1. Багатопочність у пайтоні. Обмеження Global Interpreter Lock(GIL)

Багатопоточність – це робота декількох модулів одночасно, але не заважаючи один одному. Можуть обмінюватися файлами. Можемо призупиняти виконання потоку, така властивість називається поступанням.

Оголошується початок нового потоку наступним чином.

thread.start\_new\_thread( function, args[, kwargs] )

**Глобальне блокування інтерпретатора**  — це механізм [інтерпретатора](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80), що гарантує те, що в кожен момент часу виконується код лише однієї [ниті](https://uk.wikipedia.org/wiki/%D0%9D%D0%B8%D1%82%D1%8C), для уникнення конкурентного доступу до спільних структур даних. В мовах, що реалізують GIL, завжди існує один GIL для кожного [процесу](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81_(%D1%96%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) інтерпретатора. Прикладом реалізацій інтерпретаторів, які використовують GIL є [CPython](https://uk.wikipedia.org/wiki/CPython" \o ").

Обмеження:

* Проблеми до можливості виконання множини потоків в програмі, через обмеження ЦПУ.
* GIL впливаэ лише на обмежену частину усієї екосистеми Pуthon, і може сприйматися як зло.

1. Оператор lambda – це аноніма, або незв’язна функція, при цьому досить обмежена. Використовуються зазвичай для простішого й легшого написання коду. Основні відмінності від def f(), це відсутність імені і обмеження на тіло функції на один «вислів».

Обмеження:

**def** func(x, y):

**return** x\*\*2 + y\*\*2

func = **lambda** x, y: x\*\*2 + y\*\*2