

Python Concurrency Learning Paths

Python concurrency is a big topic and can be challenging for beginners.

This page provides you with step-by-step learning paths that you can use to get started and get good at Python concurrency.

You might want to bookmark this page and come back to it again and again.

Table of Contents

1. Choose a Concurrency API
2. Threading Learning Path
3. Multiprocessing Learning Path
4. Asyncio Learning Path
5. ThreadPoolExecutor Learning Path
6. ProcessPoolExecutor Learning Path
7. ThreadPool Learning Path
8. Multiprocessing Pool Learning Path
9. Concurrent File I/O Learning Path
10. Guides
11. Need More Help?

Choose a Concurrency API

The first step is to choose a concurrency API.

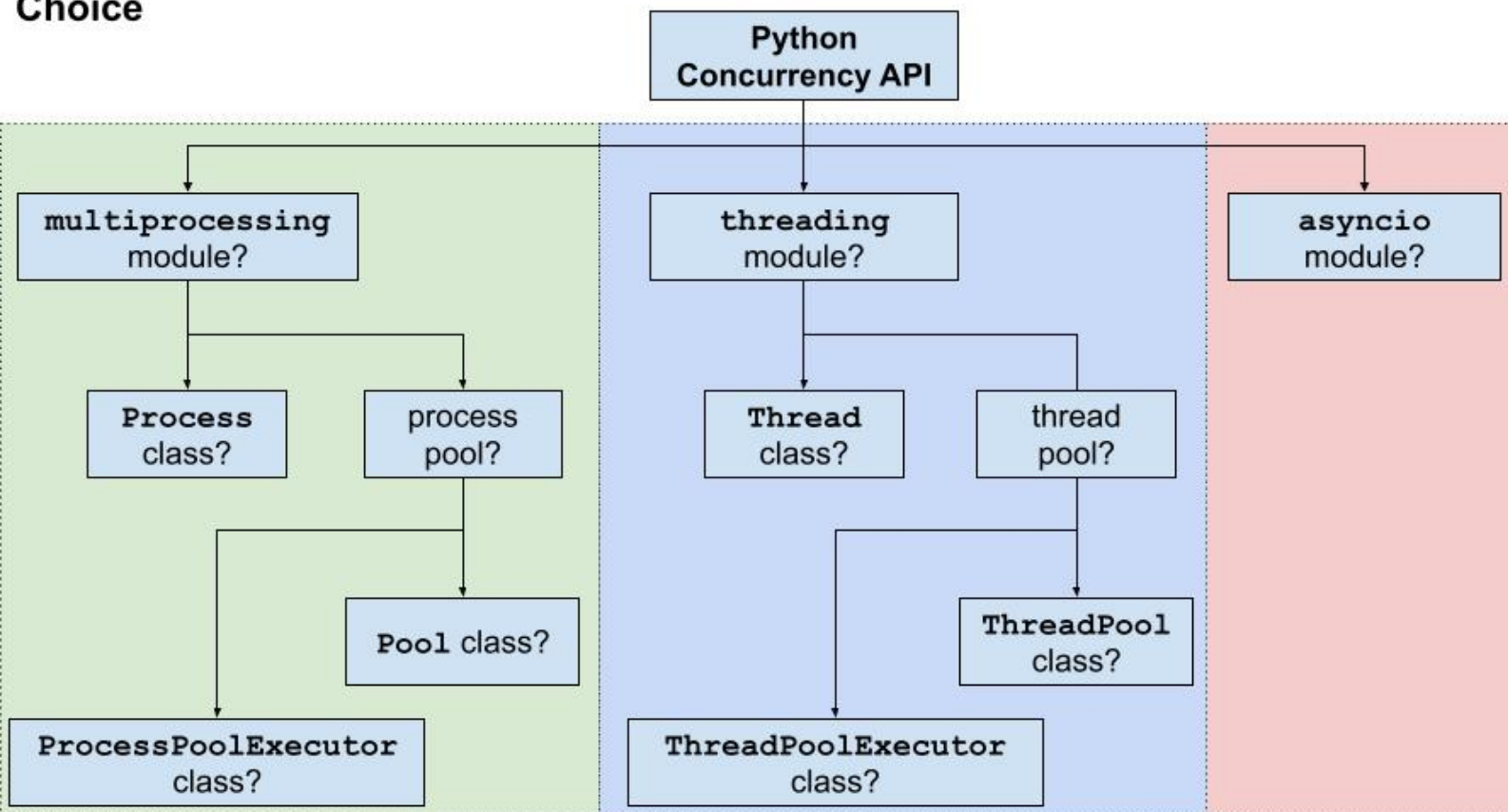
This depends on the specific details of the tasks you wish to execute.

Use a simple step-by-step procedure to help you choose the right Python concurrency API for your project:

- [How to Choose a Python Concurrency API](https://superfastpython.com/python-concurrency-choose-api/) (<https://superfastpython.com/python-concurrency-choose-api/>)

The diagram below will help to summarise the decisions you need to make when choosing a concurrency API.

Python Concurrency API Choice



SuperFastPython.com

(<https://superfastpython.com/python-concurrency-choose-api/>)

PYTHON CONCURRENCY API CHOICE

Threading Learning Path

Python provides thread-based concurrency in the “**threading**” module.

This section provides a learning path that you can use to get started and get effective with the **threading** module, super fast.

- **Step 1: Discover what a thread is in Python.**
 - [What is a Thread in Python \(https://superfastpython.com/what-is-a-thread-in-python/\)](https://superfastpython.com/what-is-a-thread-in-python/)
 - [Thread Life-Cycle in Python \(https://superfastpython.com/thread-life-cycle-in-python/\)](https://superfastpython.com/thread-life-cycle-in-python/)
- **Step 2: Discover how to run code in a new thread.**
 - [How to Run a Function in a New Thread in Python \(https://superfastpython.com/run-function-in-new-thread/\)](https://superfastpython.com/run-function-in-new-thread/)
 - [How to Extend the Thread Class in Python \(https://superfastpython.com/extend-thread-class/\)](https://superfastpython.com/extend-thread-class/)
- **Step 3: Discover how to get the most out of threads.**
 - [Threading in Python: The Complete Guide \(https://superfastpython.com/threading-in-python/\)](https://superfastpython.com/threading-in-python/)
 - [Python Threading Jump-Start \(https://superfastpython.gumroad.com/l/ptj\)](https://superfastpython.gumroad.com/l/ptj) (**my book**)

Next, you can [browse all tutorials \(https://superfastpython.com/category/threading/\)](https://superfastpython.com/category/threading/) or dive into a topic below to learn more.

Configure Threads

- [How to Configure Threads \(https://superfastpython.com/how-to-configure-threads-in-python/\)](https://superfastpython.com/how-to-configure-threads-in-python/)
- [How to Change the Thread Name \(https://superfastpython.com/thread-name/\)](https://superfastpython.com/thread-name/)
- [How to Use Daemon Threads \(https://superfastpython.com/daemon-threads-in-python/\)](https://superfastpython.com/daemon-threads-in-python/)
- [Handle Unexpected Exceptions \(https://superfastpython.com/thread-exception-handling/\)](https://superfastpython.com/thread-exception-handling/)

Basic Threading

- [Thread Blocking Call \(https://superfastpython.com/thread-blocking-call-in-python/\)](https://superfastpython.com/thread-blocking-call-in-python/)
- [What is the Main Thread \(https://superfastpython.com/main-thread/\)](https://superfastpython.com/main-thread/)
- [How to Query Thread Attributes \(https://superfastpython.com/thread-query-attributes/\)](https://superfastpython.com/thread-query-attributes/)
- [Thread Utility Functions \(https://superfastpython.com/thread-utilities/\)](https://superfastpython.com/thread-utilities/)
- [How to Use Thread-Local Data \(https://superfastpython.com/thread-local-data/\)](https://superfastpython.com/thread-local-data/)
- [How to Join a Thread \(https://superfastpython.com/join-a-thread-in-python/\)](https://superfastpython.com/join-a-thread-in-python/)
- [How to Restart a Thread \(https://superfastpython.com/restart-a-thread-in-python/\)](https://superfastpython.com/restart-a-thread-in-python/)
- [Wait for a Result from a Thread \(https://superfastpython.com/thread-wait-for-result/\)](https://superfastpython.com/thread-wait-for-result/)

Intermediate

- [Share Variables Between Threads \(https://superfastpython.com/thread-share-variables/\)](https://superfastpython.com/thread-share-variables/)
- [Threading Return Values \(https://superfastpython.com/thread-return-values/\)](https://superfastpython.com/thread-return-values/)
- [How to Stop a Thread \(https://superfastpython.com/stop-a-thread-in-python/\)](https://superfastpython.com/stop-a-thread-in-python/)
- [How to Close a Thread \(https://superfastpython.com/thread-close/\)](https://superfastpython.com/thread-close/)
- [How to Kill a Thread \(https://superfastpython.com/kill-a-thread-in-python/\)](https://superfastpython.com/kill-a-thread-in-python/)
- [Auto-Start Threads \(https://superfastpython.com/auto-start-threads-in-python/\)](https://superfastpython.com/auto-start-threads-in-python/)
- [How to Sleep a Thread \(https://superfastpython.com/thread-sleep-in-python/\)](https://superfastpython.com/thread-sleep-in-python/)

Advanced Threading

- [5 Threading Anti-Patterns \(https://superfastpython.com/thread-anti-patterns/\)](https://superfastpython.com/thread-anti-patterns/)
- [Interrupt the Main Thread \(https://superfastpython.com/interrupt-the-main-thread-in-python/\)](https://superfastpython.com/interrupt-the-main-thread-in-python/)
- [Thread Context Variables \(https://superfastpython.com/thread-context-variables-in-python/\)](https://superfastpython.com/thread-context-variables-in-python/)
- [Thread Stack Size \(https://superfastpython.com/thread-stack-size-in-python/\)](https://superfastpython.com/thread-stack-size-in-python/)
- [Context Switch Interval \(https://superfastpython.com/context-switch-interval-in-python/\)](https://superfastpython.com/context-switch-interval-in-python/)

Thread-Safe Queues

- [Thread-Safe Queue \(https://superfastpython.com/thread-queue/\)](https://superfastpython.com/thread-queue/)
- [Thread-Safe SimpleQueue \(https://superfastpython.com/thread-safe-simplequeue-in-python/\)](https://superfastpython.com/thread-safe-simplequeue-in-python/)
- [Thread-Safe LifoQueue \(https://superfastpython.com/thread-lifoqueue/\)](https://superfastpython.com/thread-lifoqueue/)
- [Thread-Safe Priority Queue \(https://superfastpython.com/thread-priority-queue/\)](https://superfastpython.com/thread-priority-queue/)
- [Queue task_done\(\) and join\(\). \(https://superfastpython.com/thread-queue-task-done-join/\)](https://superfastpython.com/thread-queue-task-done-join/)

Background Threads

- [Stop a Daemon Thread Gracefully \(https://superfastpython.com/stop-daemon-thread/\)](https://superfastpython.com/stop-daemon-thread/)
- [Watchdog Thread \(https://superfastpython.com/watchdog-thread-in-python/\)](https://superfastpython.com/watchdog-thread-in-python/)
- [Long-Running Background Task \(https://superfastpython.com/thread-long-running-background-task/\)](https://superfastpython.com/thread-long-running-background-task/)

- [Triggered Background Task \(https://superfastpython.com/thread-triggered-background-task/\)](https://superfastpython.com/thread-triggered-background-task/)
- [Periodic Background Task \(https://superfastpython.com/thread-periodic-background/\)](https://superfastpython.com/thread-periodic-background/)

Thread Safety

- [Thread Atomic Operations \(https://superfastpython.com/thread-atomic-operations/\)](https://superfastpython.com/thread-atomic-operations/)
- [Volatile Variables \(https://superfastpython.com/thread-volatile-variables-in-python/\)](https://superfastpython.com/thread-volatile-variables-in-python/)
- [Thread-Safe Counter \(https://superfastpython.com/thread-safe-counter-in-python/\)](https://superfastpython.com/thread-safe-counter-in-python/)
- [Thread-Safe Print \(https://superfastpython.com/thread-safe-print-in-python/\)](https://superfastpython.com/thread-safe-print-in-python/)
- [Thread-Safe Logging \(https://superfastpython.com/thread-safe-logging-in-python/\)](https://superfastpython.com/thread-safe-logging-in-python/)
- [Thread-Safe Dictionary \(https://superfastpython.com/thread-safe-dictionary-in-python/\)](https://superfastpython.com/thread-safe-dictionary-in-python/)
- [Thread-Safe List \(https://superfastpython.com/thread-safe-list/\)](https://superfastpython.com/thread-safe-list/)
- [Thread-Safe Write to File \(https://superfastpython.com/thread-safe-write-to-file-in-python/\)](https://superfastpython.com/thread-safe-write-to-file-in-python/)
- [Thread-Safety Context Managers \(https://superfastpython.com/thread-context-manager/\)](https://superfastpython.com/thread-context-manager/)
- [Thread-Safe Random Numbers \(https://superfastpython.com/random-thread-safe/\)](https://superfastpython.com/random-thread-safe/)

Failure Modes

- [How to Identify a Deadlock \(https://superfastpython.com/thread-deadlock-in-python/\)](https://superfastpython.com/thread-deadlock-in-python/)
- [Thread Livelocks in Python \(https://superfastpython.com/thread-livelock-in-python/\)](https://superfastpython.com/thread-livelock-in-python/)
- [Race Condition With Timing \(https://superfastpython.com/thread-race-condition-timing/\)](https://superfastpython.com/thread-race-condition-timing/)
- [Race Condition Shared Variable \(https://superfastpython.com/thread-race-condition-shared-variable/\)](https://superfastpython.com/thread-race-condition-shared-variable/)
- [Thread Starvation in Python \(https://superfastpython.com/thread-starvation-in-python/\)](https://superfastpython.com/thread-starvation-in-python/)

Sync Primitives

- [Threading Mutex Lock \(https://superfastpython.com/thread-mutex-lock/\)](https://superfastpython.com/thread-mutex-lock/)
- [Threading RLock \(https://superfastpython.com/thread-reentrant-lock/\)](https://superfastpython.com/thread-reentrant-lock/)
- [Threading Condition Variable \(https://superfastpython.com/thread-condition/\)](https://superfastpython.com/thread-condition/)
- [Threading Semaphore \(https://superfastpython.com/thread-semaphore/\)](https://superfastpython.com/thread-semaphore/)
- [Threading Event Object \(https://superfastpython.com/thread-event-object-in-python/\)](https://superfastpython.com/thread-event-object-in-python/)

- [Threading Timer Thread \(https://superfastpython.com/timer-thread-in-python/\)](https://superfastpython.com/timer-thread-in-python/)
- [Threading Barrier \(https://superfastpython.com/thread-barrier-in-python/\)](https://superfastpython.com/thread-barrier-in-python/)
- [Lock vs Semaphore \(https://superfastpython.com/lock-vs-semaphore-in-python/\)](https://superfastpython.com/lock-vs-semaphore-in-python/)
- [Advanced Semaphore Examples \(https://superfastpython.com/thread-semaphore-advanced/\)](https://superfastpython.com/thread-semaphore-advanced/)

Intermediate Locking

- [How To Lock An Object \(https://superfastpython.com/lock-an-object-in-python/\)](https://superfastpython.com/lock-an-object-in-python/)
- [How To Lock A Class \(https://superfastpython.com/lock-a-class-in-python/\)](https://superfastpython.com/lock-a-class-in-python/)
- [How to Lock a Function \(https://superfastpython.com/lock-a-function-in-python/\)](https://superfastpython.com/lock-a-function-in-python/)
- [How to Lock a Variable \(https://superfastpython.com/lock-variable-in-python/\)](https://superfastpython.com/lock-variable-in-python/)

Advanced Locking

- [Lock Contention \(https://superfastpython.com/lock-contention-in-python/\)](https://superfastpython.com/lock-contention-in-python/)
- [Lock Striping in Python \(https://superfastpython.com/lock-striping-in-python/\)](https://superfastpython.com/lock-striping-in-python/)
- [Lock Splitting in Python \(https://superfastpython.com/lock-splitting-in-python/\)](https://superfastpython.com/lock-splitting-in-python/)
- [Lock Retry with Back-Off \(https://superfastpython.com/lock-retry-with-back-off-in-python/\)](https://superfastpython.com/lock-retry-with-back-off-in-python/)
- [Thread Lock Overhead \(https://superfastpython.com/lock-overhead-in-python/\)](https://superfastpython.com/lock-overhead-in-python/)
- [How to Use a Spinlock \(https://superfastpython.com/thread-spinlock-in-python/\)](https://superfastpython.com/thread-spinlock-in-python/)
- [How to Use Busy Waiting \(https://superfastpython.com/thread-busy-waiting-in-python/\)](https://superfastpython.com/thread-busy-waiting-in-python/)

Thread Patterns

- [Thread Pipeline Pattern \(https://superfastpython.com/thread-pipeline/\)](https://superfastpython.com/thread-pipeline/)
- [Producer-Consumer Pattern \(https://superfastpython.com/thread-producer-consumer-pattern-in-python/\)](https://superfastpython.com/thread-producer-consumer-pattern-in-python/)
- [Thread Countdown Latch \(https://superfastpython.com/thread-countdown-latch/\)](https://superfastpython.com/thread-countdown-latch/)

Multiprocessing Learning Path

Python provides process-based concurrency in the “**multiprocessing**” module.

This section provides a learning path that you can use to get started and get effective with the **multiprocessing** module, super fast.

- **Step 1: Discover what a process is in Python.**

- [Process Life-Cycle in Python](https://superfastpython.com/process-life-cycle/) (<https://superfastpython.com/process-life-cycle/>)
- [Thread vs Process in Python](https://superfastpython.com/thread-vs-process/) (<https://superfastpython.com/thread-vs-process/>)

- **Step 2: Discover how to run code in a new child process.**

- [Run a Function in a Child Process](https://superfastpython.com/run-function-in-new-process/) (<https://superfastpython.com/run-function-in-new-process/>)
- [How to Extend the Process Class in Python](https://superfastpython.com/extend-process-class/) (<https://superfastpython.com/extend-process-class/>)

- **Step 3: Discover how to get the most out of processes.**

- [Multiprocessing in Python: The Complete Guide](https://superfastpython.com/multiprocessing-in-python/) (<https://superfastpython.com/multiprocessing-in-python/>)
- [Python Multiprocessing Jump-Start](https://superfastpython.com/products-pmj/) (<https://superfastpython.com/products-pmj/>) (**my book**)

Next, you can [browse all tutorials](https://superfastpython.com/category/multiprocessing/) (<https://superfastpython.com/category/multiprocessing/>) or dive into a topic below to learn more.

Configure Processes

- [Change the Process Name](https://superfastpython.com/process-name/) (<https://superfastpython.com/process-name/>)
- [Daemon Process in Python](https://superfastpython.com/daemon-process-in-python/) (<https://superfastpython.com/daemon-process-in-python/>)
- [Multiprocessing Start Methods](https://superfastpython.com/multiprocessing-start-method/) (<https://superfastpython.com/multiprocessing-start-method/>)
- [Configure Child Process](https://superfastpython.com/configure-child-process-in-python/) (<https://superfastpython.com/configure-child-process-in-python/>)

Basic Processes

- [Main Process in Python](https://superfastpython.com/main-process-in-python/) (<https://superfastpython.com/main-process-in-python/>)
- [Main Thread of Process](https://superfastpython.com/multiprocessing-main-thread/) (<https://superfastpython.com/multiprocessing-main-thread/>)
- [Get Process By Name](https://superfastpython.com/get-process-by-name-in-python/) (<https://superfastpython.com/get-process-by-name-in-python/>)
- [How to Get the Process PID](https://superfastpython.com/multiprocessing-get-pid/) (<https://superfastpython.com/multiprocessing-get-pid/>)

- [Get the Number of CPUs \(https://superfastpython.com/number-of-cpus-python/\)](https://superfastpython.com/number-of-cpus-python/)
- [Process Utility Functions \(https://superfastpython.com/process-utility-functions-in-python/\)](https://superfastpython.com/process-utility-functions-in-python/)
- [How to Query Process Status \(https://superfastpython.com/how-to-query-process-status/\)](https://superfastpython.com/how-to-query-process-status/)
- [How to Restart a Process \(https://superfastpython.com/restart-a-process-in-python/\)](https://superfastpython.com/restart-a-process-in-python/)
- [How to Join a Process \(https://superfastpython.com/join-a-process-in-python/\)](https://superfastpython.com/join-a-process-in-python/)
- [Child stop main process \(https://superfastpython.com/multiprocessing-child-stop-parent-exiting/\)](https://superfastpython.com/multiprocessing-child-stop-parent-exiting/)
- [print\(\) from a Child Process \(https://superfastpython.com/multiprocessing-print/\)](https://superfastpython.com/multiprocessing-print/)

Intermediate

- [Multiprocessing for-loops \(https://superfastpython.com/multiprocessing-for-loop/\)](https://superfastpython.com/multiprocessing-for-loop/)
- [Nested for-loops \(https://superfastpython.com/parallel-nested-for-loops-in-python/\)](https://superfastpython.com/parallel-nested-for-loops-in-python/)
- [Why Not Always Use Processes \(https://superfastpython.com/why-not-always-use-processes-in-python/\)](https://superfastpython.com/why-not-always-use-processes-in-python/)
- [Multiprocessing Best Practices \(https://superfastpython.com/multiprocessing-best-practices/\)](https://superfastpython.com/multiprocessing-best-practices/)
- [Multiprocessing Logging \(https://superfastpython.com/multiprocessing-logging-in-python/\)](https://superfastpython.com/multiprocessing-logging-in-python/)
- [Threads in Processes \(https://superfastpython.com/threads-in-processes-with-python/\)](https://superfastpython.com/threads-in-processes-with-python/)
- [Automatically Start Processes \(https://superfastpython.com/auto-start-process/\)](https://superfastpython.com/auto-start-process/)
- [Return Value From Process \(https://superfastpython.com/multiprocessing-return-value-from-process/\)](https://superfastpython.com/multiprocessing-return-value-from-process/)

Advanced Processes

- [Multiprocessing Context \(https://superfastpython.com/multiprocessing-context-in-python/\)](https://superfastpython.com/multiprocessing-context-in-python/)
- [Multiprocessing Freeze Support \(https://superfastpython.com/multiprocessing-freeze-support-in-python/\)](https://superfastpython.com/multiprocessing-freeze-support-in-python/)
- [Multiprocessing vs GIL \(https://superfastpython.com/multiprocessing-vs-gil-in-python/\)](https://superfastpython.com/multiprocessing-vs-gil-in-python/)
- [Multiprocessing vs Threading \(https://superfastpython.com/threading-vs-multiprocessing-in-python/\)](https://superfastpython.com/threading-vs-multiprocessing-in-python/)

- [Inherit Global Variables \(https://superfastpython.com/multiprocessing-inherit-global-variables-in-python/\)](https://superfastpython.com/multiprocessing-inherit-global-variables-in-python/)
- [Shared ctypes in Python \(https://superfastpython.com/multiprocessing-shared-ctypes-in-python/\)](https://superfastpython.com/multiprocessing-shared-ctypes-in-python/)

Failure Modes

- [Race Conditions \(https://superfastpython.com/multiprocessing-race-condition-python/\)](https://superfastpython.com/multiprocessing-race-condition-python/)
- [Performance may not scale \(https://superfastpython.com/performance-may-not-scale-with-cpu-cores-python/\)](https://superfastpython.com/performance-may-not-scale-with-cpu-cores-python/)
- [Common Multiprocessing Errors \(https://superfastpython.com/multiprocessing-common-errors/\)](https://superfastpython.com/multiprocessing-common-errors/)
- [Add if `__name__ == '__main__'` \(https://superfastpython.com/multiprocessing-spawn-runtimeerror/\)](https://superfastpython.com/multiprocessing-spawn-runtimeerror/)
- [Fix FileNotFoundError \(https://superfastpython.com/fileNotFoundError-multiprocessing-python/\)](https://superfastpython.com/fileNotFoundError-multiprocessing-python/)
- [Shared Process Class Attributes \(https://superfastpython.com/share-process-attributes/\)](https://superfastpython.com/share-process-attributes/)

Parents and Children

- [Parent Process vs Child Process \(https://superfastpython.com/parent-process-vs-child-process-in-python/\)](https://superfastpython.com/parent-process-vs-child-process-in-python/)
- [Get All Child Processes \(https://superfastpython.com/multiprocessing-active-children/\)](https://superfastpython.com/multiprocessing-active-children/)
- [Get Parent Process \(https://superfastpython.com/multiprocessing-parent/\)](https://superfastpython.com/multiprocessing-parent/)

Sync Primitives

- [Process-Safe in Python \(https://superfastpython.com/process-safe-in-python/\)](https://superfastpython.com/process-safe-in-python/)
- [Multiprocessing Lock \(https://superfastpython.com/multiprocessing-mutex-lock-in-python/\)](https://superfastpython.com/multiprocessing-mutex-lock-in-python/)
- [Multiprocessing RLock \(https://superfastpython.com/multiprocessing-rlock-in-python/\)](https://superfastpython.com/multiprocessing-rlock-in-python/)
- [Multiprocessing Event Object \(https://superfastpython.com/multiprocessing-event-object-in-python/\)](https://superfastpython.com/multiprocessing-event-object-in-python/)

- [Multiprocessing Semaphore \(https://superfastpython.com/multiprocessing-semaphore-in-python/\)](https://superfastpython.com/multiprocessing-semaphore-in-python/)
- [Multiprocessing Condition Var \(https://superfastpython.com/multiprocessing-condition-variable-in-python/\)](https://superfastpython.com/multiprocessing-condition-variable-in-python/)
- [Multiprocessing Barrier \(https://superfastpython.com/multiprocessing-barrier-in-python/\)](https://superfastpython.com/multiprocessing-barrier-in-python/)

Queues and Pipes

- [Multiprocessing Queue \(https://superfastpython.com/multiprocessing-queue-in-python/\)](https://superfastpython.com/multiprocessing-queue-in-python/)
- [Multiprocessing SimpleQueue \(https://superfastpython.com/multiprocessing-simplequeue-in-python/\)](https://superfastpython.com/multiprocessing-simplequeue-in-python/)
- [Multiprocessing JoinableQueue \(https://superfastpython.com/multiprocessing-joinablequeue-on-python/\)](https://superfastpython.com/multiprocessing-joinablequeue-on-python/)
- [Multiprocessing Pipe \(https://superfastpython.com/multiprocessing-pipe-in-python/\)](https://superfastpython.com/multiprocessing-pipe-in-python/)

Killing Processes

- [How to Safely Stop a Process \(https://superfastpython.com/safely-stop-a-process-in-python/\)](https://superfastpython.com/safely-stop-a-process-in-python/)
- [Kill a Process in Python \(https://superfastpython.com/kill-a-process-in-python/\)](https://superfastpython.com/kill-a-process-in-python/)
- [Kill Process By PID \(https://superfastpython.com/kill-process-by-pid-in-python/\)](https://superfastpython.com/kill-process-by-pid-in-python/)
- [Kill All Child Processes \(https://superfastpython.com/kill-all-child-processes-in-python/\)](https://superfastpython.com/kill-all-child-processes-in-python/)
- [Orphaned Processes \(https://superfastpython.com/orphan-process-in-python/\)](https://superfastpython.com/orphan-process-in-python/)
- [Exit a Process with sys.exit\(\) \(https://superfastpython.com/exit-process/\)](https://superfastpython.com/exit-process/)
- [Process Exit Codes \(https://superfastpython.com/exit-codes-in-python/\)](https://superfastpython.com/exit-codes-in-python/)

Managers

- [What is a manager \(https://superfastpython.com/multiprocessing-manager/\)](https://superfastpython.com/multiprocessing-manager/)
- [Manager example \(https://superfastpython.com/multiprocessing-manager-example/\)](https://superfastpython.com/multiprocessing-manager-example/)
- [Manager with custom class \(https://superfastpython.com/multiprocessing-manager-custom-class/\)](https://superfastpython.com/multiprocessing-manager-custom-class/)
- [Manager with primitives \(https://superfastpython.com/multiprocessing-manager-share-concurrency-primitive/\)](https://superfastpython.com/multiprocessing-manager-share-concurrency-primitive/)

- [Manager share object \(https://superfastpython.com/multiprocessing-share-object-with-processes/\)](https://superfastpython.com/multiprocessing-share-object-with-processes/).
- [Manager share queue \(https://superfastpython.com/multiprocessing-manager-share-queue/\)](https://superfastpython.com/multiprocessing-manager-share-queue/).
- [Manager server process \(https://superfastpython.com/multiprocessing-manager-server-process/\)](https://superfastpython.com/multiprocessing-manager-server-process/).
- [Manager shared namespace \(https://superfastpython.com/multiprocessing-manager-namespace/\)](https://superfastpython.com/multiprocessing-manager-namespace/).
- [Manager nested proxies \(https://superfastpython.com/multiprocessing-manager-nested-proxy-objects/\)](https://superfastpython.com/multiprocessing-manager-nested-proxy-objects/).

Asyncio Learning Path

Python provides coroutine-based concurrency in the “**asyncio**” module.

This section provides a learning path that you can use to get started and get effective with the **asyncio** module, super fast.

- **Step 1: Discover asyncio programs in Python.**
 - [Asyncio Hello World Tutorial in Python \(https://superfastpython.com/asyncio-hello-world/\)](https://superfastpython.com/asyncio-hello-world/).
 - [How to Run an Asyncio Program in Python \(https://superfastpython.com/asyncio-run-program/\)](https://superfastpython.com/asyncio-run-program/).
- **Step 2: Discover how to create and run tasks in asyncio.**
 - [What is an Asyncio Task \(https://superfastpython.com/asyncio-task/\)](https://superfastpython.com/asyncio-task/).
 - [How to Create Asyncio Tasks in Python \(https://superfastpython.com/asyncio-create-task/\)](https://superfastpython.com/asyncio-create-task/).
- **Step 3: Discover how to get the most out of asyncio.**
 - [Asyncio in Python: The Complete Guide \(https://superfastpython.com/python-asyncio/\)](https://superfastpython.com/python-asyncio/).
 - [Python Asyncio Jump-Start \(https://superfastpython.com/products-paj\)](https://superfastpython.com/products-paj) (**my book**)

Next, you can [browse all tutorials \(https://superfastpython.com/category/asyncio/\)](https://superfastpython.com/category/asyncio/) on the topic.

ThreadPoolExecutor Learning Path

Python provides thread pool-based concurrency with the “**ThreadPoolExecutor**” class.

This section provides a learning path that you can use to get started and get effective with the **ThreadPoolExecutor** class, super fast.

- **Step 1: Discover the ThreadPoolExecutor class in Python.**
 - [What Is the ThreadPoolExecutor \(https://superfastpython.com/what-is-the-threadpoolexecutor/\)](https://superfastpython.com/what-is-the-threadpoolexecutor/)
 - [ThreadPoolExecutor in Python: Quick-Start Guide \(https://superfastpython.com/threadpoolexecutor-quick-start-guide/\)](https://superfastpython.com/threadpoolexecutor-quick-start-guide/)
- **Step 2: Discover how to run code in the ThreadPoolExecutor.**
 - [How to Use as_completed\(\) with the ThreadPoolExecutor in Python \(https://superfastpython.com/threadpoolexecutor-as-completed/\)](https://superfastpython.com/threadpoolexecutor-as-completed/)
 - [How to Use map\(\) with the ThreadPoolExecutor in Python \(https://superfastpython.com/threadpoolexecutor-map/\)](https://superfastpython.com/threadpoolexecutor-map/)
 - [What are Futures in the ThreadPoolExecutor \(https://superfastpython.com/threadpoolexecutor-futures/\)](https://superfastpython.com/threadpoolexecutor-futures/)
- **Step 3: Discover how to get the most out of ThreadPoolExecutor.**
 - [ThreadPoolExecutor in Python: The Complete Guide \(https://superfastpython.com/threadpoolexecutor-in-python/\)](https://superfastpython.com/threadpoolexecutor-in-python/)
 - [Python ThreadPoolExecutor Jump-Start \(https://superfastpython.com/products-ptpej\)](https://superfastpython.com/products-ptpej)
(my book)

Next, you can [browse all tutorials](#)

(<https://superfastpython.com/category/threadpoolexecutor/>) or dive into a topic below to learn more.

Configure

- [Configure Max Workers \(https://superfastpython.com/threadpoolexecutor-number-of-threads/\)](https://superfastpython.com/threadpoolexecutor-number-of-threads/)

- [How to Configure Thread Names \(https://superfastpython.com/threadpoolexecutor-thread-names/\)](https://superfastpython.com/threadpoolexecutor-thread-names/)
- [Worker Initializer Functions \(https://superfastpython.com/threadpoolexecutor-initializer/\)](https://superfastpython.com/threadpoolexecutor-initializer/)
- [How to Use the Context Manager \(https://superfastpython.com/threadpoolexecutor-context-manager/\)](https://superfastpython.com/threadpoolexecutor-context-manager/)

Basics

- [map\(\) vs. submit\(\) \(https://superfastpython.com/threadpoolexecutor-map-vs-submit/\)](https://superfastpython.com/threadpoolexecutor-map-vs-submit/)
- [wait\(\) vs. as_completed\(\) \(https://superfastpython.com/threadpoolexecutor-wait-vs-as-completed/\)](https://superfastpython.com/threadpoolexecutor-wait-vs-as-completed/)
- [Common Usage Patterns \(https://superfastpython.com/threadpoolexecutor-usage-patterns/\)](https://superfastpython.com/threadpoolexecutor-usage-patterns/)
- [Wait For The First Task To Finish \(https://superfastpython.com/threadpoolexecutor-wait-first-result/\)](https://superfastpython.com/threadpoolexecutor-wait-first-result/)
- [Wait For All Tasks to Finish \(https://superfastpython.com/threadpoolexecutor-wait-all-tasks/\)](https://superfastpython.com/threadpoolexecutor-wait-all-tasks/)
- [How To Get Task Results \(https://superfastpython.com/threadpoolexecutor-get-results/\)](https://superfastpython.com/threadpoolexecutor-get-results/)
- [How to Check Task Status \(https://superfastpython.com/threadpoolexecutor-task-status/\)](https://superfastpython.com/threadpoolexecutor-task-status/)
- [How to Add a Callback to a Task \(https://superfastpython.com/threadpoolexecutor-add-callback/\)](https://superfastpython.com/threadpoolexecutor-add-callback/)

Intermediate

- [Best Practices \(https://superfastpython.com/threadpoolexecutor-best-practices/\)](https://superfastpython.com/threadpoolexecutor-best-practices/)
- [How to Submit All Function Types \(https://superfastpython.com/threadpoolexecutor-all-function-types/\)](https://superfastpython.com/threadpoolexecutor-all-function-types/)
- [How to Check Thread Names \(https://superfastpython.com/threadpoolexecutor-thread-names-2/\)](https://superfastpython.com/threadpoolexecutor-thread-names-2/)
- [Handle Exceptions in Tasks \(https://superfastpython.com/threadpoolexecutor-exception-handling-2/\)](https://superfastpython.com/threadpoolexecutor-exception-handling-2/)
- [Handle Exceptions with Pool \(https://superfastpython.com/threadpoolexecutor-exception-handling/\)](https://superfastpython.com/threadpoolexecutor-exception-handling/)
- [How to Use With Thread-Local \(https://superfastpython.com/threadpoolexecutor-thread-local/\)](https://superfastpython.com/threadpoolexecutor-thread-local/)

- [Number of Remaining Tasks \(https://superfastpython.com/number-of-remaining-tasks-in-the-threadpoolexecutor/\)](https://superfastpython.com/number-of-remaining-tasks-in-the-threadpoolexecutor/)
- [How to Show Progress for Tasks \(https://superfastpython.com/threadpoolexecutor-progress/\)](https://superfastpython.com/threadpoolexecutor-progress/)
- [Log From Tasks \(https://superfastpython.com/threadpoolexecutor-logging/\)](https://superfastpython.com/threadpoolexecutor-logging/)
- [How To Stop Running Tasks \(https://superfastpython.com/threadpoolexecutor-stop-tasks/\)](https://superfastpython.com/threadpoolexecutor-stop-tasks/)
- [How to Cancel Tasks \(https://superfastpython.com/threadpoolexecutor-cancel-task/\)](https://superfastpython.com/threadpoolexecutor-cancel-task/)
- [How Does it Work \(https://superfastpython.com/how-does-threadpoolexecutor-work/\)](https://superfastpython.com/how-does-threadpoolexecutor-work/)

Comparisons

- [ThreadPoolExecutor vs. Thread \(https://superfastpython.com/threadpoolexecutor-vs-threads/\)](https://superfastpython.com/threadpoolexecutor-vs-threads/)
- [ThreadPoolExecutor vs. GIL \(https://superfastpython.com/threadpoolexecutor-vs-gil/\)](https://superfastpython.com/threadpoolexecutor-vs-gil/)
- [ThreadPoolExecutor vs. AsyncIO \(https://superfastpython.com/threadpoolexecutor-vs-asyncio/\)](https://superfastpython.com/threadpoolexecutor-vs-asyncio/)
- [ThreadPoolExecutor vs. PPE \(https://superfastpython.com/threadpoolexecutor-vs-processpoolexecutor/\)](https://superfastpython.com/threadpoolexecutor-vs-processpoolexecutor/)

Advanced

- [How to Limit Pending Tasks \(https://superfastpython.com/threadpoolexecutor-limit-pending-tasks/\)](https://superfastpython.com/threadpoolexecutor-limit-pending-tasks/)
- [How to Retry Failed Tasks \(https://superfastpython.com/threadpoolexecutor-retry-tasks/\)](https://superfastpython.com/threadpoolexecutor-retry-tasks/)
- [How to Stop All Tasks if Task Fails \(https://superfastpython.com/threadpoolexecutor-stop-if-task-fails/\)](https://superfastpython.com/threadpoolexecutor-stop-if-task-fails/)
- [How to Submit a Follow-up Task \(https://superfastpython.com/threadpoolexecutor-followup-task/\)](https://superfastpython.com/threadpoolexecutor-followup-task/)
- [ThreadPoolExecutor Thread-Safe \(https://superfastpython.com/threadpoolexecutor-thread-safe/\)](https://superfastpython.com/threadpoolexecutor-thread-safe/)
- [How To Identify Deadlocks \(https://superfastpython.com/threadpoolexecutor-deadlock/\)](https://superfastpython.com/threadpoolexecutor-deadlock/)

Failure Modes

Failure Modes

- [Common Errors](https://superfastpython.com/threadpoolexecutor-common-errors/) (<https://superfastpython.com/threadpoolexecutor-common-errors/>)
- [Why ThreadPoolExecutor Slower?](https://superfastpython.com/threadpoolexecutor-slower/) (<https://superfastpython.com/threadpoolexecutor-slower/>)
- [Why Do Tasks Fail Silently?](https://superfastpython.com/threadpoolexecutor-fails-silently/) (<https://superfastpython.com/threadpoolexecutor-fails-silently/>)

Example Applications

- [Download API Docs Concurrently](https://superfastpython.com/threadpoolexecutor-example/) (<https://superfastpython.com/threadpoolexecutor-example/>)
- [Download Files \(API Example\)](https://superfastpython.com/threadpoolexecutor-api-example/) (<https://superfastpython.com/threadpoolexecutor-api-example/>)
- [Download Files on Webpage](https://superfastpython.com/threadpoolexecutor-download-files/) (<https://superfastpython.com/threadpoolexecutor-download-files/>)
- [Download Books Concurrently](https://superfastpython.com/threadpoolexecutor-download-books/) (<https://superfastpython.com/threadpoolexecutor-download-books/>)
- [Validate Links Concurrently](https://superfastpython.com/threadpoolexecutor-validate-links/) (<https://superfastpython.com/threadpoolexecutor-validate-links/>)
- [Multithreaded Port Scanner](https://superfastpython.com/threadpoolexecutor-port-scanner/) (<https://superfastpython.com/threadpoolexecutor-port-scanner/>)
- [Query Quake III Arena Servers](https://superfastpython.com/threadpoolexecutor-query-quake3-servers/) (<https://superfastpython.com/threadpoolexecutor-query-quake3-servers/>)
- [Check the Status of Websites](https://superfastpython.com/threadpoolexecutor-website-status/) (<https://superfastpython.com/threadpoolexecutor-website-status/>)
- [Analyze HackerNews Posts](https://superfastpython.com/threadpoolexecutor-download-hackernews-articles/) (<https://superfastpython.com/threadpoolexecutor-download-hackernews-articles/>)

ProcessPoolExecutor Learning Path

Python provides process pool-based concurrency with the “**ProcessPoolExecutor**” class.

This section provides a learning path that you can use to get started and get effective with the **ProcessPoolExecutor** class, super fast.

- **Step 1: Discover the ProcessPoolExecutor class in Python.**
 - [What is the ProcessPoolExecutor \(https://superfastpython.com/what-is-the-processpoolexecutor/\)](https://superfastpython.com/what-is-the-processpoolexecutor/)
 - [ProcessPoolExecutor Quick-Start Guide \(https://superfastpython.com/processpoolexecutor-quick-start-guide/\)](https://superfastpython.com/processpoolexecutor-quick-start-guide/)
- **Step 2: Discover how to run code in the ProcessPoolExecutor.**
 - [How to Use map\(\) With the ProcessPoolExecutor in Python \(https://superfastpython.com/processpoolexecutor-map/\)](https://superfastpython.com/processpoolexecutor-map/)
 - [How to use as_completed\(\) With the ProcessPoolExecutor in Python \(https://superfastpython.com/processpoolexecutor-as-completed/\)](https://superfastpython.com/processpoolexecutor-as-completed/)
 - [What Are Future Objects in the ProcessPoolExecutor \(https://superfastpython.com/processpoolexecutor-future-objects/\)](https://superfastpython.com/processpoolexecutor-future-objects/)
- **Step 3: Discover how to get the most out of ProcessPoolExecutor.**
 - [ProcessPoolExecutor in Python: The Complete Guide \(https://superfastpython.com/processpoolexecutor-in-python/\)](https://superfastpython.com/processpoolexecutor-in-python/)
 - [Python ProcessPoolExecutor Jump-Start \(https://superfastpython.gumroad.com/l/pppej\)](https://superfastpython.gumroad.com/l/pppej) (**my book**)

Next, you can [browse all tutorials \(https://superfastpython.com/category/processpoolexecutor/\)](https://superfastpython.com/category/processpoolexecutor/) or dive into a topic below to learn more.

Configure

- [Configure Max Workers \(https://superfastpython.com/processpoolexecutor-number-of-workers/\)](https://superfastpython.com/processpoolexecutor-number-of-workers/)
- [Config Worker Initialize Function \(https://superfastpython.com/processpoolexecutor-initializer/\)](https://superfastpython.com/processpoolexecutor-initializer/)
- [How to Use The Context Manager \(https://superfastpython.com/processpoolexecutor-context-manager/\)](https://superfastpython.com/processpoolexecutor-context-manager/)

- [Config Multiprocessing Context \(https://superfastpython.com/processpoolexecutor-multiprocessing-context/\)](https://superfastpython.com/processpoolexecutor-multiprocessing-context/)

Basics

- [map\(\) vs submit\(\) \(https://superfastpython.com/processpoolexecutor-map-vs-submit/\)](https://superfastpython.com/processpoolexecutor-map-vs-submit/)
- [wait\(\) vs as_completed\(\) \(https://superfastpython.com/processpoolexecutor_wait_vs_as_completed/\)](https://superfastpython.com/processpoolexecutor_wait_vs_as_completed/)
- [Common Usage Patterns \(https://superfastpython.com/processpoolexecutor-usage-patterns/\)](https://superfastpython.com/processpoolexecutor-usage-patterns/)
- [How to Wait For The First Result \(https://superfastpython.com/processpoolexecutor-first-result/\)](https://superfastpython.com/processpoolexecutor-first-result/)
- [How to Wait For a Task to Finish \(https://superfastpython.com/processpoolexecutor-wait-tasks/\)](https://superfastpython.com/processpoolexecutor-wait-tasks/)
- [How to Get Results \(https://superfastpython.com/how-to-get-results-from-the-processpoolexecutor-in-python/\)](https://superfastpython.com/how-to-get-results-from-the-processpoolexecutor-in-python/)
- [How to Check the Status of a Task \(https://superfastpython.com/processpoolexecutor-task-status/\)](https://superfastpython.com/processpoolexecutor-task-status/)
- [How to Add a Callback to a Task \(https://superfastpython.com/processpoolexecutor-add-callback/\)](https://superfastpython.com/processpoolexecutor-add-callback/)

Intermediate

- [Best Practices \(https://superfastpython.com/processpoolexecutor-best-practices/\)](https://superfastpython.com/processpoolexecutor-best-practices/)
- [How to Handle Exceptions \(https://superfastpython.com/processpoolexecutor-exception-handling/\)](https://superfastpython.com/processpoolexecutor-exception-handling/)
- [Task Exception Handling \(https://superfastpython.com/processpoolexecutor-exception-handling-2/\)](https://superfastpython.com/processpoolexecutor-exception-handling-2/)
- [How Does It Work \(https://superfastpython.com/how-does-processpoolexecutor-work/\)](https://superfastpython.com/how-does-processpoolexecutor-work/)
- [How to Show Progress \(https://superfastpython.com/processpoolexecutor-show-progress/\)](https://superfastpython.com/processpoolexecutor-show-progress/)
- [Number of Remaining Tasks \(https://superfastpython.com/processpoolexecutor-remaining-tasks/\)](https://superfastpython.com/processpoolexecutor-remaining-tasks/)
- [Check PIDs and Thread Names \(https://superfastpython.com/processpoolexecutor-processes-and-threads/\)](https://superfastpython.com/processpoolexecutor-processes-and-threads/)

- [How to Cancel Tasks \(https://superfastpython.com/processpoolexecutor-cancel-task/\)](https://superfastpython.com/processpoolexecutor-cancel-task/)
- [Common Errors \(https://superfastpython.com/processpoolexecutor-common-errors/\)](https://superfastpython.com/processpoolexecutor-common-errors/)

Advanced

- [How to Limit Pending Tasks \(https://superfastpython.com/processpoolexecutor-limit-tasks/\)](https://superfastpython.com/processpoolexecutor-limit-tasks/)
- [How To Retry Failed Tasks \(https://superfastpython.com/processpoolexecutor-retry-tasks/\)](https://superfastpython.com/processpoolexecutor-retry-tasks/)
- [Stop All Tasks if One Fails \(https://superfastpython.com/processpoolexecutor-stop-all-tasks-if-one-fails/\)](https://superfastpython.com/processpoolexecutor-stop-all-tasks-if-one-fails/)
- [How To Stop Running Tasks \(https://superfastpython.com/processpoolexecutor-stop-running-tasks/\)](https://superfastpython.com/processpoolexecutor-stop-running-tasks/)
- [How to Submit a Follow-up Task \(https://superfastpython.com/processpoolexecutor-followup-task/\)](https://superfastpython.com/processpoolexecutor-followup-task/)
- [Use TPE in ProcessPoolExecutor \(https://superfastpython.com/threadpoolexecutor-within-processpoolexecutor-in-python/\)](https://superfastpython.com/threadpoolexecutor-within-processpoolexecutor-in-python/)

Example Applications

- [Hash a Dictionary of Words \(https://superfastpython.com/processpoolexecutor-example/\)](https://superfastpython.com/processpoolexecutor-example/)
- [Search Text Files Concurrently \(https://superfastpython.com/processpoolexecutor-search-text-files/\)](https://superfastpython.com/processpoolexecutor-search-text-files/)
- [Calculate Fibonacci Numbers \(https://superfastpython.com/processpoolexecutor-fibonacci-numbers/\)](https://superfastpython.com/processpoolexecutor-fibonacci-numbers/)
- [Monte Carlo Estimate of Pi \(https://superfastpython.com/processpoolexecutor-monte-carlo-pi/\)](https://superfastpython.com/processpoolexecutor-monte-carlo-pi/)
- [Check if Numbers are Prime \(https://superfastpython.com/processpoolexecutor-prime-numbers/\)](https://superfastpython.com/processpoolexecutor-prime-numbers/)
- [Create Photo Thumbnails \(https://superfastpython.com/processpoolexecutor-thumbnail-images/\)](https://superfastpython.com/processpoolexecutor-thumbnail-images/)

ThreadPool Learning Path

Python provides thread pool-based concurrency with the “**multiprocessing.pool.ThreadPool**” class.

This section provides a learning path that you can use to get started and get effective with the **ThreadPool** class, super fast.

- **Step 1: Discover the ThreadPool class in Python.**
 - [ThreadPool Class in Python](https://superfastpython.com/threadpool-class-in-python/) (<https://superfastpython.com/threadpool-class-in-python/>).
 - [ThreadPool Example](https://superfastpython.com/threadpool-example/) (<https://superfastpython.com/threadpool-example/>).
- **Step 2: Discover how to run code in the ThreadPool.**
 - [ThreadPool apply\(\) vs map\(\) vs imap\(\) vs starmap\(\)](https://superfastpython.com/threadpool-apply-vs-map-vs-imap-vs-starmap/) (<https://superfastpython.com/threadpool-apply-vs-map-vs-imap-vs-starmap/>).
- **Step 3: Discover how to get the most out of ThreadPool.**
 - [ThreadPool in Python: The Complete Guide](https://superfastpython.com/threadpool-python/) (<https://superfastpython.com/threadpool-python/>).
 - [Python ThreadPool Jump-Start](https://superfastpython.com/ptpj-sidebar) (<https://superfastpython.com/ptpj-sidebar>) (**my book**)

Next, you can [browse all tutorials](https://superfastpython.com/category/threadpool/) (<https://superfastpython.com/category/threadpool/>) on this topic.

Basics

- [The dummy module](https://superfastpython.com/multiprocessing-dummy-module-python/) (<https://superfastpython.com/multiprocessing-dummy-module-python/>).
- [ThreadPool class](https://superfastpython.com/threadpool-class-in-python/) (<https://superfastpython.com/threadpool-class-in-python/>).
- [ThreadPool life-cycle](https://superfastpython.com/threadpool-life-cycle/) (<https://superfastpython.com/threadpool-life-cycle/>).
- [Should we use ThreadPool](https://superfastpython.com/should-we-use-the-threadpool-class/) (<https://superfastpython.com/should-we-use-the-threadpool-class/>).
- [Concurrent for-loops](https://superfastpython.com/threadpool-for-loop/) (<https://superfastpython.com/threadpool-for-loop/>).

Configure

- [Config num workers](https://superfastpython.com/threadpool-number-of-workers/) (<https://superfastpython.com/threadpool-number-of-workers/>).
- [Config worker init](https://superfastpython.com/threadpool-worker-initializer/) (<https://superfastpython.com/threadpool-worker-initializer/>).

- [Use context manager \(https://superfastpython.com/threadpool-context-manager/\)](https://superfastpython.com/threadpool-context-manager/)
- [How to configure \(https://superfastpython.com/how-to-configure-the-threadpool/\)](https://superfastpython.com/how-to-configure-the-threadpool/)

Intermediate

- [How to shutdown \(https://superfastpython.com/threadpool-close-and-terminate/\)](https://superfastpython.com/threadpool-close-and-terminate/)
- [terminate\(\) not supported \(https://superfastpython.com/threadpool-terminate/\)](https://superfastpython.com/threadpool-terminate/)
- [How to join \(https://superfastpython.com/threadpool-join/\)](https://superfastpython.com/threadpool-join/)
- [Get worker names \(https://superfastpython.com/threadpool-worker-names/\)](https://superfastpython.com/threadpool-worker-names/)
- [Get worker thread details \(https://superfastpython.com/threadpool-thread-details/\)](https://superfastpython.com/threadpool-thread-details/)

Advanced

- [When are workers started \(https://superfastpython.com/threadpool-when-workers-started/\)](https://superfastpython.com/threadpool-when-workers-started/)
- [Does ThreadPool stop main \(https://superfastpython.com/threadpool-stop-main-exiting/\)](https://superfastpython.com/threadpool-stop-main-exiting/)
- [How to get results from async \(https://superfastpython.com/threadpool-get-result/\)](https://superfastpython.com/threadpool-get-result/)
- [Tasks fail silently \(https://superfastpython.com/threadpool-fail-silently/\)](https://superfastpython.com/threadpool-fail-silently/)
- [Is ThreadPool slower \(https://superfastpython.com/threadpool-slower/\)](https://superfastpython.com/threadpool-slower/)
- [Get num remaining tasks \(https://superfastpython.com/threadpool-remaining-tasks/\)](https://superfastpython.com/threadpool-remaining-tasks/)
- [ThreadPool exception handling \(https://superfastpython.com/threadpool-exception-handling/\)](https://superfastpython.com/threadpool-exception-handling/)

Callback Functions

- [ThreadPool callback functions \(https://superfastpython.com/threadpool-callback/\)](https://superfastpython.com/threadpool-callback/)
- [Use error callback functions \(https://superfastpython.com/threadpool-error-callback/\)](https://superfastpython.com/threadpool-error-callback/)

Issue Tasks

- [Issue tasks with apply \(https://superfastpython.com/threadpool-apply/\)](https://superfastpython.com/threadpool-apply/)
- [Issue tasks with apply_async \(https://superfastpython.com/threadpool-apply_async/\)](https://superfastpython.com/threadpool-apply_async/)
- [Issue tasks with map \(https://superfastpython.com/threadpool-map/\)](https://superfastpython.com/threadpool-map/)
- [Issue tasks with map_async \(https://superfastpython.com/threadpool-map_async/\)](https://superfastpython.com/threadpool-map_async/)
- [Issue tasks with imap \(https://superfastpython.com/threadpool-imap/\)](https://superfastpython.com/threadpool-imap/)

- [imap_unordered](https://superfastpython.com/threadpool-<u>imap_unordered</u>/)
- [starmap](https://superfastpython.com/threadpool-<u>starmap</u>/)
- [starmap_async](https://superfastpython.com/threadpool-<u>starmap_async</u>/)
- [apply-vs-map-vs-
imap-vs-starmap](https://superfastpython.com/threadpool-<u>apply-vs-map-vs-
imap-vs-starmap</u>/)
- [map with multiple args](https://superfastpython.com/threadpool-<u>map-multiple-arguments</u>/)
- [How to use AsyncResult](https://superfastpython.com/threadpool-<u>AsyncResult</u>/)
- [How to configure chunksize](https://superfastpython.com/threadpool-<u>map-chunksize</u>/)

Common Tasks

- [ThreadPool best practices](https://superfastpython.com/threadpool-<u>best-practices</u>/)
- [Common usage patterns](https://superfastpython.com/threadpool-<u>usage-patterns</u>/)
- [Common errors](https://superfastpython.com/threadpool-<u>common-errors</u>/)
- [How to log from tasks](https://superfastpython.com/threadpool-<u>logging</u>/)
- [How to stop all tasks](https://superfastpython.com/threadpool-<u>stop-all-tasks</u>/)
- [Stop all if one task fails](https://superfastpython.com/threadpool-<u>stop-all-tasks-on-failure</u>/)
- [Issue follow-up tasks](https://superfastpython.com/threadpool-<u>follow-up-tasks</u>/)
- [Wait for all tasks to finish](https://superfastpython.com/threadpool-<u>wait-for-all-tasks</u>/)
- [How to show progress](https://superfastpython.com/threadpool-<u>show-progress</u>/)
- [How to get first result](https://superfastpython.com/threadpool-<u>first-result</u>/)

Comparisons

- [ThreadPool vs GIL](https://superfastpython.com/threadpool-<u>gil</u>/)
- [ThreadPool vs Pool](https://superfastpython.com/threadpool-<u>vs-pool-in-python</u>/)
- [ThreadPool vs Thread](https://superfastpython.com/threadpool-<u>vs-thread</u>/)
- [ThreadPool vs ThreadPoolExecutor](https://superfastpython.com/threadpool-<u>vs-threadpoolexecutor</u>/)

Multiprocessing Pool Learning Path

Python provides process pool-based concurrency with the “**multiprocessing.Pool**” class.

This section provides a learning path that you can use to get started and get effective with the **multiprocessing.Pool** class, super fast.

- **Step 1: Discover the multiprocessing.Pool class in Python.**
 - [Multiprocessing Pool Class in Python \(https://superfastpython.com/multiprocessing-pool-class/\)](https://superfastpython.com/multiprocessing-pool-class/)
 - [Multiprocessing Pool Example \(https://superfastpython.com/multiprocessing-pool-example/\)](https://superfastpython.com/multiprocessing-pool-example/)
- **Step 2: Discover how to run code in the multiprocessing.Pool.**
 - [Pool apply\(\) vs map\(\) vs imap\(\) vs starmap\(\) \(https://superfastpython.com/multiprocessing-pool-issue-tasks/\)](https://superfastpython.com/multiprocessing-pool-issue-tasks/)
- **Step 3: Discover how to get the most out of multiprocessing.Pool.**
 - [Multiprocessing Pool in Python: The Complete Guide \(https://superfastpython.com/multiprocessing-pool-python/\)](https://superfastpython.com/multiprocessing-pool-python/)
 - [Python Multiprocessing Pool Jump-Start \(https://superfastpython.com/products-pmpj\)](https://superfastpython.com/products-pmpj) **(my book)**

Next, you can [browse all tutorials \(https://superfastpython.com/category/pool/\)](https://superfastpython.com/category/pool/) on this topic.

Configure

- [Config num workers \(https://superfastpython.com/multiprocessing-pool-num-workers/\)](https://superfastpython.com/multiprocessing-pool-num-workers/)
- [Use worker initializer \(https://superfastpython.com/multiprocessing-pool-initializer/\)](https://superfastpython.com/multiprocessing-pool-initializer/)
- [Config max tasks per child \(https://superfastpython.com/multiprocessing-pool-max-tasks-per-child-in-python/\)](https://superfastpython.com/multiprocessing-pool-max-tasks-per-child-in-python/)
- [Config pool context \(https://superfastpython.com/multiprocessing-pool-context/\)](https://superfastpython.com/multiprocessing-pool-context/)
- [How to configure \(https://superfastpython.com/multiprocessing-pool-configure/\)](https://superfastpython.com/multiprocessing-pool-configure/)

Basics

- [Pool PEP \(https://superfastpython.com/multiprocessing-pool-pep/\)](https://superfastpython.com/multiprocessing-pool-pep/)
- [Context manager interface \(https://superfastpython.com/multiprocessing-pool-context-manager/\)](https://superfastpython.com/multiprocessing-pool-context-manager/)
- [How to shutdown \(https://superfastpython.com/shutdown-the-multiprocessing-pool-in-python/\)](https://superfastpython.com/shutdown-the-multiprocessing-pool-in-python/)
- [How to join \(https://superfastpython.com/join-a-multiprocessing-pool-in-python/\)](https://superfastpython.com/join-a-multiprocessing-pool-in-python/)
- [Exception Handling \(https://superfastpython.com/multiprocessing-pool-exception-handling/\)](https://superfastpython.com/multiprocessing-pool-exception-handling/)
- [Pool stop main exiting \(https://superfastpython.com/multiprocessing-pool-stop-main-exiting/\)](https://superfastpython.com/multiprocessing-pool-stop-main-exiting/)
- [When workers are started \(https://superfastpython.com/multiprocessing-pool-when-workers-started/\)](https://superfastpython.com/multiprocessing-pool-when-workers-started/)
- [Workers restarted if killed \(https://superfastpython.com/multiprocessing-pool-workers-restarted-if-killed/\)](https://superfastpython.com/multiprocessing-pool-workers-restarted-if-killed/)

Intermediate

- [Share global vars \(https://superfastpython.com/multiprocessing-pool-shared-global-variables/\)](https://superfastpython.com/multiprocessing-pool-shared-global-variables/)
- [Stop all tasks \(https://superfastpython.com/multiprocessing-pool-stop-all-tasks/\)](https://superfastpython.com/multiprocessing-pool-stop-all-tasks/)
- [Kill all tasks \(https://superfastpython.com/multiprocessing-pool-kill-all-tasks/\)](https://superfastpython.com/multiprocessing-pool-kill-all-tasks/)
- [Get all worker PIDs \(https://superfastpython.com/multiprocessing-pool-pid/\)](https://superfastpython.com/multiprocessing-pool-pid/)
- [Get all worker names \(https://superfastpython.com/multiprocessing-pool-worker-names/\)](https://superfastpython.com/multiprocessing-pool-worker-names/)
- [Get all thread details \(https://superfastpython.com/multiprocessing-pool-thread-details/\)](https://superfastpython.com/multiprocessing-pool-thread-details/)
- [Tasks fail silently \(https://superfastpython.com/multiprocessing-pool-fail-silently/\)](https://superfastpython.com/multiprocessing-pool-fail-silently/)
- [Get first result \(https://superfastpython.com/multiprocessing-pool-first-result/\)](https://superfastpython.com/multiprocessing-pool-first-result/)
- [Get result from async \(https://superfastpython.com/multiprocessing-pool-get-result/\)](https://superfastpython.com/multiprocessing-pool-get-result/)

Advanced

- [Wait for all tasks to finish \(https://superfastpython.com/multiprocessing-pool-wait-for-all-tasks/\)](https://superfastpython.com/multiprocessing-pool-wait-for-all-tasks/)
- [Stop all if one fails \(https://superfastpython.com/multiprocessing-pool-stop-all-tasks-on-failure/\)](https://superfastpython.com/multiprocessing-pool-stop-all-tasks-on-failure/)
- [Issue follow-up tasks \(https://superfastpython.com/multiprocessing-pool-follow-up-tasks/\)](https://superfastpython.com/multiprocessing-pool-follow-up-tasks/)

- [Share pool with workers \(https://superfastpython.com/multiprocessing-pool-share-with-workers/\)](https://superfastpython.com/multiprocessing-pool-share-with-workers/)
- [Show progress of tasks \(https://superfastpython.com/multiprocessing-pool-show-progress/\)](https://superfastpython.com/multiprocessing-pool-show-progress/)
- [Log from Pool \(https://superfastpython.com/multiprocessing-pool-logging/\)](https://superfastpython.com/multiprocessing-pool-logging/)
- [Num remaining tasks \(https://superfastpython.com/multiprocessing-pool-remaining-tasks/\)](https://superfastpython.com/multiprocessing-pool-remaining-tasks/)

Common Usage

- [Pool Life-Cycle \(https://superfastpython.com/multiprocessing-pool-life-cycle/\)](https://superfastpython.com/multiprocessing-pool-life-cycle/)
- [Usage Patterns \(https://superfastpython.com/multiprocessing-pool-usage-patterns/\)](https://superfastpython.com/multiprocessing-pool-usage-patterns/)
- [Best Practices \(https://superfastpython.com/multiprocessing-pool-best-practices/\)](https://superfastpython.com/multiprocessing-pool-best-practices/)
- [Common Errors \(https://superfastpython.com/multiprocessing-pool-common-errors/\)](https://superfastpython.com/multiprocessing-pool-common-errors/)
- [Parallel for-loop \(https://superfastpython.com/multiprocessing-pool-for-loop/\)](https://superfastpython.com/multiprocessing-pool-for-loop/)

Use with Primitives

- [Use semaphore in pool \(https://superfastpython.com/multiprocessing-pool-semaphore/\)](https://superfastpython.com/multiprocessing-pool-semaphore/)
- [Use lock in pool \(https://superfastpython.com/multiprocessing-pool-mutex-lock/\)](https://superfastpython.com/multiprocessing-pool-mutex-lock/)
- [Use event in pool \(https://superfastpython.com/multiprocessing-pool-event/\)](https://superfastpython.com/multiprocessing-pool-event/)
- [Use condition in pool \(https://superfastpython.com/multiprocessing-pool-condition-variable/\)](https://superfastpython.com/multiprocessing-pool-condition-variable/)
- [Use barrier in pool \(https://superfastpython.com/multiprocessing-pool-barrier/\)](https://superfastpython.com/multiprocessing-pool-barrier/)

Issue Tasks

- [Issue tasks with apply \(https://superfastpython.com/multiprocessing-pool-apply/\)](https://superfastpython.com/multiprocessing-pool-apply/)
- [Issue tasks with apply_async \(https://superfastpython.com/multiprocessing-pool-apply_async/\)](https://superfastpython.com/multiprocessing-pool-apply_async/)
- [Issue tasks with map \(https://superfastpython.com/multiprocessing-pool-map/\)](https://superfastpython.com/multiprocessing-pool-map/)
- [Issue tasks with map_async \(https://superfastpython.com/multiprocessing-pool-map_async/\)](https://superfastpython.com/multiprocessing-pool-map_async/)
- [Issue tasks with imap \(https://superfastpython.com/multiprocessing-pool-imap/\)](https://superfastpython.com/multiprocessing-pool-imap/)

- [Issue tasks with imap_unordered](https://superfastpython.com/multiprocessing-pool-
imap_unordered/) ([https://superfastpython.com/multiprocessing-pool-
imap_unordered/](https://superfastpython.com/multiprocessing-pool-
imap_unordered/))
- [Issue tasks with starmap](https://superfastpython.com/multiprocessing-pool-starmap/) (<https://superfastpython.com/multiprocessing-pool-starmap/>)
- [Issue tasks with starmap_async](https://superfastpython.com/multiprocessing-pool-
starmap_async/) ([https://superfastpython.com/multiprocessing-pool-
starmap_async/](https://superfastpython.com/multiprocessing-pool-
starmap_async/))
- [How to choose method](https://superfastpython.com/multiprocessing-pool-issue-tasks/) (<https://superfastpython.com/multiprocessing-pool-issue-tasks/>)
- [How to use AsyncResult](https://superfastpython.com/multiprocessing-pool-asyncresult/) (<https://superfastpython.com/multiprocessing-pool-asyncresult/>)
- [map\(\) multiple arguments](https://superfastpython.com/multiprocessing-pool-map-
multiple-arguments/) ([https://superfastpython.com/multiprocessing-pool-map-
multiple-arguments/](https://superfastpython.com/multiprocessing-pool-map-
multiple-arguments/))
- [map\(\) chunksize](https://superfastpython.com/multiprocessing-pool-map-chunksize/) (<https://superfastpython.com/multiprocessing-pool-map-chunksize/>)

Callback Functions

- [Callback functions](https://superfastpython.com/multiprocessing-pool-callback/) (<https://superfastpython.com/multiprocessing-pool-callback/>)
- [Error callback functions](https://superfastpython.com/multiprocessing-pool-error-
callback-functions-in-python/) ([https://superfastpython.com/multiprocessing-pool-error-
callback-functions-in-python/](https://superfastpython.com/multiprocessing-pool-error-
callback-functions-in-python/))

Comparisons

- [Pool vs Process](https://superfastpython.com/multiprocessing-pool-vs-process/) (<https://superfastpython.com/multiprocessing-pool-vs-process/>)
- [Pool vs ProcessPoolExecutor](https://superfastpython.com/multiprocessing-pool-vs-
processpoolexecutor/) ([https://superfastpython.com/multiprocessing-pool-vs-
processpoolexecutor/](https://superfastpython.com/multiprocessing-pool-vs-
processpoolexecutor/))
- [Pool vs GIL](https://superfastpython.com/multiprocessing-pool-gil/) (<https://superfastpython.com/multiprocessing-pool-gil/>)

Concurrent File I/O Learning Path

Working with files can be slow.

This section provides a learning path that you can use to get started and get effective at using Python concurrency to speed-up file I/O operations, super fast.

- **Step 1: Discover Python concurrency techniques.**

- [Python Concurrency: A Whirlwind Tour](https://superfastpython.com/python-concurrency-whirlwind-tour/) (<https://superfastpython.com/python-concurrency-whirlwind-tour/>)
- **Step 2: Discover Python file I/O operations.**
 - [Python File IO: A Whirlwind Tour](https://superfastpython.com/python-file-io-tour/) (<https://superfastpython.com/python-file-io-tour/>)
 - [AIOFiles for AsyncIO in Python](https://superfastpython.com/aiofiles-for-asyncio-in-python/) (<https://superfastpython.com/aiofiles-for-asyncio-in-python/>)
- **Step 3: Discover how to perform file I/O concurrently.**
 - [Multithreaded File Saving in Python](https://superfastpython.com/multithreaded-file-saving/) (<https://superfastpython.com/multithreaded-file-saving/>)
 - [Multithreaded File Loading in Python](https://superfastpython.com/multithreaded-file-loading/) (<https://superfastpython.com/multithreaded-file-loading/>)
 - [Multithreaded File Deletion in Python](https://superfastpython.com/multithreaded-file-deletion/) (<https://superfastpython.com/multithreaded-file-deletion/>)
 - [Multithreaded File Copying in Python](https://superfastpython.com/multithreaded-file-copying/) (<https://superfastpython.com/multithreaded-file-copying/>)
 - [Multithreaded File Moving in Python](https://superfastpython.com/multithreaded-file-moving/) (<https://superfastpython.com/multithreaded-file-moving/>)
 - [Multithreaded File Renaming in Python](https://superfastpython.com/multithreaded-file-rename/) (<https://superfastpython.com/multithreaded-file-rename/>)
 - [Multithreaded File Appending in Python](https://superfastpython.com/multithreaded-file-append/) (<https://superfastpython.com/multithreaded-file-append/>)
 - [Multithreaded File Unzipping in Python](https://superfastpython.com/multithreaded-unzip-files/) (<https://superfastpython.com/multithreaded-unzip-files/>)
 - [Multithreaded File Zipping in Python](https://superfastpython.com/multithreaded-zip-files/) (<https://superfastpython.com/multithreaded-zip-files/>)

Next, you can [browse all tutorials](https://superfastpython.com/category/concurrent-fileio/) (<https://superfastpython.com/category/concurrent-fileio/>) on this topic.

Guides

Sometimes an extensive guide on a specific topic is required.

This section lists large guides on specific topics that may help.

Complete Guides

- [Threading in Python: The Complete Guide](https://superfastpython.com/threading-in-python/) (<https://superfastpython.com/threading-in-python/>)
- [Multiprocessing in Python: The Complete Guide](https://superfastpython.com/multiprocessing-in-python/) (<https://superfastpython.com/multiprocessing-in-python/>)
- [Asyncio in Python: The Complete Guide](https://superfastpython.com/python-asyncio/) (<https://superfastpython.com/python-asyncio/>)
- [ThreadPoolExecutor in Python: The Complete Guide](https://superfastpython.com/threadpoolexecutor-in-python/) (<https://superfastpython.com/threadpoolexecutor-in-python/>)
- [ProcessPoolExecutor in Python: The Complete Guide](https://superfastpython.com/processpoolexecutor-in-python/) (<https://superfastpython.com/processpoolexecutor-in-python/>)
- [ThreadPool in Python: The Complete Guide](https://superfastpython.com/threadpool-python/) (<https://superfastpython.com/threadpool-python/>)
- [Multiprocessing Pool in Python: The Complete Guide](https://superfastpython.com/multiprocessing-pool-python/) (<https://superfastpython.com/multiprocessing-pool-python/>)

Other Guides

- [Why Learn Python Concurrency](https://superfastpython.com/why-learn-python-concurrency/) (<https://superfastpython.com/why-learn-python-concurrency/>)
- [Python Concurrency Learning Paths](https://superfastpython.com/learning-paths/) (<https://superfastpython.com/learning-paths/>)
- [Choose a Python Concurrency API](https://superfastpython.com/python-concurrency-choose-api/) (<https://superfastpython.com/python-concurrency-choose-api/>)
- [Python Concurrency Books](https://superfastpython.com/python-concurrency-books/) (<https://superfastpython.com/python-concurrency-books/>)
- [Python Concurrency Glossary](https://superfastpython.com/python-concurrency-glossary/) (<https://superfastpython.com/python-concurrency-glossary/>)

Need More Help?

Do you need more help with Python concurrency?

Perhaps you need help with a topic not covered above?

- [Contact me directly](https://superfastpython.com/contact/) (<https://superfastpython.com/contact/>) and I will do my best to help.

COPYRIGHT © 2022 SUPER FAST PYTHON

LINKEDIN ([HTTPS://WWW.LINKEDIN.COM/COMPANY/SUPER-FAST-PYTHON/](https://www.linkedin.com/company/super-fast-python/)) | TWITTER ([HTTPS://TWITTER.COM/SUPERFASTPYTHON](https://twitter.com/superfastpython)) |

FACEBOOK ([HTTPS://WWW.FACEBOOK.COM/SUPERFASTPYTHON](https://www.facebook.com/superfastpython)) | RSS ([HTTPS://SUPERFASTPYTHON.COM/FEED/](https://superfastpython.com/feed/)).