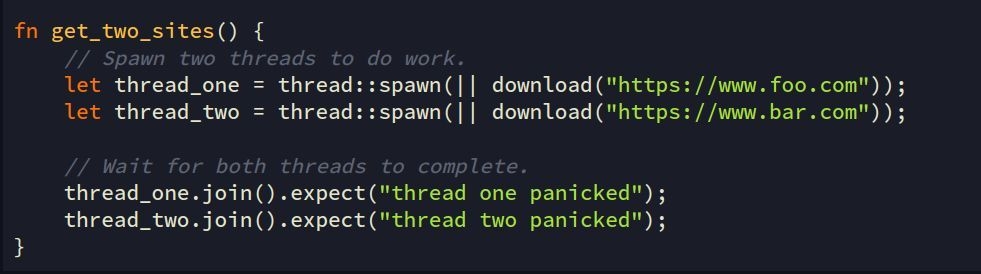
**INTRODUCTION TO ASYNC PROGRAMMING**

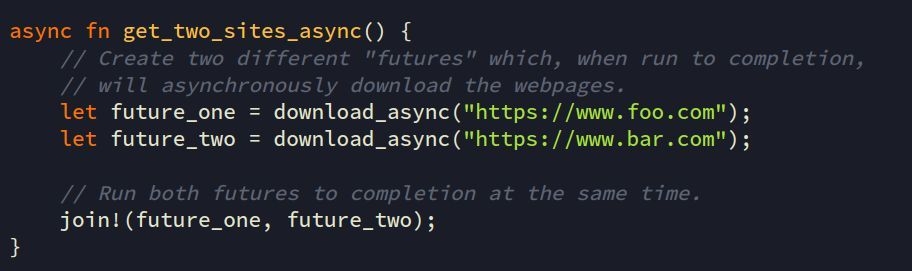
Async (or asynchronous) programming is a proficiency to run multiple operations in our applications inside a single thread. Whatever the nature of your application a web server, a database or an operating system, using async programing you can get the most out of the underlying hardware.

In Rust, when we talk about async, we’re talking about running code concurrently, or having multiple operations running on a single thread. Multi-threading is a related, but a different concept. Multithreading is ideal for when you’ve got computationally intensive tasks that can be spread across multiple cores

In a multi-threaded application, if you wanted to download two different webpages at the same time, you would spread the work across two different threads. For example:

Since threads are designed to run multiple tasks at a same time, they still have some limitations which are mentioned below.

* There's a lot of difficulties in switching between different threads and sharing data between them.
* Even a thread which just sits and does nothing also use valuable system resources.

These are the overheads that asynchronous Rust is designed to reduce. We can rewrite the above function using Rust's async notation. Which will allow us to run multiple tasks at a time without generating multiple threads. For example: