Getting Started With Azure Functions

by Gunnar Peipman MVB · Mar. 22, 17 · Cloud Zone

Azure Functions is a serverless computing offer by Microsoft. Functions are scripts that can be written in JavaScript, C#, F#, Python, PHP, Bash, Batch, or PowerShell. Functions can be developed on Visual Studio using preview level tools or directly through the browser in the Azure portal. This blog post introduces Azure Functions and shows how to build your first function.

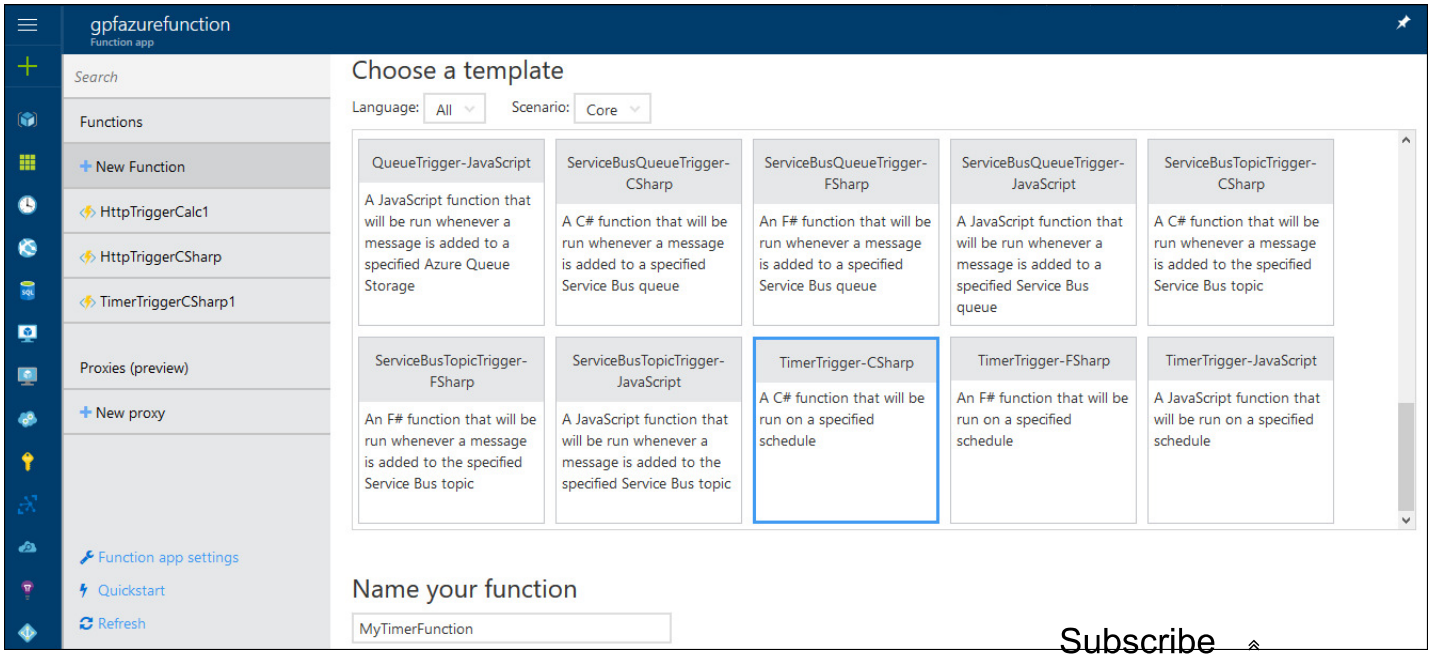
Functions are run by triggered events. When run, functions may accept input parameters and can also return output to a supported service. For example, we may have a time-triggered function that accepts Azure Table as an input and Azure Notification Hub as an output service. The function processes the latest records in Azure Table Storage and generates notifications for users.

The following table lists triggers, inputs, and outputs that Azure Functions currently supports.

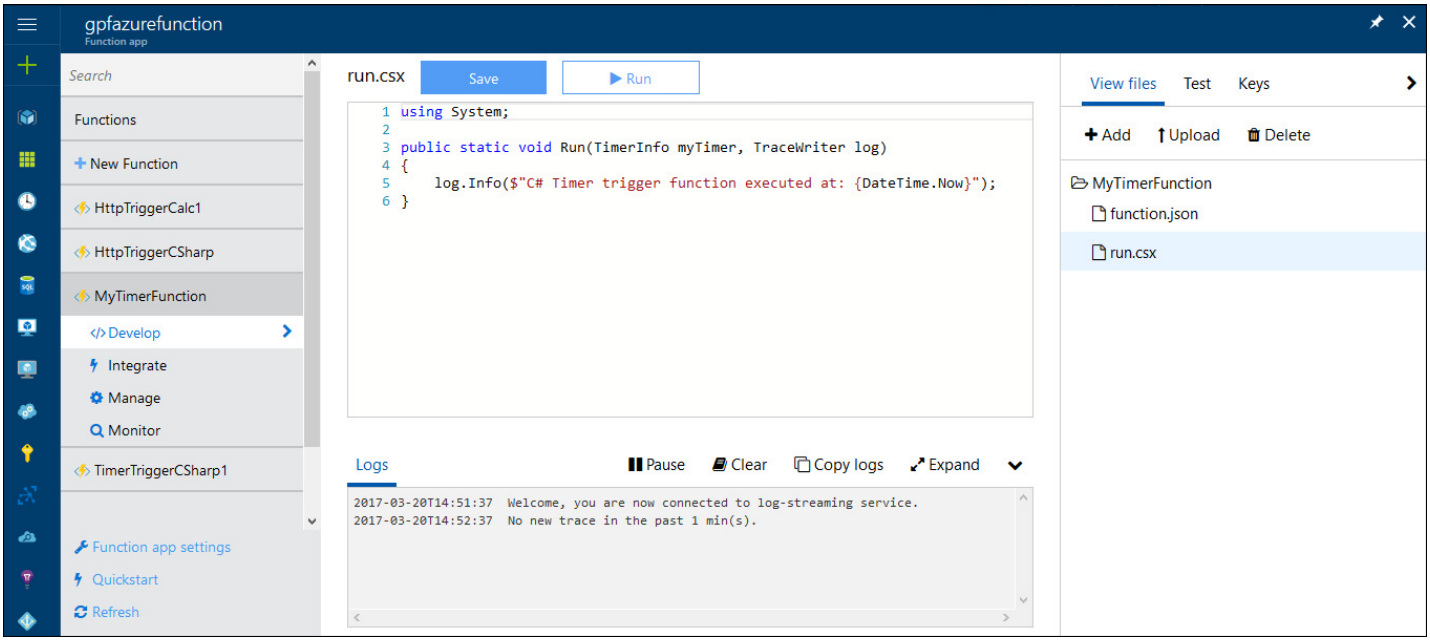
TRIGGERS	INPUTS	OUTPUTS
Blob	Azure Blob Storage	Azure Event Hub
Event hub	External File	Azure Queue Storage
External file	External Table	Azure Blob Storage
External table	Azure Table Storage	External File
Generic web hook	Azure DocumentDB Document	External Table
GitHub commenter	Azure Mobile Table Record	HTTP
GitHub web hook	Bot Framework	Azure Service Bus
Http request		Azure Table Storage
Manual trigger		Azure DocumentDB document
Queue trigger		Azure Mobile Table Record
Service Bus queue trigger		Azure Notification Hub
Service Bus topic trigger		SendGrid
Timer trigger		Twilio SMS
		Bot Framework

Building Your First Azure Function

Log into your Azure account and create a new function from the left menu. For this example, we'll choose TimerTrigger-CSharp as the function type. Give the function a name and leave the interval as it is.

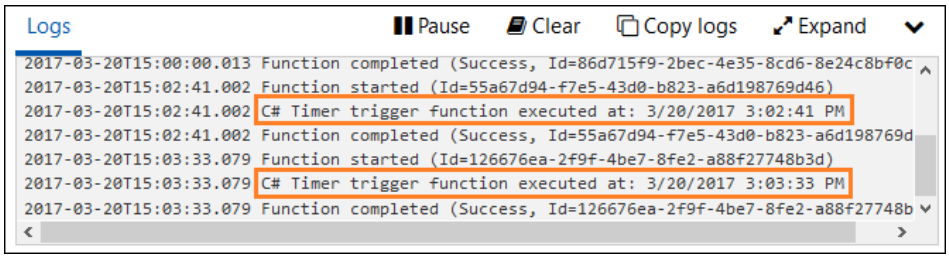


The new function is created and opened in the function editor.



Although the editor runs in a browser, it has syntax highlighting. If the function is not developed online and tooling doesn't support publishing to Azure, then users can upload function files to the server in the editor window. Also, keys needed to call functions are available here.

The default timer trigger function is primitive. It just writes out to console when it was executed. Let's click Run and see the output, which is written to the log window below the function editor.



Congrats! You just successfully ran your first Azure Function!

Visual Studio Tools for Azure Functions

Azure Functions can also be built on Visual Studio. There is a preview of Visual Studio Tools for Azure Functions available. There are some problems and limitations, as described in the .NET Web Development and Tools Blog, but still it is possible to develop functions in a familiar environment.

Visual Studio Tools comes with an Azure Functions emulator that we can use to run and debug functions on a local box. To find out more, I again refer you to the .NET Web Development and Tools Blog, as they have described the tools in-depth.

I will come back to tooling with my next posts that show what and how we can build on Azure Functions.

Wrapping Up

Azure Functions is serverless computing service by Microsoft. Functions are, usually by nature, simple scripts and can be built in the Azure portal or in Visual Studio. Visual Studio has preview-level tools for functions that we can use to build and debug functions on our development boxes. The Azure portal lets us monitor our functions and see if they have any health issues or problems. It was easy to build our first function using the browser and online function editor. We can consider Azure Functions as our choice for serverless, as it seems to work well.

Like This Article? Read More From DZone