(b):
```

```
            if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
```

```
                buffer.write(b)
```

```
                output_func(buffer.getvalue())
```

```
            else:
```

```
                old_write(b)
```

```
    return
```

```
Prints as st.code()
```

```
Prints as st.info()
```

```
Prints as st.markdown()
```

```
You can print regular success messages
```

```
And you can redirect errors as well
```

3 Likes

---

[In page Console / Print](#)

---

[Charly\\_Wargnier](#) 10 October 26, 2020, 6:32pm

Thanks! Will try now! 😊

---

[Charly\\_Wargnier](#) 11 October 26, 2020, 7:14pm



```
with st_stdout("info"):
    # Inside this block, every function that prints to your terminal
    # will have their output redirected to output.info(), a streamlit func
```

I guess that the function which prints to your terminal is `searchconsole.authenticate()`, right? You'd do something like this:

```
with st_stdout("info"):
    searchconsole.authenticate(client_config="GSCTatieLouCredentials.json")
```

1 Like

okld 13 October 26, 2020, 8:00pm

Small notice, I've edited my two last replies. I've updated the code to avoid using that `output = st.empty()` in your code.

1 Like

Charly Wagnier 14 October 27, 2020, 10:41am

Thanks Synode! It works great! 🙌

I've added your suggested code as the solution.

On a rather cosmetic note, I was wondering whether the displayed text could be “wrapped”?

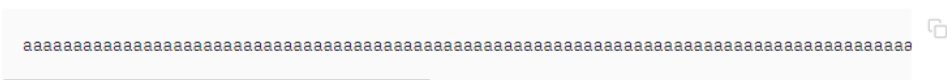
Reason being: The URL that is retrieved from the terminal is very long and far overflows the box's bounds, if that makes sense.

Thanks,  
Charly

okld 15 October 27, 2020, 7:57pm

The only streamlit element that could work in your case is “code” I guess:

```
with st_stdout("code"):
    # Your function
```



At least it won't overflow the box's bounds.

1 Like

---

[Charly Wagnier](#) 17 October 31, 2020, 2:09pm

Nice, thanks Synode!

That would be ideal yet I assume it's not possible to wrap Python variables in a Markdown hyperlink via `st.markdown`?

---

[nicjohnso](#) 18 December 29, 2020, 6:42pm

Hi All,

Trying to fit this proof of concept into my use case, which involves tracking the number of iterations through a function, think:

```
for x in range(10):
```

I'm using with `st_stdout("success")`: to print out x and I'm getting a success box that looks like

```
0 1 2 3 4 5 6 7 8 9
```

Ideally I would just get the current x value, and previous values would be overwritten, I've tried carriage returns and flushing stdout to no avail.

Any thoughts are much appreciated.

---

[komodovaran](#) 19 February 11, 2021, 10:21am

Is there any way to *also* print the captured output to the terminal? Now I only get visual output, but I lose terminal loggers.

I guess you could wrap it in

```
print(stuff)
with st_stdout():
    print(stuff)
```

and wrap it in `print_here_and_there(obj)` or even `print = print_here_and_there` if you so choose, but it seems a bit... I don't know... roundabout?



**Edit:**

I did it anyway, and it's actually not half bad:

```
def print_st(*args: Any) -> None:
    for arg in args:
        print(arg, "\n")
    with st_stdout():
        for arg in args:
            print(arg, "\n")
```

Now Streamlit and the terminal are completely in sync.

1 Like

---

[okld](#) 20 February 11, 2021, 4:56pm

You could also remove that `else` statement from my `new_write()` function:

```
# Before
def new_write(b):
    if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
        buffer.write(b)
        output_func(buffer.getvalue())
    else:
        old_write(b)

# After
def new_write(b):
    if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
        buffer.write(b)
        output_func(buffer.getvalue())
    old_write(b)
```

1 Like

---

[Charly Wagnier](#) 21 February 25, 2021, 1:15pm

Hola Synode!

Is it possible to display countdown timer via your above helpers?

See the code below:

```
import time
import datetime
import os
```

```

_clear_cmd = 'cls' if os.name == 'nt' else 'clear'

def test():
    d0 = datetime.datetime.now()
    d1 = datetime.datetime(2021, 3, 15)
    days_dif = d1 - d0
    seconds = int(days_dif.total_seconds())
    while seconds:
        days = seconds // (24 * 60 * 60)
        hours = (seconds-days * 86400) // 3600
        minutes = (seconds-days * 86400-hours*3600) // 60
        seconds_timer = (seconds-days * 86400-hours*3600-minutes*60)
        timer = (f"{days} days {hours} hours {minutes} minutes and {seconds_t}
        print('test')

        print(timer, end='\r')
        time.sleep(1)
        os.system(_clear_cmd)
        seconds -= 1

```

Thanks,  
Charly

---

[okld](#) 22 February 27, 2021, 3:35pm

Hello [@Charly Wagnier](#) ,

So instead of appending the text, you want it to be overridden.

You could try something like this:

```

# Before
def new_write(b):
    if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
        buffer.write(b)
        output_func(buffer.getvalue())
    else:
        old_write(b)

# After
def new_write(b):
    if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
        buffer.write(b)
        output_func(b) # Changed here
    else:
        old_write(b)

```

1 Like

[Charly Wagnier](#) 23 February 27, 2021, 5:01pm

Thanks Synode! I'll have a proper look asap. 😊

[filipespacheco](#) 24 April 27, 2021, 5:37pm

Did you get a solution?

[BugzTheBunny](#) 25 April 29, 2021, 4:31pm

Hey Filip!

If you could send some kind of notification in a few hours when i will be near the PC, I will provied you with a solution I've managed to create using some of the code here, which works great, not exactly a console, but you are able to see the output if it's comming from 'logging', and you can see it on the screen 😊

Best regards.

1 Like

[filipespacheco](#) 26 May 3, 2021, 6:10pm

```
Administrator: C:\WINDOWS\system32\cmd.exe - streamlit run
with open(file_path, "rb") as f:
PermissionError: [Errno 13] Permission denied:
Stopping...

(tensorflow)

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://204.53.170.192:8501

Average time of function evaluating (secs): 0.09821967124938964

||| 6.0% GA is running...6 gen from 100...best value = 16359.806084983396
||| 10.0% GA is running...10 gen from 100...best value = 15942.6865178398
||| 12.0% GA is running...12 gen from 100...best value = 13628.7813364905
||| 13.0% GA is running...13 gen from 100...best value = 13392.3315775618
||| 28.0% GA is running...28 gen from 100...best value = 13292.0011650483
||| 29.0% GA is running...29 gen from 100...best value = 11343.0370385975
||| 52.0% GA is running...52 gen from 100...best value = 11154.7901802696
||| 65.0% GA is running...65 gen from 100...best value = 10473.8579763065
||| 75.0% GA is running...75 gen from 100...best value = 10473.5202141674
```

This is my problem. I'm utilizing a library that prints this things in my Console output.

I would like to print the same messages in Streamlit page.

Can you help me?

[Ejor](#) 27 May 4, 2021, 5:18pm

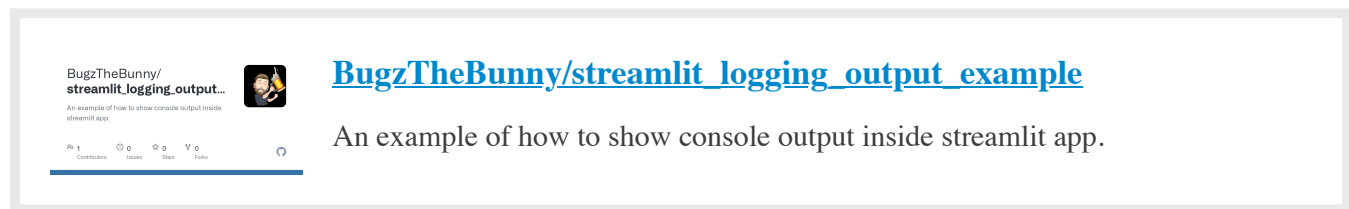
Hello [@BugzTheBunny](#) ! I'm interested if you could provide a solution for the output coming from "logging"  
😊

This code is great, thank you [@okld](#) for your work and [@Charly\\_Wagnier](#) for asking the question 😊

Have a nice day!

[BugzTheBunny](#) 28 May 4, 2021, 8:02pm

Hi guys, ive made a demo of the samples of the code here, you may view it in this repo:



[@Ejor](#) [@filipespacheco](#)

1 Like

[Ejor](#) 29 May 14, 2021, 1:16pm

Thank you so much [@BugzTheBunny](#) , it works like a charm! (And I'm really sorry I haven't seen your message sooner).

1 Like

[BugzTheBunny](#) 30 May 18, 2021, 8:12am

Not a problem, glad it works and can be helpful!

[tomsj](#) 31 May 30, 2021, 4:44pm

the codes is work, but mix the output into one line, so i add "\r\n" in the new\_write(), just FYI.

```
def new_write(b):  
    if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
```

```

        buffer.write(b+"\r\n")
        output_func(buffer.getvalue())
    else:
        old_write(b)

```

1 Like

---

[aptperson](#) 32 October 13, 2021, 12:37am

Is there a way to limit the number of lines that are printed?

I have an app that has a long running process with many steps. Each step prints to the console as part of its progress. I only want to display the last 10 lines of stderr. Is this possible?

Thanks in advance

---

[sis](#) 33 November 19, 2021, 9:35am

If you're using the (excellent) [loguru](#) as logging manager, I manage to redirect console logs to streamlit apps by adding a related sink to a logger.

For example, if you want `st.warning` to catch `logger.warning` (and nothing else) and `st.error` to catch `logger.error` (and nothing else), the following snippet should work

```

from loguru import logger
import streamlit as st

def redirect_loguru_to_streamlit():
    def _filter_warning(record):
        return record["level"].no == logger.level("WARNING").no
    if 'warning_logger' not in st.session_state:
        st.session_state['warning_logger'] = logger.add(st.warning, filter
    if 'error_logger' not in st.session_state:
        st.session_state['error_logger'] = logger.add(st.error, level='ERF

redirect_loguru_to_streamlit()

def main():
    logger.info('This should not be printed in app.')
    logger.warning('This should be printed as `st.warning`.')
    logger.error('This should be printed as `st.error`.')
    if st.button('Rerun'):
        st.write('You should not see duplicated logs.')

```

```
if __name__ == '__main__':
    main()
```

2 Likes

---

[Gennady Matveev](#) 34 December 11, 2021, 3:36pm

Doesn't seem to work. Instead of overwriting, just shows an empty placeholder. Anybody managed to succeed with it?

```
def new_write(b):
    if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
        buffer.write(b)
        output_func(b) # Changed here
```

**Edit**

Solution found:

```
def new_write(b):
    if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
        buffer.write(b)
        sleep(1)
        buffer.seek(0) # returns pointer to 0 position
        output_func(b)
```

---

[srush](#) 35 March 7, 2022, 2:35pm

This workaround broke in recent versions of streamlit, as `REPORT_CONTEXT_ATTR_NAME` was removed. Anyone have a new approach?

2 Likes

---

[gerardrbentley](#) 36 March 16, 2022, 7:42pm

**EDIT:** the counter example works on streamlit 1.7 if you swap out the import and context constant:

```
# from streamlit.report_thread import REPORT_CONTEXT_ATTR_NAME
from streamlit.script_run_context import SCRIPT_RUN_CONTEXT_ATTR_NAME
...
# if getattr(current_thread(), REPORT_CONTEXT_ATTR_NAME, None):
if getattr(current_thread(), SCRIPT_RUN_CONTEXT_ATTR_NAME, None):
```

Got something working I think similar using contextlib. I'll check later if there was extra logs gotten from REPORT\_CONTEXT\_ATTR\_NAME it misses.

Examples:

- `print`
- `df.info()`
- output of success and error subprocess
- traceback of python error

[cloud](#) (py39)

[github](#)

Code:

```
from contextlib import redirect_stdout, redirect_stderr
import io
import sys
import subprocess
import traceback
import streamlit as st
import pandas as pd
st.set_page_config(layout='wide')

st.header("Left: Body, Middle: Std Out, Right: Std Err")
body, stdout, stderr = st.columns(3)

with redirect_stdout(io.StringIO()) as stdout_f, redirect_stderr(io.StringIO()) as stderr_f:
    try:
        print('Hello World!')
        df = pd.DataFrame({"test": [1,2,3]})
        df.info()
        good_process = subprocess.run(["ls", "-lah", "."], capture_output=True)
        stdout_f.write(good_process.stdout)
        stderr_f.write(good_process.stderr)
        bad_process = subprocess.run(["ls", "wtf"], capture_output=True, check=False)
        stdout_f.write(bad_process.stdout)
        stdout_f.write(bad_process.stderr) # Also print in in middle column
        stderr_f.write(bad_process.stderr)
```

Cheers! 🍻

2 Likes

---

[Yashvardhan Prasad](#) 37 August 12, 2022, 11:59am

Is there a way to redirect output image as well?

[Yuji](#) 38 January 2, 2023, 11:48pm

For streamlit 1.16.0, it works with:

```
from streamlit.runtime.scriptrunner.script_run_context import SCRIPT_RUN_CONTEXT_ATTR_NAME
...
if getattr(current_thread(), SCRIPT_RUN_CONTEXT_ATTR_NAME, None):
```

[sgoede](#) 39 July 6, 2023, 8:56am

When I use the code below, the current output is printed as intended:

```
@contextmanager
def st_redirect(src, dst):
    placeholder = st.empty()
    output_func = getattr(placeholder, dst)

    with StringIO() as buffer:
        old_write = src.write

        def new_write(b):
            if getattr(current_thread(), SCRIPT_RUN_CONTEXT_ATTR_NAME, None):
                buffer.write(b)
                output_func(b)
            else:
                old_write(b)

        try:
            src.write = new_write
            yield
        finally:
            src.write = old_write

@contextmanager
def st_stdout(dst):
    "this will show the prints"
```

However, when a user changes the input widget(s) value(s) **whilst the terminal output is printed** the app will crash.

Is there a way to check if the user has changed the initial value, and if so, change the redirect to the correct one (i believe is done in getarr?)



[A L](#) 40 July 29, 2023, 9:27pm

Hello, can you help, I am a begginer with streamlit, how to use this pass in my code. I am running a job, and have an output to terminal and to info.log. How can I print it to my streamlit page?