Armen Melkumyan • 1st
Technical / Solutions Architect
1yr • 🌐

🔍💻 From .Net C# Interview Questions

🏭 Exploring Task.Factory in C#:

Task.Factory: Part of the Task Parallel Library (TPL)

Provides methods for creating 🛠️ and starting 🚀 Task instances.

More configurable ⚙️ than basic Task constructors or Task.Run.

🔑 Key Features of Task.Factory:

Task Creation: Various methods for tailored task configurations.

Custom Task Schedulers: Specify a TaskScheduler for bespoke scheduling.

Task Creation Options: Supports diverse TaskCreationOptions for enhanced control.

Continuation Tasks: Easy creation of tasks that start after another task completes.

🧑‍💻 Usage Examples:

Opt for LongRunning tasks for dedicated threading.

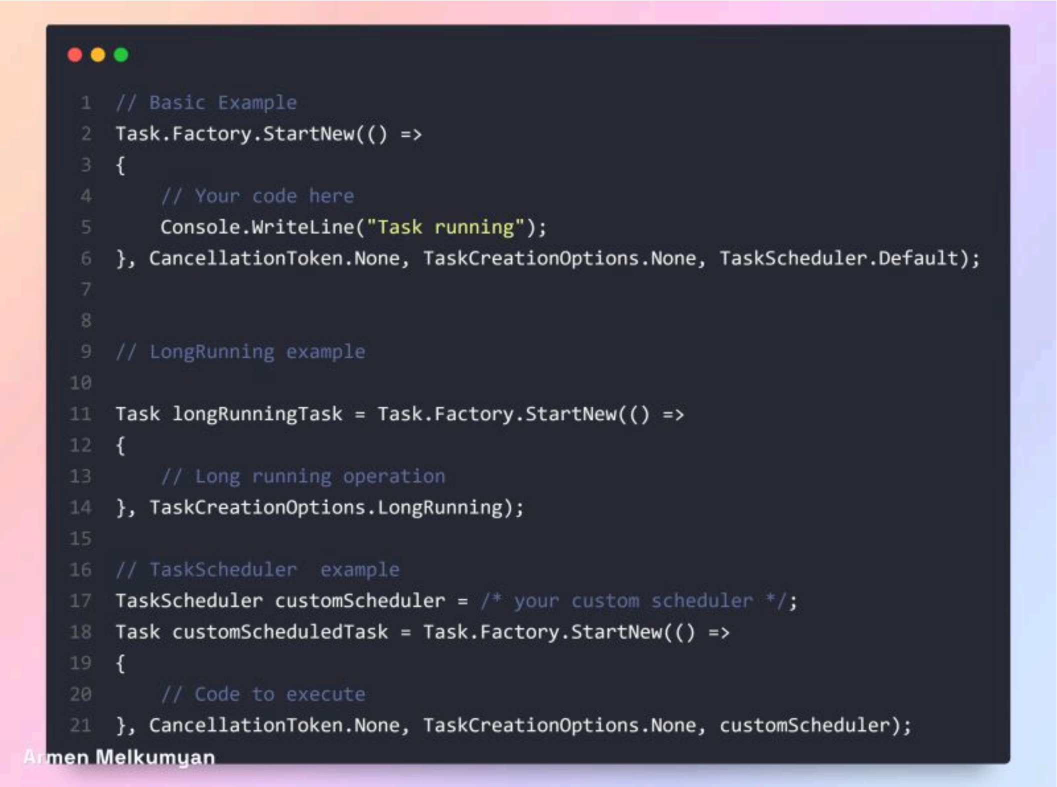
Define a custom TaskScheduler for precise execution control.

Utilize continuation tasks for sequential task flows.

#CSharpProgramming #DotNetDevelopers #TaskFactory

#ParallelComputing #SoftwareDevelopment #CodingInterviews

#TechTips #DeveloperCommunity



```csharp
// Basic Example
Task.Factory.StartNew(() =>
{
    // Your code here
    Console.WriteLine("Task running");
}, CancellationToken.None, TaskCreationOptions.None, TaskScheduler.Default);


// LongRunning example

Task longRunningTask = Task.Factory.StartNew(() =>
{
    // Long running operation
}, TaskCreationOptions.LongRunning);

// TaskScheduler  example
TaskScheduler customScheduler = /* your custom scheduler */;
Task customScheduledTask = Task.Factory.StartNew(() =>
{
    // Code to execute
}, CancellationToken.None, TaskCreationOptions.None, customScheduler);
```

18