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

From C# .Net interview questions

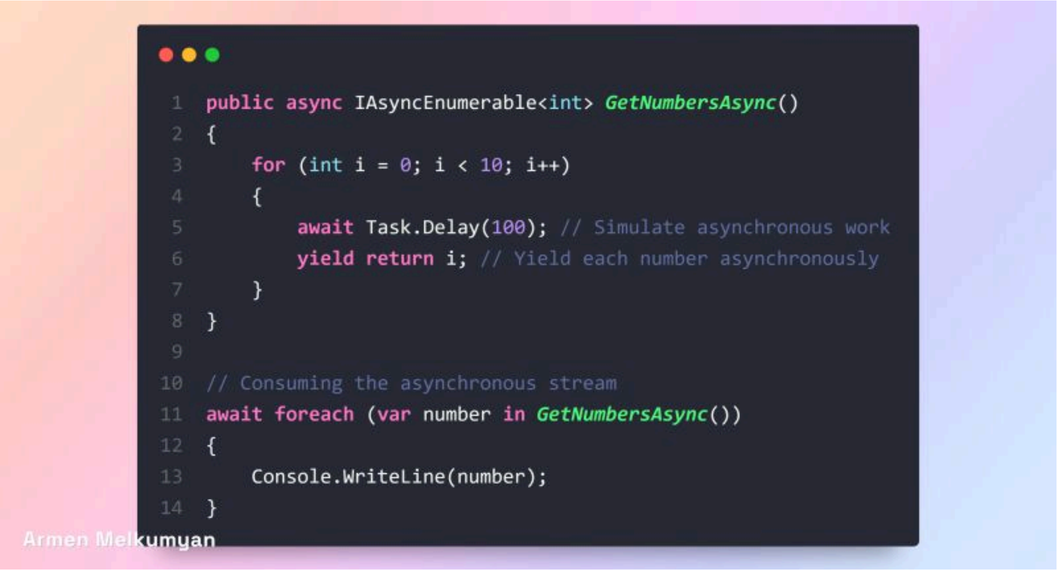
What are asynchronous streams and how do you implement them in C#?

Answer:

Asynchronous streams are a feature introduced in C# 8.0 that allows you to perform asynchronous operations while iterating over a collection. They are implemented using the IAsyncEnumerable<T> interface and the async and await keywords. You can define an asynchronous iterator method with the async keyword and use yield return to return elements one at a time. The consumer of the asynchronous stream can then use await foreach to asynchronously iterate over the elements.

In below example, GetNumbersAsync is an asynchronous iterator method that yields numbers 0 through 9, simulating an asynchronous operation before each yield. The await foreach statement is used to consume the numbers as they are produced by the asynchronous stream.

#Csharp #AsynchronousStreams #EfficientCoding



```
public async IAsyncEnumerable<int> GetNumbersAsync()
{
    for (int i = 0; i < 10; i++)
    {
        await Task.Delay(100); // Simulate asynchronous work
        yield return i; // Yield each number asynchronously
    }
}

// Consuming the asynchronous stream
await foreach (var number in GetNumbersAsync())
{
    Console.WriteLine(number);
}
```

12