

Why use context managers?

ThreadPoolExecutor – Bad version!

```
from concurrent.futures import ThreadPoolExecutor

# Bad!!!
pool = ThreadPoolExecutor()
for k, v in data.items():
    pool.submit(make_a_sentence, k, v)
# Wait on the results and do something with them.
pool.shutdown()
```

[<link to slides>](#)

Why use context managers?

ThreadPoolExecutor – Good version!

```
from concurrent.futures import ThreadPoolExecutor

# Good, safe, context managed!
with ThreadPoolExecutor() as pool:
    for k, v in data.items():
        pool.submit(make_a_sentence, k, v)
    # Wait on the results and do something with them.
```

[<link to slides>](#)