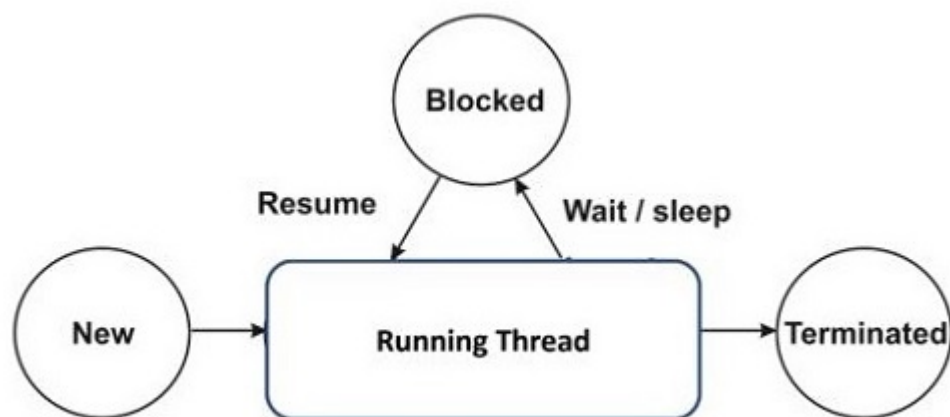


Python - Ciclo de vida do thread

Um objeto thread passa por diferentes estágios. Quando um novo objeto thread é criado, ele deve ser iniciado. Isso chama o método `run()` da classe thread. Este método contém a lógica do processo a ser executado pela nova thread. O thread conclui sua tarefa quando o método `run()` termina e o thread recém-criado se funde com o thread principal.

Enquanto um thread está em execução, ele pode ser pausado por um período predefinido ou pode ser solicitado que seja pausado até que um determinado evento ocorra. O encadeamento é retomado após o intervalo especificado ou o processo termina.



A biblioteca padrão do Python possui dois módulos, `"_thread"` e `"threading"`, que incluem a funcionalidade para lidar com threads. O módulo `"_thread"` é uma API de baixo nível. No Python 3, foi incluído o **módulo `threading`**, que fornece funcionalidades mais abrangentes para gerenciamento de threads.

Python O Módulo `_thread`

O módulo **`_thread`** (módulo **`thread`** anterior) faz parte da biblioteca padrão do Python desde a versão 2. É uma API de baixo nível para gerenciamento de threads e funciona como suporte para muitos dos outros módulos com recursos avançados de execução simultânea, como `threading` e `multiprocessing`.

Python - o módulo de `threading`

O módulo de `threading` mais recente fornece suporte de alto nível muito mais poderoso para gerenciamento de threads.

A classe `Thread` representa uma atividade executada em um thread de controle separado. Existem duas maneiras de especificar a atividade: passando um objeto que pode ser chamado ao construtor ou substituindo o método `run()` em uma subclasse.

```
threading.Thread(target, name, args, kwarg, daemon)
```

Parâmetros

- **target** - função a ser invocada quando um novo thread é iniciado. O padrão é None, o que significa que nada é chamado.
- **name** - é o nome do thread. Por padrão, um nome exclusivo é construído como "Thread-N".
- **daemon** - Se definido como True, o novo thread é executado em segundo plano.
- **args e kwargs** - argumentos opcionais a serem passados para a função de destino.