

Navigation

- [index](#)
- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »
- [The Python Standard Library](#) »
- [9. Data Types](#) »

9.9. mutex — Mutual exclusion support¶

Deprecated since version The: `mutex` module has been removed in Python 3.0.

The `mutex` module defines a class that allows mutual-exclusion via acquiring and releasing locks. It does not require (or imply) [threading](#) or multi-tasking, though it could be useful for those purposes.

The `mutex` module defines the following class:

```
class mutex.mutex¶
```

Create a new (unlocked) mutex.

A mutex has two pieces of state — a “locked” bit and a queue. When the mutex is not locked, the queue is empty. Otherwise, the queue contains zero or more `(function, argument)` pairs representing functions (or methods) waiting to acquire the lock. When the mutex is unlocked while the queue is not empty, the first queue entry is removed and its `function(argument)` pair called, implying it now has the lock.

Of course, no multi-threading is implied – hence the funny interface for [lock\(\)](#), where a function is called once the lock is acquired.

9.9.1. Mutex Objects¶

[mutex](#) objects have following methods:

```
mutex.test()¶
```

Check whether the mutex is locked.

```
mutex.testandset()¶
```

“Atomic” test-and-set, grab the lock if it is not set, and return `True`, otherwise, return `False`.

```
mutex.lock(function, argument)¶
```

Execute `function(argument)`, unless the mutex is locked. In the case it is locked, place the function and argument on the queue. See [unlock\(\)](#) for explanation of when `function(argument)` is executed in that case.

```
mutex.unlock()¶
```

Unlock the mutex if queue is empty, otherwise execute the first element in the queue.

[Table Of Contents](#)

[9.9. mutex — Mutual exclusion support](#)

- [9.9.1. Mutex Objects](#)

Previous topic

[9.8. sched — Event scheduler](#)

Next topic

[9.10. queue — A synchronized queue class](#)

This Page

- [Show Source](#)

Navigation

- [index](#)
- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »

- [The Python Standard Library](#) »
- [9. Data Types](#) »

© [Copyright](#) 1990-2010, Python Software Foundation.

The Python Software Foundation is a non-profit corporation. [Please donate.](#)

Last updated on Feb 26, 2010. Created using [Sphinx](#) 0.6.3.