Serverless Architecture in Cloud-based Applications

*1 What is this article about?*

The idea of this article is to give you a very understanding about Serverless architecture by using one of the top 3 cloud provider Azure. At the end of this article, you will be able to understand about the serverless architecture with the help of Azure functions.

*2 What is a serverless architecture?*

A serverless architecture is a way to build and run applications and services without having to manage infrastructure. Your application still runs on servers, but all the server management is done by Azure/AWS/GCP cloud providers. You no longer have to provision, scale, and maintain servers to run your applications, databases, and storage systems.

**2.1 Example services across different cloud providers**

* Azure Function in Azure
* AWS Lambda in AWS
* [Google Cloud Functions](https://www.techtarget.com/searchcloudcomputing/definition/Google-Cloud-Functions)

*3 Why use serverless architectures?*

By using a serverless architecture, we as developers can focus on their core product instead of worrying about managing and operating servers or runtimes, either in the cloud or on-premises. This reduced overhead lets developers reclaim time and energy that can be spent on developing great products which scale and that are reliable.

So, in a nutshell, serverless architecture is a way to build your [cloud-based application](https://relevant.software/blog/modern-cloud-based-application-development/) without managing infrastructure. Still, serverless does not mean there are *no servers* at all. The term is somewhat elusive. Servers are simply eliminated from the app development since they are managed by the vendors.

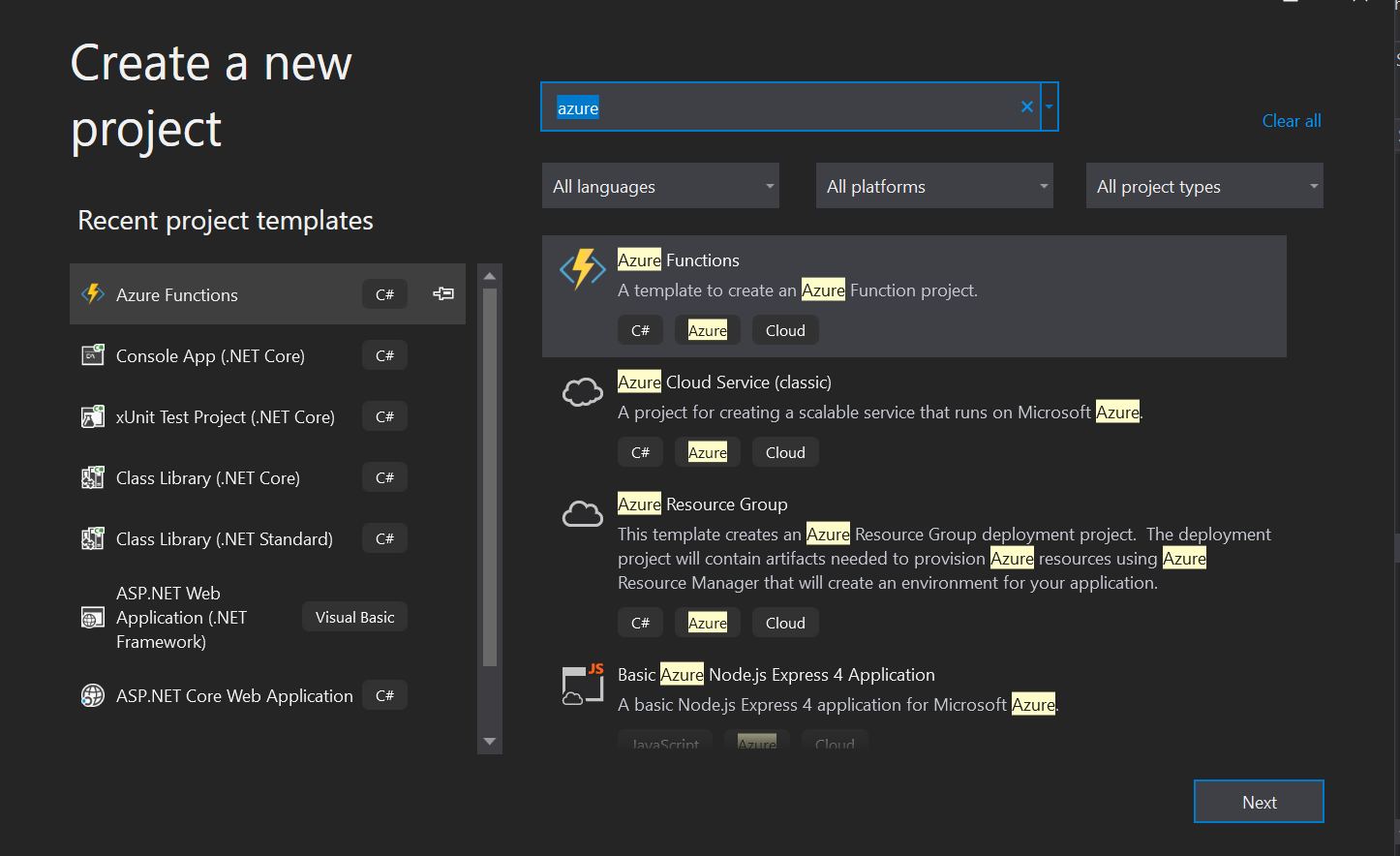
*4 How Azure Functions works*

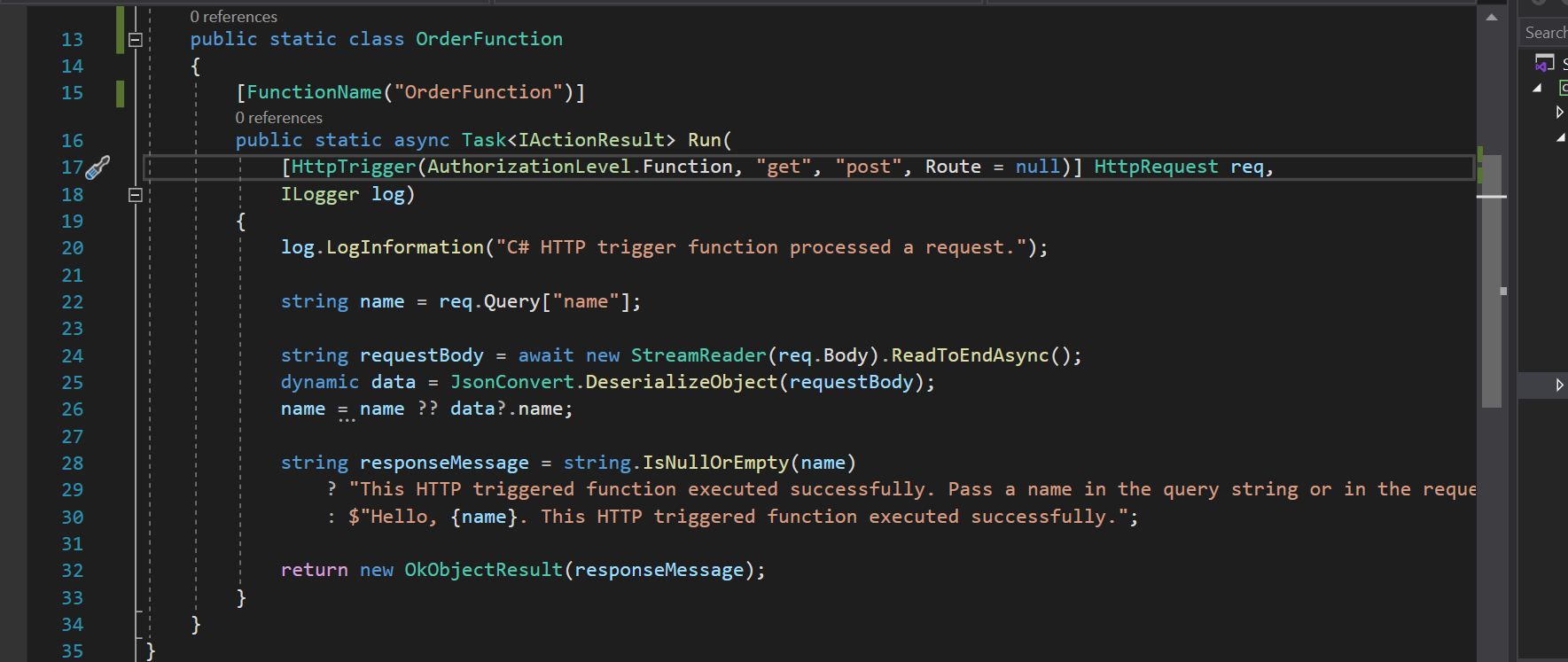
The Azure Functions serverless platform enables teams to build event-driven apps that run code when triggered by events. The platform automatically manages all the computing resources required in those processes.

*5 Create Azure Functions with different methods*

*5.1 Visual studio*

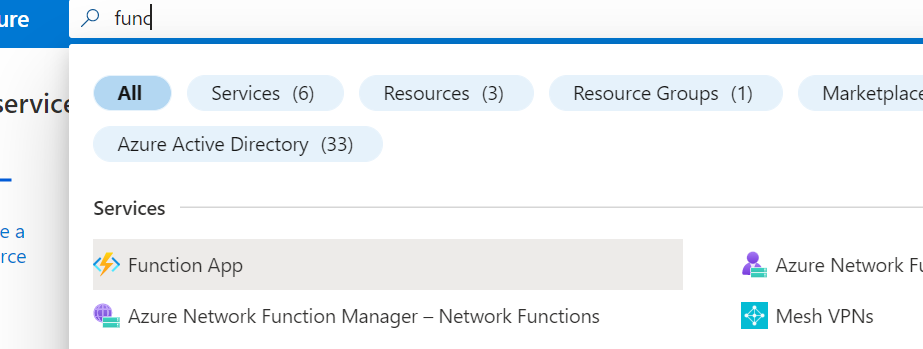
* Create new project
* Search Azure function and create a new project with required information.
* Right click on the function app project and choose publish option to Azure and then we need to provide the valid subscription and then storage account details (Storage account is mandatory when we create or deploy function app)
* Once done everything it will deploy into azure

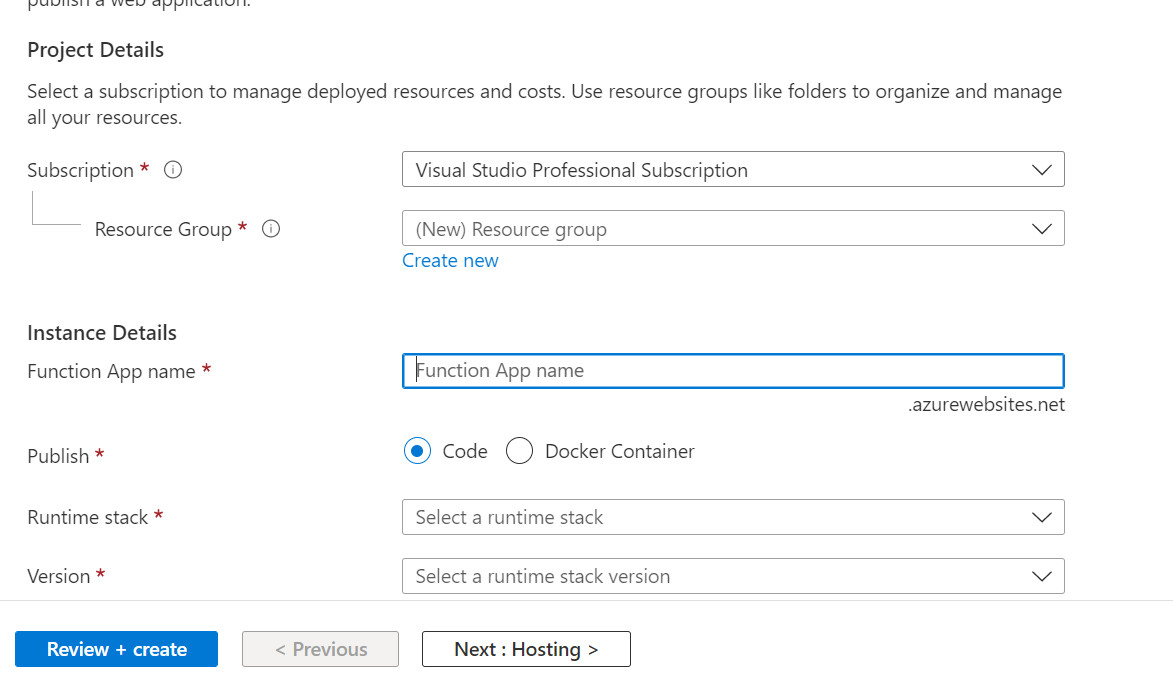




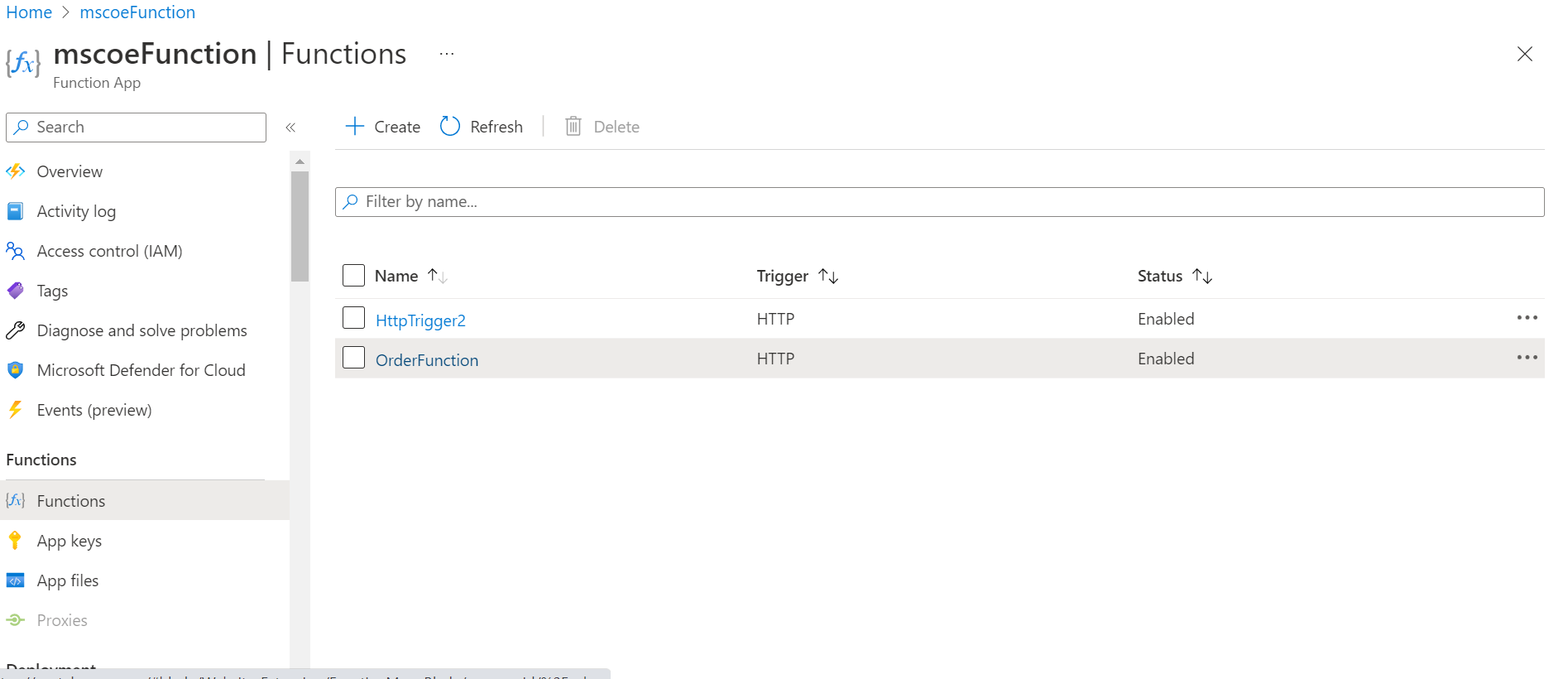
*5.2 Azure portal:*

* Search function app in the portal and we need to click function app in searched item





* We need to provide the valid subscription Function app name, Resource group and then then storage account details (Storage account is mandatory when we create or deploy function app) etc.,
* Finally review and create it.
* Once we can create many functions inside the function app with different triggers



*5.3 Provisioning or managing Azure Functions Using Terraform (IAC)*

Please refer the following link

https://www.hildeberto.com/2021/03/terraform-azure-function.html

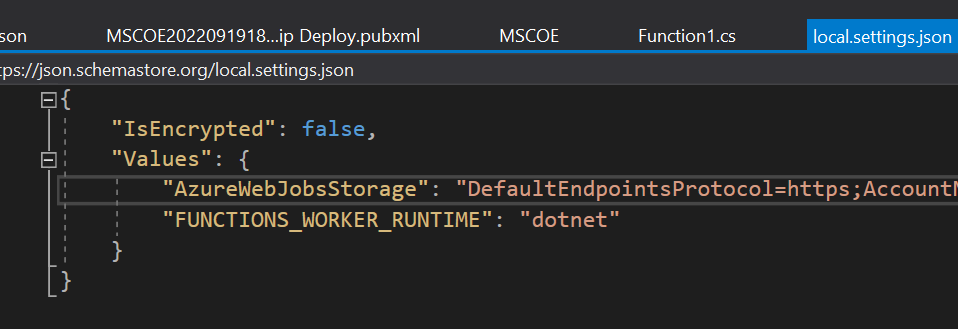
*6 Host.json settings reference for Azure Functions*

The host.json metadata file contains configuration options that affect all functions in a function app instance. Configurations in host.json related to bindings/triggers are applied equally to each function in the function app.



*7 Local settings reference for Azure Functions*

Local settings in a function app contain configuration options that affect all functions for that function app. When you run locally, these configurations will be useful.

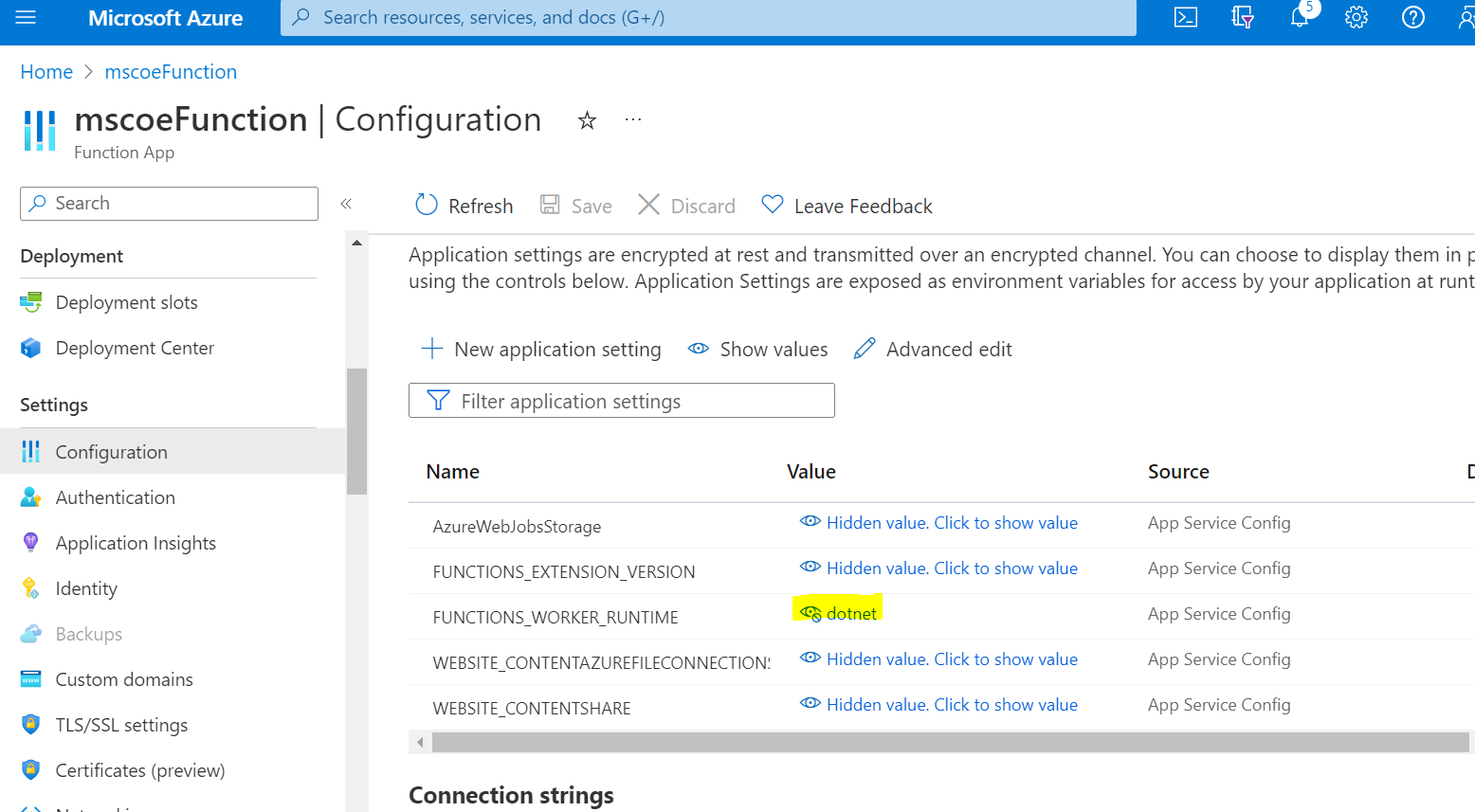


*8 App settings reference for Azure Functions*

App settings in a function app contain configuration options that affect all functions for that function app. When you run locally, these settings are accessed as local [environment variables](https://learn.microsoft.com/en-us/azure/azure-functions/functions-develop-local#local-settings-file)/local.settings.josn. This article lists the app settings that are available in function apps.

There are several ways that you can add, update, and delete function app settings:

* [In the Azure portal.](https://learn.microsoft.com/en-us/azure/azure-functions/functions-how-to-use-azure-function-app-settings#settings)
* [By using the Azure CLI.](https://learn.microsoft.com/en-us/cli/azure/functionapp/config/appsettings#az-functionapp-config-appsettings-set)
* [By using Azure PowerShell.](https://learn.microsoft.com/en-us/powershell/module/az.functions/update-azfunctionappsetting)



*9 Azure Functions triggers*

Triggers cause a function to run. A trigger defines how a function is invoked and a function must have exactly one trigger. Triggers have associated data, which is often provided as the payload of the function.

*10 Bindings (Input and Output)*

Binding to a function is a way of declaratively connecting another resource to the function; bindings may be connected as *input bindings*, *output bindings*, or both. Data from bindings is provided to the function as parameters.

You can mix and match different bindings to suit your needs. Bindings are optional and a function might have one or multiple input and/or output bindings.

There are many types of triggers. Few of the below mentioned triggers and its use cases

