**What is Azure Functions?**

Azure Functions is a serverless application platform. It allows developers to host business logic that can be executed without provisioning infrastructure. Functions provides intrinsic scalability and you are charged only for the resources used. You can write your function code in the language of your choice, including C#, F#, JavaScript, Python, and PowerShell Core. Support for package managers like NuGet and NPM is also included, so you can use popular libraries in your business logic.

Your business logic runs as functions and you don't have to manually provision or scale infrastructure. The cloud provider manages infrastructure. Your app is automatically scaled out or down depending on load.

**Choosing a service plan**

1. Consumption service plan
   1. automatic scaling and bills you when your functions are running
   2. configurable timeout period for the execution of a function. By default, it is 5 min but may be configured to have a timeout as long as 10 minutes.
2. Azure App Service plan
   1. avoid timeout periods by having your function run continuously on a VM that you define

**Event driven**

Functions are *event driven*. This means they run only in response to an event (called a "trigger"), such as receiving an HTTP request, or a message being added to a queue. You configure a trigger as part of the function definition. This approach simplifies your code by allowing you to declare where the data comes from (trigger/input binding) and where it goes (output binding). You don't need to write code to watch queues, blobs, hubs, etc. You can focus purely on the business logic.

**Triggers**

Functions are event driven, which means they run in response to an event. The type of event that starts the function is called a trigger. You must configure a function with exactly one trigger.

Azure supports triggers for the following services.

| **Service** | **Trigger description** |
| --- | --- |
| Blob storage | Start a function when a new or updated blob is detected. |
| Azure Cosmos DB | Start a function when inserts and updates are detected. |
| Event Grid | Start a function when an event is received from Event Grid. |
| HTTP | Start a function with an HTTP request. |
| Microsoft Graph Events | Start a function in response to an incoming webhook from the Microsoft Graph. Each instance of this trigger can react to one Microsoft Graph resource type. |
| Queue storage | Start a function when a new item is received on a queue. The queue message is provided as input to the function. |
| Service Bus | Start a function in response to messages from a Service Bus queue. |
| Timer | Start a function on a schedule. |

**Hello world! with Azure Function in Visual Studio 2019**

1. Open **Visual Studio 2019**
2. Go to **Create new project -> search “function”**
3. Select project **Azure Function** (Note: select language C#)
4. Provide **Project Name**
5. Poop up will open to choose template, it will display the number of triggers available.
6. Select **Http trigger** (Storage account: Storage emulator & Authorization level: Function)
7. Renamed the default **Function1.cs** class name to **DemoFunction.cs** (also updated the name in **FuunctionName** attribute)

**Run the default code**

1. To run the function app, click F5 or Debug -> run
2. A console window will open with the bolt sign in it
3. Once it is running, you will get the **URL in console** (ex. http://localhost:7071/api/DemoFunction)
4. Copy that and open in browser
5. It takes **name** query string (ex. <http://localhost:7071/api/DemoFunction?name=cp>)
6. You will see the result in browser
7. Alternatively, you can use Postman

**A demo app to return the employee detail**

Please find the below git repository link to download the code

<https://github.com/copaliwal/Azure-Function-demo>