

Interrumpir un proceso en Python

[crayon-6a9d6cb36e2f7624222945/]

A terminal window titled 'lamac — -zsh — 80x24' showing a Python script execution. The script prints five lines of 'Contando: ', 0 through 4. After the fifth line, it prints 'El hilo ha sido interrumpido' and returns to the prompt. The terminal has a dark background with light gray text.

```
lamac@lamacs-MacBook-Air-2 ~ % python interrup.py
('Contando: ', 0)
('Contando: ', 1)
('Contando: ', 2)
('Contando: ', 3)
('Contando: ', 4)
El hilo ha sido interrumpido
lamac@lamacs-MacBook-Air-2 ~ %
```

En Python la detención de un proceso se realiza mediante una bandera booleana `self._stop_event` para indicar al subprocesso que debe detenerse. Esta bandera se verifica periódicamente dentro del método `run()` del subprocesso y se establece como `True` mediante un método personalizado, como `stop()`, que se define en la clase.

En este ejemplo, se define la clase `MyThread` que hereda de `Thread`. En el constructor de la clase, se inicializa la bandera booleana `_stop_event` como un objeto `Event` de `threading`.

Dentro del método `run()`, se utiliza un ciclo `while` para contar indefinidamente y se verifica continuamente si la bandera

`_stop_event` está establecida. Si la bandera es verdadera, el ciclo `while` se detiene y el mensaje «El hilo ha sido interrumpido» se imprime en la consola.

El método `stop()` se define en la clase para establecer la bandera `_stop_event` como `True`.

En el bloque `if __name__ == "__main__":`, se crea una instancia de `MyThread` y se inicia. Después de esperar 5 segundos, se llama al método `stop()` para interrumpir el subproceso, y finalmente se llama al método `join()` para esperar a que el subproceso termine antes de que el programa principal finalice.