



C# Corner



# Asynchronous and Parallel

C# Corner

Login

Post

Ask Question



Anil Kumar

Updated date Feb 12, 2015

43k

7

5

[Download Free .NET & JAVA Files API](#)[Try Free File Format APIs for Word/Excel/PDF](#)

Asynchronous programming is not something you spend much time thinking about. Yes, believe me. Just start analysing the requirements in the developer's eye, keeping in mind that you want to develop a resource and time efficient application.

## Laying the groundwork, In the Developer's Eyes

**Scenario 1:** As a developer, I want to continue the execution of code to the next line instead of getting stuck or waiting for the completion/result of long-running code (for example network or database operations).

Example case: A long-running execution happens and instead of waiting for the completion/result, we want to continue further without wasting time.

**Scenario 2:** As a developer, I want to finish a task, time-efficiently and utilizing other available cores of CPUs or machines by dividing the task into independent chunks.

being used to execute a task. Here, we want to utilize the other cores of CPU by distributing the job.

## Mechanism to do the preceding

Well, now we have 2 valid ideas to boost the performance of an application. For the implementation of these ideas we need some special mechanism in coding. And these mechanisms are known as asynchronous and parallel programming respectively. :) So far easy? Good. :)

Asynchronously means non-blocking, in other words you can proceed without waiting on a

when the second line is a time-consuming method call (maybe a database call). Now, you want to apply some mechanism in your code to execute the second line without waiting for the completion or result of your first line. This is the non-blocking style.

Getting to the **second scenario**, just think that if you can break your task into independent pieces of work, then what? Would you not like to get these independent pieces of work done in parallel to finish faster? This is called parallelism. In this case there must be available free resources to do the job in parallel. (A resource could be cores of a CPU or even multiple machines.)

Ahh! You must clearly understand the requirements of these 2 mechanisms in programming. So far we have concluded that these two approaches enable us to finish our task faster (utilizing the wait-time and free-resource).

### Let's clarify the things with more technical terms :)

If we say that asynchronous provides no wait time and continuous execution of the code then this line itself clarifies that we are talking about single-threaded processing. It also makes clear that it is all about a "concurrent work" fashion.

**Concurrent?** In a single-threaded process, at the same time, many lines of code can be executed concurrently. To do this, Async and Callbacks mechanisms are used. It involves heavy Context Switching and Time Scheduling.

**\* You can conclude at this point that:** "An Asynchronous Task might use Parallelism".

In the case of parallel programming, there must be multiple threads, otherwise how can that

**Asynchronous:** Hey you, do this task and return with the results. Meanwhile I will get done other tasks. (No wasting of time. :))

**Parallel:** Hey you, do this task and return with the results. Meanwhile I will do nothing but wait for you. You can smartly use other free resources/cores of the CPUs/machines to boost the work time.

[Post](#)
[Ask Question](#)
[Asynchronous](#)
[irrespective of programming](#)
[Multithreading](#)
[Parallel](#)

Next Recommended Article

[Parallel Programming Part 1: Introducing Task Programming Library](#)

## OUR BOOKS



Anil Kumar *TOP 500*

Anil Kumar is a passionate web developer, previous business owner, social executive, and a father. He has 10+ years of exposure in diverse technologies at variant roles. Anil is much inspired with Mr. Mahesh of c-sharpCo... [Read more](#)

<http://CodePattern.net>

5

7



Type your comment here and press Enter Key (Minimum 10 characters)



Explained very clearly. Thanks for the article and effort!

[dkfjksdjn kjfnskdj](#)

1961 19 0

Oct 24, 2020

0 0 Reply

Very nice article. Thank you :)

[kanika luthra](#)

Aug 23, 2020


[:come a member](#)
[Login](#)


Glad to know you like it. tn timer

[Anil Kumar](#)

200 10.9k 1.4m

Post

Ask Question

Aug 23, 2018

0



Nice article. Thanks for sharing

[Santhakumar Munuswamy](#)

37 34.4k 1.2m

Jul 08, 2015

0

1 Reply



Thank you [Santhakumar Munuswamy](#)

[Anil Kumar](#)

200 10.9k 1.4m

Oct 23, 2015

0



Very nice Anil its fluent explanation.....Like it

[Pramod Lawate](#)

1515 466 186.3k

Feb 12, 2015

0

1 Reply



Thanks for your nice words!

[Anil Kumar](#)

200 10.9k 1.4m

Feb 12, 2015

0

## FEATURED ARTICLES

Getting Started Wth MongoDB Atlas

What Is a Full Stack Developer


C# 9 Cheat Sheet

❖ Clean Architecture End To End In .NET 5

Flutter Vs React Native Which is Best

[View All](#)

[:come a member](#)[Login](#)[Post](#)[Ask Question](#)

- 01  Clean Architecture End To End In .NET 5
- 02 Bridge Design Pattern With Java
- 03 Getting Started With Azure Service Bus Queues And ASP.NET Core - Part 1
- 04 How To Add A Document Viewer In Angular 10
- 05 Flutter Vs React Native - Best Choice To Build Mobile App In 2021
- 06 C# 9 Cheat Sheet
- 07 Deploying ASP.NET and DotVVM web applications on Azure
- 08 Integrate CosmosDB Server Objects with ASP.NET Core MVC App
- 09 What Is a Full Stack Developer
- 10 Getting Started With Azure Service Bus Queues And ASP.NET Core Background Services

[View All](#) 



[Come a member](#)

[Login](#)

[Post](#)

[Ask Question](#)

[About Us](#) [Contact Us](#) [Privacy Policy](#) [Terms](#) [Media Kit](#) [Sitemap](#) [Report a Bug](#) [FAQ](#) [Partners](#)

[C# Tutorials](#) [Common Interview Questions](#) [Stories](#) [Consultants](#) [Ideas](#) [Certifications](#)

©2020 C# Corner. All contents are copyright of their authors.