What are Azure Functions?

How to create them in .NET? 👇

Azure Functions is a game-changer for those looking to streamline and scale their applications.

Here are the key highlights of why it's become a go-to for developers in the cloud era:

Serverless Flexibility: Azure Functions handle infrastructure management, so you can focus solely on writing code. Just deploy your function, and Azure manages the rest, including scaling based on demand.

Trigger-Driven Execution: Functions can be triggered by various events - HTTP requests, queue messages, database changes, and more. This event-driven model makes Azure Functions perfect for microservices, automation tasks, and real-time processing.

Seamless Integration with Azure Ecosystem: With built-in support for Cosmos DB, Event Hubs, Blob Storage, and other Azure services, connecting and expanding your cloud environment is straightforward.

Dynamic Scaling: Azure Functions automatically scale up or down based on traffic. This elasticity keeps costs down by charging you only when your code runs - ideal for unpredictable workloads.

Language & **Framework Choice**: Write functions in C#, JavaScript, Python, Java, or PowerShell. Plus, with .NET 8, Azure Functions now leverages the isolated worker process model for enhanced dependency management and ASP .NET Core compatibility.

Whether you're building APIs, automating workflows, or processing real-time data, Azure Functions offers a powerful, cost-effective solution.

Here is the simple process of creating an Azure Function in .NET.

Be part of the group of 16k+ engineers in **TheCodeMan.net** and get knowledge and news from the .NET world every week: **https://thecodeman.net**