**Asynchronous vs. Synchronous Programming**

In **synchronous** operations tasks are performed**one at a time**and only when one is completed, the following is unblocked. In other words, you need to wait for a task to finish to move to the next one.

In **asynchronous** operations, on the other hand, you can move to another task before the previous one finishes. This way, with asynchronous programming you’re able to **deal with multiple requests simultaneously**, thus completing more tasks in a much shorter period of time.

**Return types of the Asynchronous methods**

In asynchronous programming we have three return types:

* Task<TResult>, for an async method that returns a value
* Task, to use it for an async method that does not return a value
* void, which we can use for an event handler

**Task<IEnumerable<T>> VS IAsyncEnumerable<T>**

**Task<IEnumerable<T>>** provides records once the data in collection is ready to send to the caller.

**IAsyncEnumerable<T>** provides records as they are ready, which means it will send you record as they are available rather than waiting for the whole collection to be filled up.