**Microsoft azure**

**Cloud service:**

Cloud service is any service made available to the users on demand via internet from a cloud computing providers servers as opposed to being provided from a company’s own on-premises server. cloud computing means storing and accessing data and programs over the Internet instead of your computer's hard drive

**Types of cloud computing:**

1. **Public cloud:**

Public clouds are owned and operated by a third-party cloud service providers, which deliver their computing resources like servers and storage over the Internet. Microsoft Azure is an example of a public cloud. With a public cloud, all hardware, software, and other supporting infrastructure is owned and managed by the cloud provider. You access these services and manage your account using a web browser.

1. **Private cloud:**

A private cloud refers to cloud computing resources used exclusively by a single business or organization. A private cloud can be physically located on the company’s on-site datacenter. Some companies also pay third-party service providers to host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network.

1. **Hybrid cloud:**

Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives your business greater flexibility, more deployment options, and helps optimize your existing infrastructure, security, and compliance.

**Types of cloud services:**

1. **Infrastructure as a service**

Infrastructure as a service (IaaS) is an instant computing infrastructure, provisioned and managed over the Internet. IaaS helps you avoid the expense and complexity of buying and managing your own physical servers and other datacenter infrastructure. Each resource is offered as a separate service component, and you only need to rent a particular one for as long as you need it

1. **Platform as a service**

Platform as a service (PaaS) is a complete development and deployment environment in the cloud, with resources that enable you to deliver everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applicat

ions.

1. **Software as a service**

Software as a service (SaaS) allows users to connect to and use cloud-based apps over the Internet. Common examples are email, calendaring, and office tools (such as Microsoft Office 365).

**Microsoft azure:**

Microsoft Azure is an ever-expanding set of cloud services to help your organization meet your business challenges. It’s the freedom to build, manage, and deploy applications on a massive, global network using your favourite tools and frameworks.

**Microsoft function apps:**

Azure Functions is a solution for easily running small pieces of code, or "functions," in the cloud. You can write just the code you need for the problem at hand, without worrying about a whole application or the infrastructure to run it. Functions can make development even more productive, and you can use your development language of choice, such as C#, F#, Node.js, Java, or PHP.

* **Features:**

**Choice of language: function apps could be written using any languages like C#, F#, javascript etc.**

**Pay-per-use pricing model:** Pay only for the time spent running your code.

**Bring your own dependencies:** Functions supports NuGet and NPM, so you can use your favourite libraries.

**Integrated security** - Protect HTTP-triggered functions with OAuth providers such as Azure Active Directory, Facebook, Google, Twitter, and Microsoft Account.

**Flexible development** - Code your functions right in the portal or set up continuous integration and deploy your code through GitHub, Azure DevOps Services, and other supported development tools.

**Open-source** - The Functions runtime is open-source and available on GitHub.

* **Why functions?**

Functions is a great solution for processing data, integrating systems, working with the internet-of-things (IoT), and building simple APIs and microservices (Microservice architecture, or simply microservices, is a distinctive method of developing software systems that tries to focus on building single-function modules with well-defined interfaces and operations.)

We can consider Functions for tasks like image or order processing, file maintenance, or for any tasks that you want to run on a schedule.

**HTTPTrigger** - Trigger the execution of your code by using an HTTP request.

**TimerTrigger** - Execute clean-up or other batch tasks on a predefined schedule.

**CosmosDBTrigger** - Process Azure Cosmos DB documents when they are added or updated in collections in a NoSQL database.

**BlobTrigger** - Process Azure Storage blobs when they are added to containers. You might use this function for image resizing.

**QueueTrigger** - Respond to messages as they arrive in an Azure Storage queue.

**ServiceBusQueueTrigger** - Connect your code to other Azure services or on-premises services by listening to message queues.

**ServiceBusTopicTrigger** - Connect your code to other Azure services or on-premises services by subscribing to topics.

Azure Functions supports triggers, which are ways to start execution of your code, and bindings, which are ways to simplify coding for input and output data.

* **Integrations:**

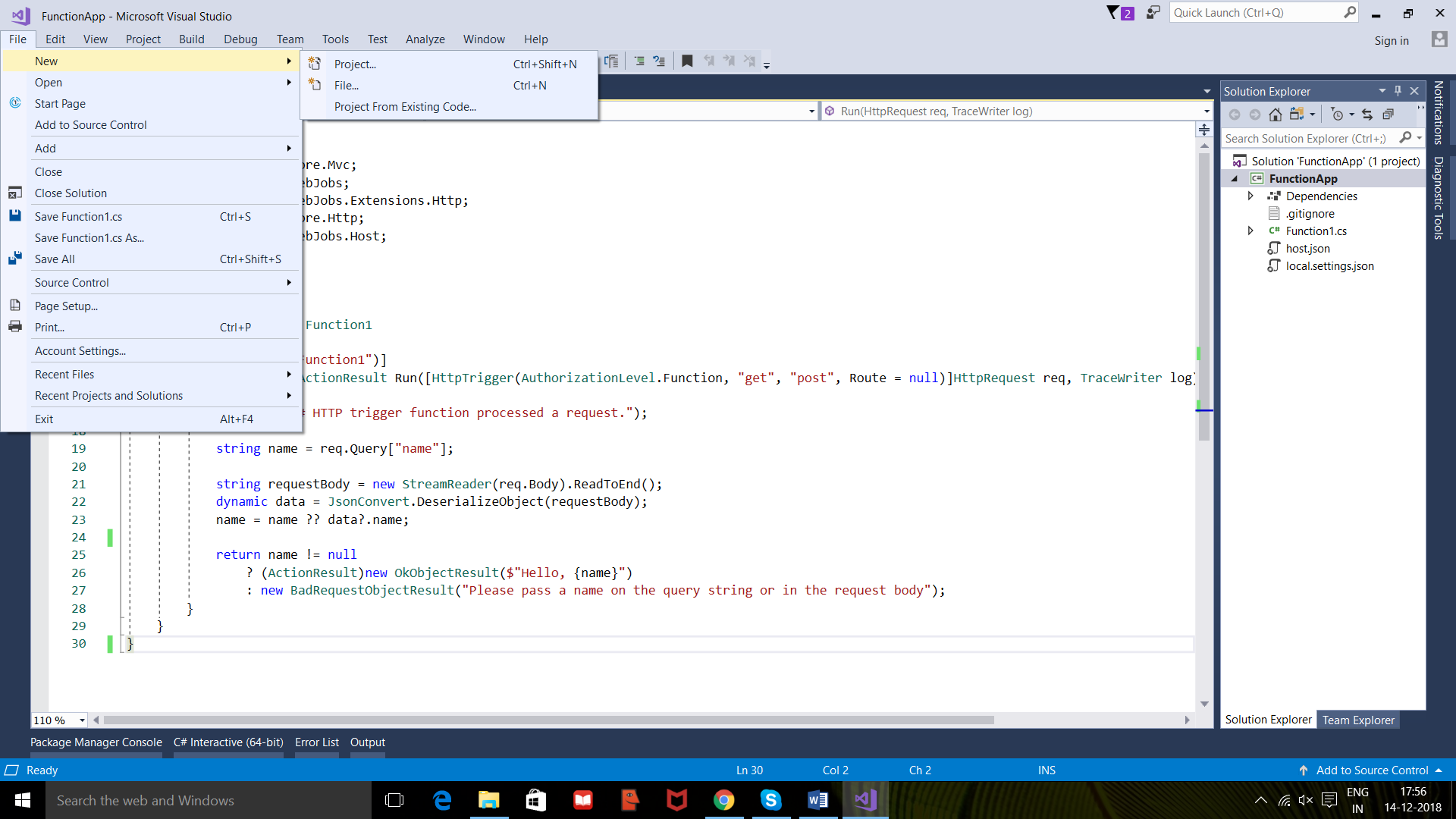
Azure Functions integrates with various Azure and 3rd-party services. These services can trigger your function and start execution, or they can serve as input and output for your code. The following service integrations are supported by Azure Functions:

* Azure Cosmos DB
* Azure Event Hubs
* Azure Event Grid
* Azure Notification Hubs
* Azure Service Bus (queues and topics)
* Azure Storage (blob, queues, and tables)
* On-premises (using Service Bus)
* Twilio (SMS messages)

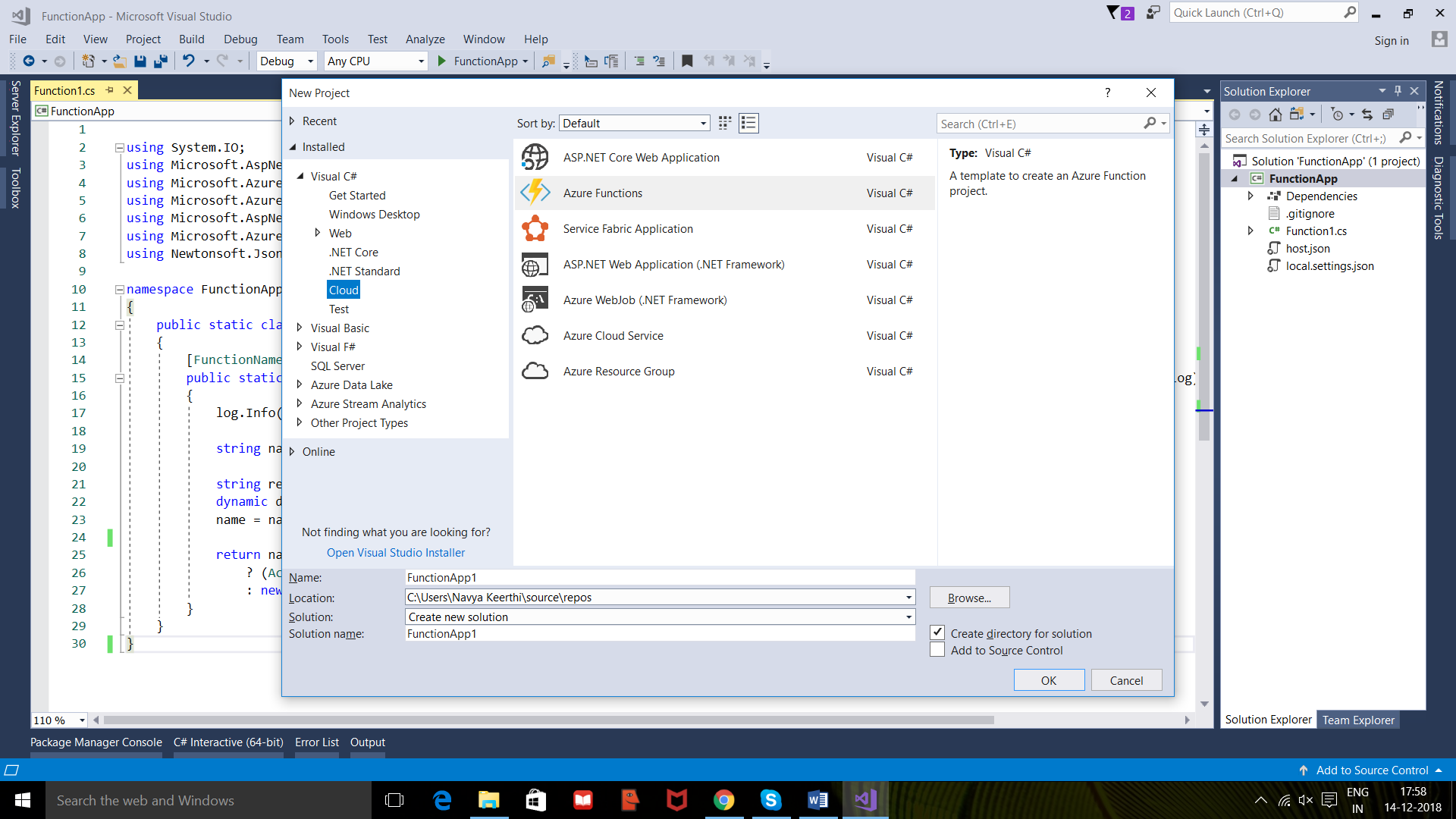
**How to create Functions app in Visual studio?**

**Step 1:** Open visual studio.

**Step 2:** from file menu click on new -> project

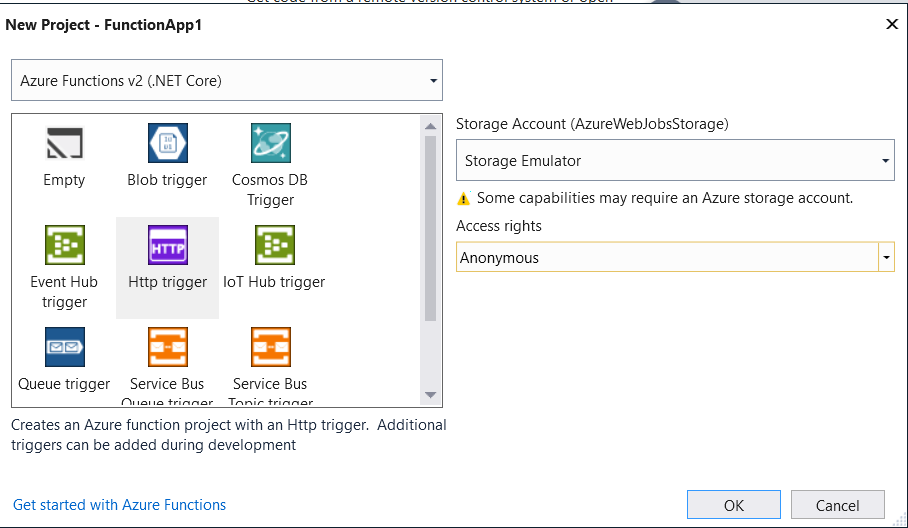


**Step 3:** extend installed, Visual C#, web, cloud and select Azure Functions.



Give the name of the project and click ok.

**Step 4:** you will get the following window.



Here

**Azure Functions 2.x (.NET Core)** indicates function project that uses the version 2.x runtime of Azure Functions which supports .NET Core.

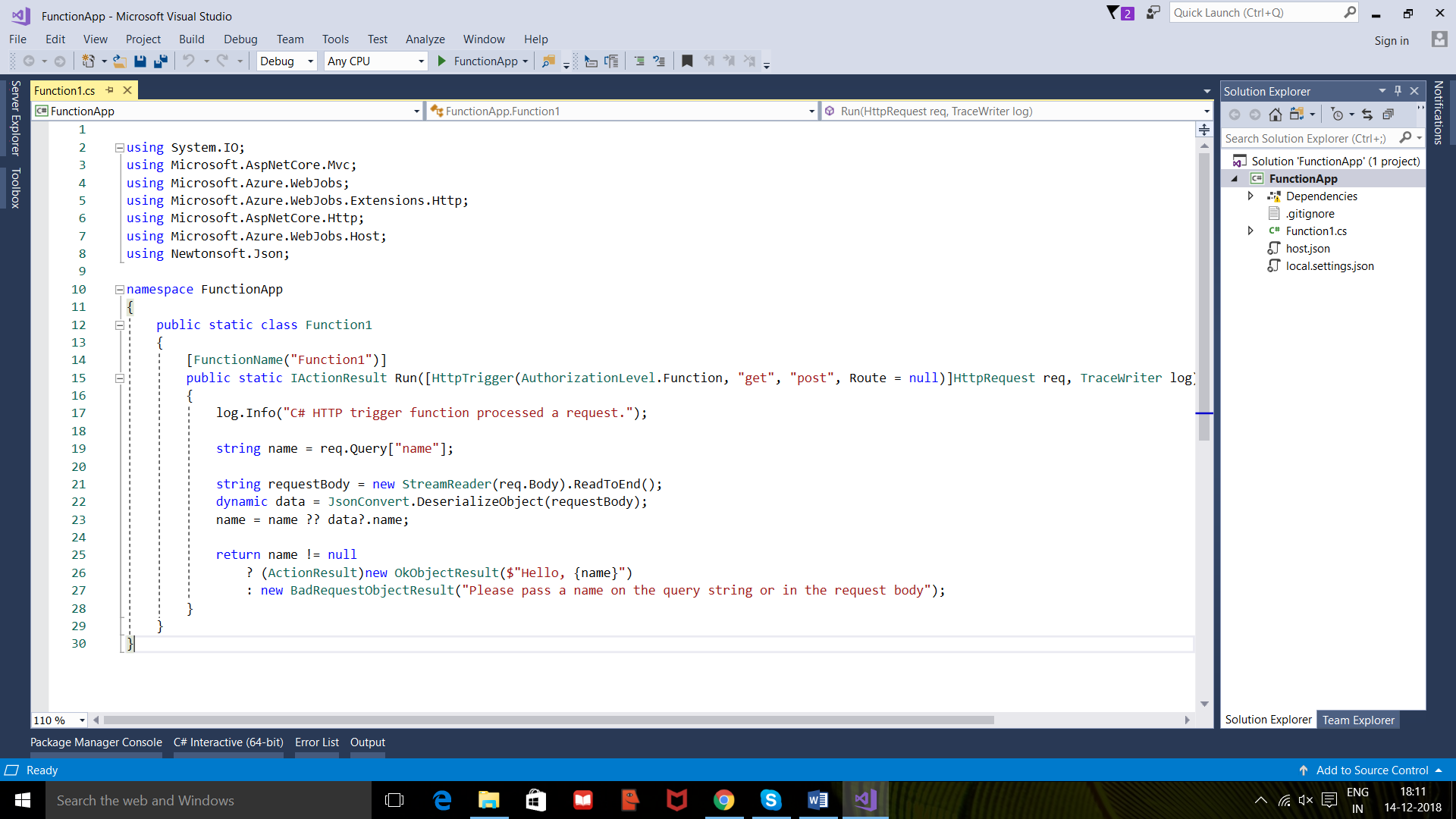
**HTTP trigger** indicates that this creates a function triggered by an HTTP request.

**Storage Emulator** indicates that HTTP trigger doesn't use the Storage account connection. All other trigger types require a valid Storage account connection string.

**Anonymous** indicates that the created function can be triggered by any client without providing a key. This authorization setting makes it easy to test your new function.

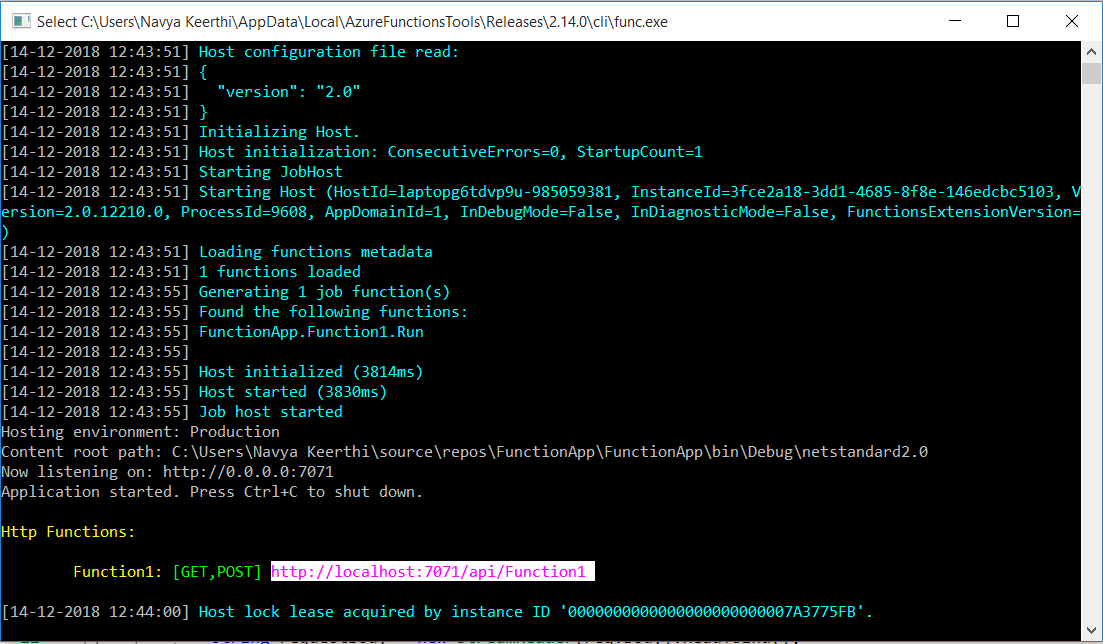
**Step 5:** now click ok to create a function app.

You will be displayed as follows.



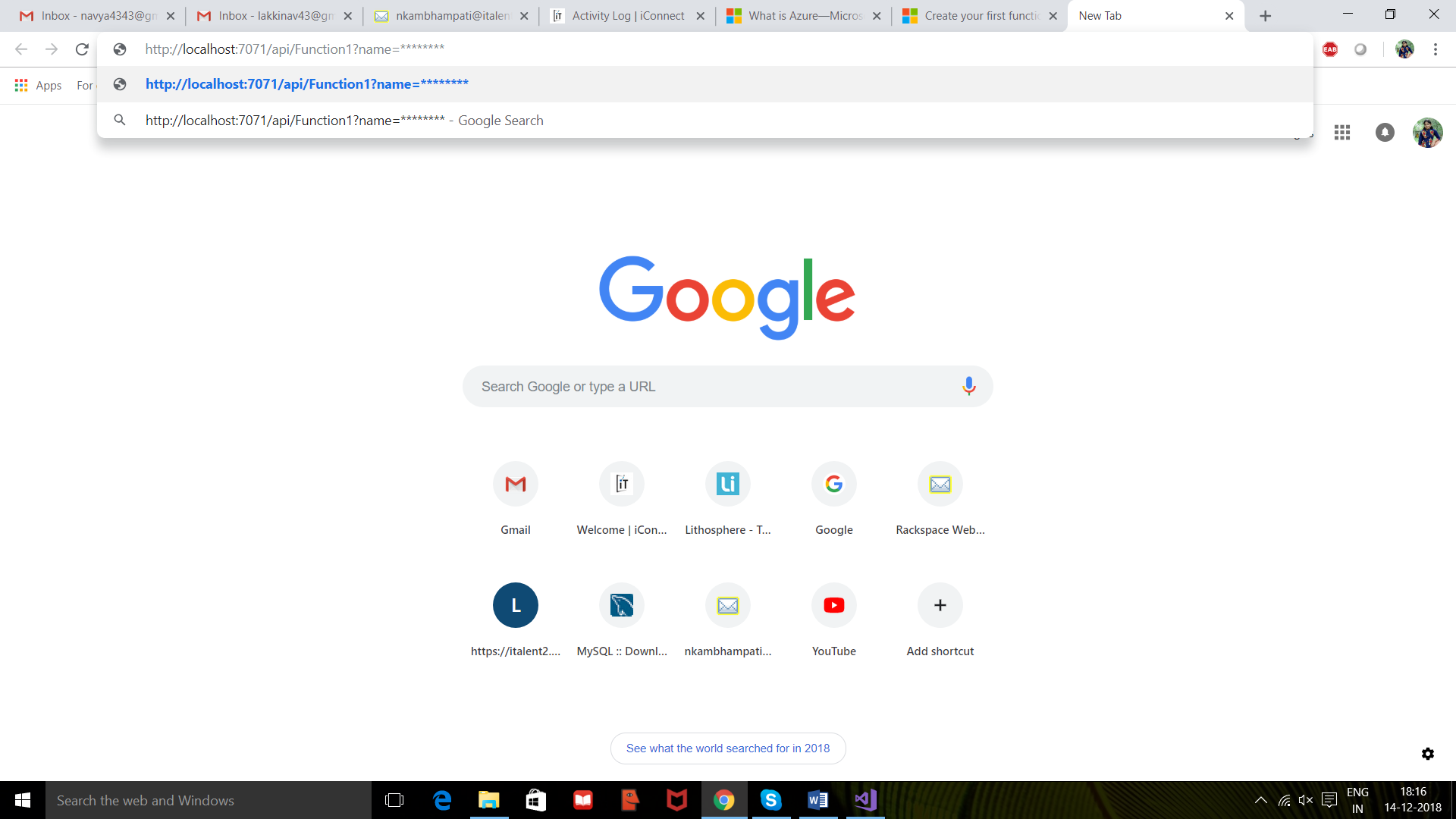
**Step 6:** now run the code by clicking F5 key.

**Step 7:** You will be getting the response as below



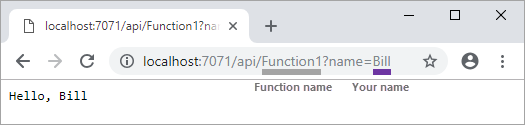
**Step 8:**

Copy the url and hit it in local host with name as query parameter as beow



**Step 9:**

You will get getting response as a message saying hello \*\*\*\* as shown below



This is how we execute a function app in visual studio.

**Reference links:**

1. <https://azure.microsoft.com/en-us/overview/what-is-azure/>
2. <https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-your-first-function-visual-studio>