В синхронном коде каждая операция ожидает окончание предыдущей, по этому программа может не отвечать пока выполняется какая то большая операция.

**Блокирующие операции** – операции которые мешают выполнению других операций.

Например:

* Если браузер будет работать в 1 потоке, то загрузка файла будет блокировать выполнение других функций браузера.
* I/O – ввод/вывод – взаимодействие с “Миром”, пример ниже.
  + conn\_data = conn.read(1024) # network
  + file\_data = fileobj.read(1024) # filesystem

Решение проблем блокировки:

* **Процесс** - отдельная **подпрограмма**(**процесс**) от текущей программы. Требует большие затраты на создание, требует очень много памяти, сложно связать с основной программой, оверхед со стороны **ОС** (**user-space to scheduling, context switching**)
* **Поток** - подпрограмма, которая выполняется внутри текущей(в текущем процессе). Требует большие затраты на создание(меньше по сравнению с процессами). При работе в высоконагруженных систем выполнения могут произойти например **race conditional, dedlock, starvation**, да еще и **GIL**. Если не ограничить **Pool Threads**, программа встанет из за переполнения файлового дескриптора или ограничений системы.
* **Асинхронность** – когда программа работает используя **Event Loop** (цикл событий), где **Event** это подпрограммы (**coroutine**), а **Loop** место где происходит их хранение и выполнение путем прохода по всем событиям.