

---

# **studio-mgr**

***Release 2.0.dev1***

**Brian Skinn**

**Nov 20, 2020**



# CONTENTS

<b>1</b>	<b>User’s Guide</b>	<b>3</b>
<b>2</b>	<b>API</b>	<b>5</b>
<b>3</b>	<b>Inheritance Information</b>	<b>9</b>
<b>4</b>	<b>Indices and tables</b>	<b>13</b>
	<b>Python Module Index</b>	<b>15</b>
	<b>Index</b>	<b>17</b>



*Intro stuff to go here*



**USER'S GUIDE**

*There will be things here...*





*API page.*

*Definitely need to explain here what the public API is.*

*Anticipate having some of the private API presented here, but have to make it clear what's private vs public.*

*stdio\_mgr code module.*

stdio\_mgr provides a context manager for convenient mocking and/or wrapping of stdin/stdout/stderr interactions.

**Author** Brian Skinn ([bskinn@alum.mit.edu](mailto:bskinn@alum.mit.edu))

**File Created** 24 Mar 2018

**Copyright** (c) Brian Skinn 2018-2019

**Source Repository** [http://www.github.com/bskinn/stdio\\_mgr](http://www.github.com/bskinn/stdio_mgr)

**Documentation** See README.rst at the GitHub repository

**License** The MIT License; see [LICENSE.txt](#) for full license terms

## Members

**class** stdio\_mgr.stdio\_mgr.**RandomTextIO**

Class to capture writes to a buffer even when detached.

Subclass of `TextIOWrapper` that utilises an internal buffer defaulting to utf-8 encoding.

As a subclass of `TextIOWrapper`, it is not thread-safe.

All writes are immediately flushed to the buffer.

This class provides `getvalue()` which emulates the behavior of `StringIO.getvalue()`, decoding the buffer using the `encoding`. The value is available even if the stream is detached or closed.

**getvalue()**

Obtain buffer of text sent to the stream.

**write(\*args, \*\*kwargs)**

Flush after each write.

**class** stdio\_mgr.stdio\_mgr.**SafeCloseRandomTextIO**

Class to capture writes to a buffer even when detached, and safely close.

Subclass of `_SafeCloseIOBase` and `RandomTextIO`.

**class** stdio\_mgr.stdio\_mgr.**SafeCloseTeeStdin**(tee, \*args, \*\*kwargs)

Class to tee contents to a side buffer on read, and safely close.

Subclass of `_SafeCloseIOBase` and `TeeStdin`.

**class** stdio\_mgr.stdio\_mgr.**SimulateStdin** (*init\_text=""*, *encoding='utf-8'*)

Class to simulate content appearing on stdin.

Subclass of `TextIOWrapper` that provides `getvalue()` which emulates the behavior of `StringIO.getvalue()`, decoding the buffer using the `encoding`.

This class also provides the method `append()`, which is not available for the base `TextIOWrapper` type. This method adds new content to the end of the stream while leaving the read position unchanged.

As a subclass of `TextIOWrapper`, it is not thread-safe.

Instantiation takes two arguments:

*init\_text*

*str* (*optional*) – Text to use as the initial contents of the stream to be teed. Default is an empty *str*.

*encoding*

*str* (*optional*) – Encoding for the underlying `TextIOWrapper`. Default is “utf-8”.

**append** (*text*)

Write to end of stream while maintaining seek position.

Actually stores the current position; seeks to end; writes *text*; and seeks to prior position.

**Parameters** *text* – *str* – Text to append to the current stream contents.

**getvalue** ()

Obtain pending buffer of text for stdin.

**class** stdio\_mgr.stdio\_mgr.**StdioManager** (*in\_str=""*, *close=True*)

Substitute temporary text buffers for *stdio* in a managed context.

Context manager.

Substitutes empty `RandomTextIO`s for `sys.stdout` and `sys.stderr`, and a `TeeStdin` for `sys.stdin` within the managed context.

Upon exiting the context, the original stream objects are restored within `sys`, and the temporary streams are closed.

**Parameters** *in\_str* – *str* (*optional*) – Initialization text for the `TeeStdin` substitution for *stdin*. Default is an empty string.

**Yields**

- *in\_* – `TeeStdin` – Temporary stream for *stdin*.
- *out\_* – `RandomTextIO` – Temporary stream for *stdout*, initially empty.
- *err\_* – `RandomTextIO` – Temporary stream for *stderr*, initially empty.

**property** *stderr*

Return capturing stderr stream.

**property** *stdin*

Return capturing stdin stream.

**property** *stdout*

Return capturing stdout stream.

**class** stdio\_mgr.stdio\_mgr.**TeeStdin** (*tee*, *\*args*, *\*\*kwargs*)

Class to tee simulated contents to a side buffer on read.

Subclass of `SimulateStdin` and `_Tee` that simulates a stdin stream while tee'ing all content *read* from the stream to *tee*.

To emphasize: teeing occurs on content *read*, **not write**..

As a subclass of `TextIOWrapper`, it is not thread-safe.

**class** `stdio_mgr.stdio_mgr._MultiCloseContextManager` (*iterable=()*, /)

Manage multiple closable members of a tuple.

**class** `stdio_mgr.stdio_mgr._PersistedBytesIO` (*closure\_callback*)

Class to persist the value after close.

A copy of the bytes value is given to a callback prior to the `close()`.

**close()**

Send buffer to callback and close.

**class** `stdio_mgr.stdio_mgr._SafeCloseIOBase`

Class to ignore `ValueError` when exiting the context.

Subclass of `TextIOBase` that disregards `ValueError`, which can occur if the file has already been closed, when exiting the context.

**class** `stdio_mgr.stdio_mgr._Tee` (*tee*, \**args*, \*\**kwargs*)

Class to tee contents to a side buffer on read.

Subclass of `TextIOWrapper` that overrides `read()` and `readline()` to tee all content *read* from the stream to *tee*.

To emphasize: teeing occurs on content *read*, **not write**.

As a subclass of `TextIOWrapper`, it is not thread-safe.

Instantiation takes an additional argument:

*tee*

`TextIOBase` – Text stream to write content of each read

**read** (*size=None*)

Tee text to side buffer when read.

Overrides `TextIOWrapper.read()` to implement the teeing.

**Parameters** *size* – `int` or `None` (*optional*) – Number of characters to return; a negative or `None` value reads to EOF.

**readline** (*size=-1*)

Tee text to side buffer when read.

Overrides `TextIOWrapper.readline()` to implement the teeing.

**Parameters** *size* – `int` (*optional*) – Number of characters to return; a negative value reads an entire line, regardless of length

`stdio_mgr.stdio_mgr.stdio_mgr`

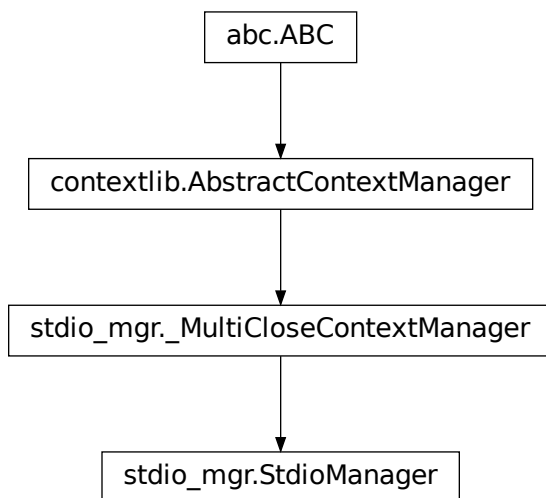
alias of `stdio_mgr.stdio_mgr.StdioManager`



## INHERITANCE INFORMATION

The pages below provide class inheritance diagrams and attribute inheritance information for various classes. This information is mainly intended to facilitate development either on or with `stdio_mgr`.

### 3.1 StdioManager



```
$ python attr_origins.py stdio_mgr StdioManager
__abstractmethods__  :: frozenset()
__add__              :: <slot wrapper '__add__' of 'tuple' objects>
__class__            :: <class 'abc.ABCMeta'>
__contains__         :: <slot wrapper '__contains__' of 'tuple' objects>
__delattr__          :: <slot wrapper '__delattr__' of 'object' objects>
__dir__              :: <method '__dir__' of 'object' objects>
__enter__            :: <function StdioManager.__enter__>
__eq__               :: <slot wrapper '__eq__' of 'tuple' objects>
__exit__             :: <function StdioManager.__exit__>
__format__           :: <method '__format__' of 'object' objects>
```

(continues on next page)

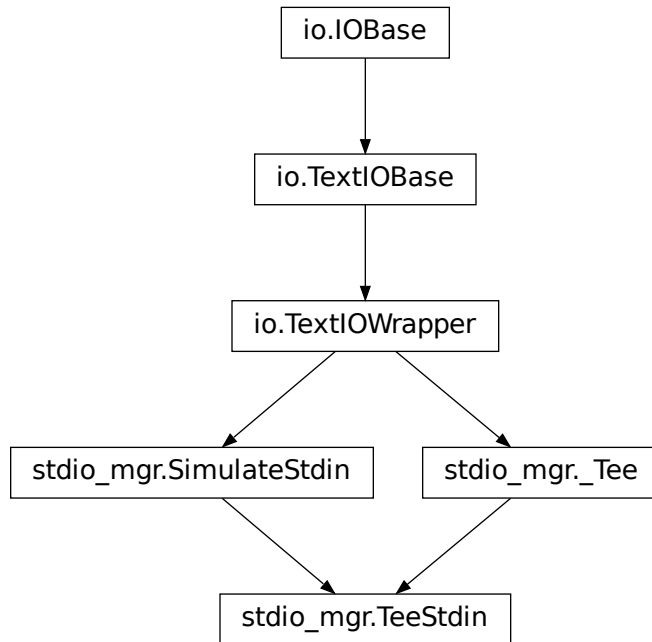
(continued from previous page)

```

__ge__          :: <slot wrapper '__ge__' of 'tuple' objects>
__getattr__     :: <slot wrapper '__getattr__' of 'tuple' objects>
__getitem__     :: <slot wrapper '__getitem__' of 'tuple' objects>
__getnewargs__  :: <method '__getnewargs__' of 'tuple' objects>
__gt__          :: <slot wrapper '__gt__' of 'tuple' objects>
__hash__        :: <slot wrapper '__hash__' of 'tuple' objects>
__init__        :: <slot wrapper '__init__' of 'object' objects>
__init_subclass__ :: <built-in method __init_subclass__ of ABCMeta object>
__iter__        :: <slot wrapper '__iter__' of 'tuple' objects>
__le__          :: <slot wrapper '__le__' of 'tuple' objects>
__len__         :: <slot wrapper '__len__' of 'tuple' objects>
__lt__          :: <slot wrapper '__lt__' of 'tuple' objects>
__module__      :: 'stdio_mgr.stdio_mgr'
__mul__         :: <slot wrapper '__mul__' of 'tuple' objects>
__ne__          :: <slot wrapper '__ne__' of 'tuple' objects>
__new__         :: <function StdioManager.__new__>
__reduce__      :: <method '__reduce__' of 'object' objects>
__reduce_ex__   :: <method '__reduce_ex__' of 'object' objects>
__repr__        :: <slot wrapper '__repr__' of 'tuple' objects>
__rmul__        :: <slot wrapper '__rmul__' of 'tuple' objects>
__setattr__     :: <slot wrapper '__setattr__' of 'object' objects>
__sizeof__      :: <method '__sizeof__' of 'object' objects>
__slots__       :: ()
__str__         :: <slot wrapper '__str__' of 'object' objects>
__subclasshook__ :: <bound method AbstractContextManager.__subclasshook__ of
↳<class 'stdio_mgr.stdio_mgr.StdioManager'>>
__weakref__     :: <attribute '__weakref__' of 'AbstractContextManager' objects>
_abc_impl       :: <_abc_data object>
count           :: <method 'count' of 'tuple' objects>
index           :: <method 'index' of 'tuple' objects>
stderr          :: <property object of 'StdioManager' object>
stdin           :: <property object of 'StdioManager' object>
stdout          :: <property object of 'StdioManager' object>

```

## 3.2 TeeStdin



```

$ python attr_origins.py stdio_mgr.stdio_mgr TeeStdin
_CHUNK_SIZE      :: <attribute '_CHUNK_SIZE' of '_io.TextIOWrapper' objects>
__class__        :: <class 'type'>
__del__          :: <slot wrapper '__del__' of '_io._IOBase' objects>
__delattr__      :: <slot wrapper '__delattr__' of 'object' objects>
__dir__          :: <method '__dir__' of 'object' objects>
__enter__        :: <method '__enter__' of '_io._IOBase' objects>
__eq__           :: <slot wrapper '__eq__' of 'object' objects>
__exit__         :: <method '__exit__' of '_io._IOBase' objects>
__format__       :: <method '__format__' of 'object' objects>
__ge__           :: <slot wrapper '__ge__' of 'object' objects>
__getattr__      :: <slot wrapper '__getattr__' of 'object' objects>
__getstate__     :: <method '__getstate__' of '_io.TextIOWrapper' objects>
__gt__           :: <slot wrapper '__gt__' of 'object' objects>
__hash__         :: <slot wrapper '__hash__' of 'object' objects>
__init__         :: <function _Tee.__init__>
__init_subclass__:: <built-in method __init_subclass__ of type object>
__iter__         :: <slot wrapper '__iter__' of '_io._IOBase' objects>
__le__           :: <slot wrapper '__le__' of 'object' objects>
__lt__           :: <slot wrapper '__lt__' of 'object' objects>
__module__       :: 'stdio_mgr.stdio_mgr'
__ne__           :: <slot wrapper '__ne__' of 'object' objects>
__new__          :: <built-in method __new__ of type object>
__next__         :: <slot wrapper '__next__' of '_io.TextIOWrapper' objects>
__reduce__       :: <method '__reduce__' of 'object' objects>

```

(continues on next page)

(continued from previous page)

```

__reduce_ex__      :: <method '__reduce_ex__' of 'object' objects>
__repr__           :: <slot wrapper '__repr__' of '_io.TextIOWrapper' objects>
__setattr__        :: <slot wrapper '__setattr__' of 'object' objects>
__sizeof__         :: <method '__sizeof__' of 'object' objects>
__str__            :: <slot wrapper '__str__' of 'object' objects>
__subclasshook__   :: <built-in method __subclasshook__ of type object>
_checkClosed       :: <method '_checkClosed' of '_io._IOBase' objects>
_checkReadable     :: <method '_checkReadable' of '_io._IOBase' objects>
_checkSeekable     :: <method '_checkSeekable' of '_io._IOBase' objects>
_checkWritable     :: <method '_checkWritable' of '_io._IOBase' objects>
_finalizing        :: <member '_finalizing' of '_io.TextIOWrapper' objects>
append             :: <function SimulateStdin.append>
buffer             :: <member 'buffer' of '_io.TextIOWrapper' objects>
close              :: <method 'close' of '_io.TextIOWrapper' objects>
closed             :: <attribute 'closed' of '_io.TextIOWrapper' objects>
detach             :: <method 'detach' of '_io.TextIOWrapper' objects>
encoding           :: <member 'encoding' of '_io.TextIOWrapper' objects>
errors             :: <attribute 'errors' of '_io.TextIOWrapper' objects>
fileno             :: <method 'fileno' of '_io.TextIOWrapper' objects>
flush              :: <method 'flush' of '_io.TextIOWrapper' objects>
getvalue           :: <function SimulateStdin.getvalue>
isatty            :: <method 'isatty' of '_io.TextIOWrapper' objects>
line_buffering     :: <member 'line_buffering' of '_io.TextIOWrapper' objects>
name               :: <attribute 'name' of '_io.TextIOWrapper' objects>
newlines           :: <attribute 'newlines' of '_io.TextIOWrapper' objects>
read               :: <function _Tee.read>
readable           :: <method 'readable' of '_io.TextIOWrapper' objects>
readline           :: <function _Tee.readline>
readlines          :: <method 'readlines' of '_io._IOBase' objects>
reconfigure        :: <method 'reconfigure' of '_io.TextIOWrapper' objects>
seek               :: <method 'seek' of '_io.TextIOWrapper' objects>
seekable           :: <method 'seekable' of '_io.TextIOWrapper' objects>
tell               :: <method 'tell' of '_io.TextIOWrapper' objects>
truncate           :: <method 'truncate' of '_io.TextIOWrapper' objects>
writable           :: <method 'writable' of '_io.TextIOWrapper' objects>
write              :: <method 'write' of '_io.TextIOWrapper' objects>
write_through      :: <member 'write_through' of '_io.TextIOWrapper' objects>
writelines         :: <method 'writelines' of '_io._IOBase' objects>

```



## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`



## PYTHON MODULE INDEX

### S

`stdio_mgr.stdio_mgr`, 5



## Symbols

`_MultiCloseContextManager` (class in `stdio_mgr.stdio_mgr`), 7  
`_PersistedBytesIO` (class in `stdio_mgr.stdio_mgr`), 7  
`_SafeCloseIOBase` (class in `stdio_mgr.stdio_mgr`), 7  
`_Tee` (class in `stdio_mgr.stdio_mgr`), 7

## A

`append()` (`stdio_mgr.stdio_mgr.SimulateStdin` method), 6

## C

`close()` (`stdio_mgr.stdio_mgr._PersistedBytesIO` method), 7

## G

`getvalue()` (`stdio_mgr.stdio_mgr.RandomTextIO` method), 5  
`getvalue()` (`stdio_mgr.stdio_mgr.SimulateStdin` method), 6

## M

module  
`stdio_mgr.stdio_mgr`, 5

## R

`RandomTextIO` (class in `stdio_mgr.stdio_mgr`), 5  
`read()` (`stdio_mgr.stdio_mgr._Tee` method), 7  
`readline()` (`stdio_mgr.stdio_mgr._Tee` method), 7

## S

`SafeCloseRandomTextIO` (class in `stdio_mgr.stdio_mgr`), 5  
`SafeCloseTeeStdin` (class in `stdio_mgr.stdio_mgr`), 5  
`SimulateStdin` (class in `stdio_mgr.stdio_mgr`), 5  
`stderr()` (`stdio_mgr.stdio_mgr.StdioManager` property), 6  
`stdin()` (`stdio_mgr.stdio_mgr.StdioManager` property), 6

`stdio_mgr` (in module `stdio_mgr.stdio_mgr`), 7  
`stdio_mgr.stdio_mgr`  
module, 5  
`StdioManager` (class in `stdio_mgr.stdio_mgr`), 6  
`stdout()` (`stdio_mgr.stdio_mgr.StdioManager` property), 6

## T

`TeeStdin` (class in `stdio_mgr.stdio_mgr`), 6

## W

`write()` (`stdio_mgr.stdio_mgr.RandomTextIO` method), 5