

ExitStack

ExitStack is a context manager that will allow you to easily programmatically combine other context managers and cleanup functions. It sounds kind of confusing at first, so let's take a look at an example from the Python documentation to help us understand this idea a bit better:

```
from contextlib import ExitStack
with ExitStack() as stack:
    file_objects = [stack.enter_context(open(filename))
                    for filename in filenames]
```



This code basically creates a series of context managers inside the list comprehension. The **ExitStack** maintains a stack of registered callbacks that it will call in reverse order when the instance is closed, which happens when we exit the bottom of the **with** statement.

There are a bunch of neat examples in the Python documentation for **contextlib** where you can learn about topics like the following:

- Catching exceptions from `__enter__` methods
- Supports a variable number of context managers
- Replacing any use of try-finally
- and much more!

I highly recommend checking it out so you get a good feel for how powerful this class is.