

Async Parallel

```
static Task ParallelForEachAsync<T> (IEnumerable<T> source,
                                     int degreeofParallelization,
                                     Func<T, Task> body)
{
    async Task AwaitPartition(IEnumerator<T> partition)
    {
        using (partition)
        {
            while (partition.MoveNext())
            {
                await body(partition.Current);
            }
        }
    }

    return Task.WhenAll(Partitioner
                        .Create(source)
                        .GetPartitions(degreeofParallelization)
                        .AsParallel()
                        .Select(AwaitPartition));
}
```

Usage:

```
var tasks = Enumerable.Range(0, 10000).Select(_ => new Func<Task>(() =>
Worker(results))).ToList();

await ParallelForEachAsync(tasks, 1000, async func =>
{
    await func();
});
```

https://www.youtube.com/embed/IHuyI_WTpME

Revision #9
Created 2022-09-15 17:57:20 UTC by Guest
Updated 2024-11-19 23:44:52 UTC by Admin