



Azure Functions

Lesson Objectives



At the end of this module you will be able to:

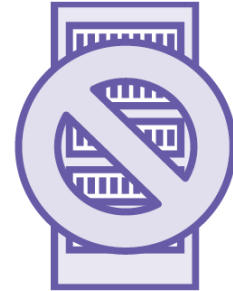
- ✓ Understand Serverless computing
- ✓ Create Function App and host Azure functions in it
- ✓ Create Azure functions which responds to variety of events
- ✓ Deploy Azure Functions



Serverless Compute Service



- Serverless computing is the abstraction of servers, infrastructure, and operating systems.
- Serverless computing is event-driven, and resources are allocated as soon as they're triggered by an event, we need to pay for what we use
- Serverless apps provisions to manage any servers, so that we can take our mind off at the infrastructure concerns



© 2018 Capgemini. All rights reserved.

3

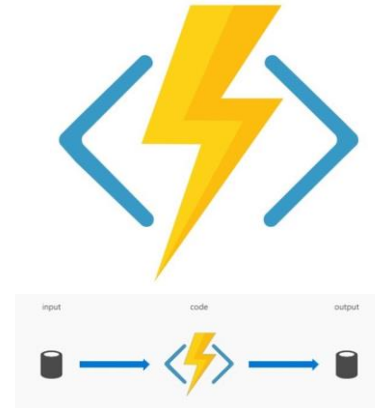
Serverless computing allow us to spend all our time in building and deploying great apps, and none of our time in managing servers because the infrastructure we need to run and scale your apps is managed for you.

We can Focus our efforts on our business. Redirect resources from infrastructure management into innovating and bringing apps to market faster

Serverless compute scales from nothing to handle tens of thousands of concurrent functions almost instantly (within seconds), to match any workload, and without requiring scale configuration. It reacts to events and triggers in near-real time

Azure Functions Introduction

- Azure Functions is a serverless compute service that enables you to run code on-demand without having to explicitly provision or manage infrastructure
- Azure Function app is required to host the execution of Azure functions
- Function app lets you to group functions as a logic unit for easier management, deployment, and sharing of resources.



© 2018 Capgemini. All rights reserved.

4

Microsoft Azure Function represents a solution for running small pieces of the code (known as functions) in the cloud serverless platform - function as a service (FaaS). Basically, the cloud provider gives you a "function boilerplate" with inputs for processing your business logic with delivery outputs. All Inputs and Outputs are binding based on the requirements in a declarative way. The Functions in the Serverless architecture model are processing in the Push manner based on the event-driven triggers and/or REST API Gateway for pushing (forwarding) messages to the specific function. The Triggers and/or API Gateway are parts of the Serverless Architecture and they are fully transparent to the Functions, in other words, the user is focusing on the body of the Functions and its Inputs/Outputs bindings.

The Serverless Architecture is based on the Push Model, where the event is Pushed to the Function. This model allows to scale very well and performing high throughputs for event streams. Note, that the consumer of the serverless FaaS platform doesn't need to handle scalability, restarting, recycling, triggering, hosting, etc. application. It is fully transparent and it is handled by cloud provider, such as Microsoft Azure.

It's a great feature and next evolution of the cloud computing.

Azure Functions Use Cases



Experiments and prototyping

Automating development processes

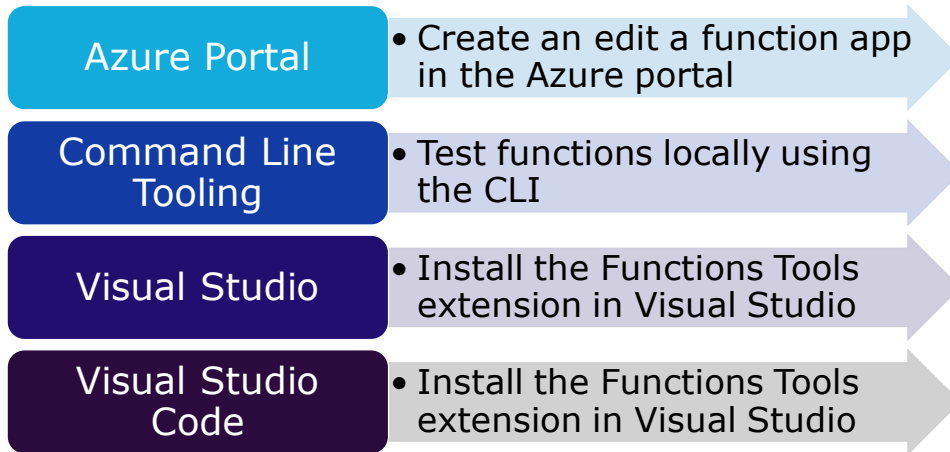
Decomposing or extending monolithic applications

Independent scaling

Adapters for integrating systems

Go serverless

Development environment



Triggers and bindings



- Azure Functions run a script or piece of code in response to a variety of events



HTTP

Function can be triggered based on a HTTP request



Blob

Function can be triggered when files are uploaded to or updated in Azure Blob storage.



GitHub webhook

Function can be triggered by an HTTP webhook request with a GitHub-specific payload.



Azure Cosmos DB

Function can be triggered when data is added to or changed in Azure Cosmos DB.



Queue

Function can be triggered when messages are submitted to an Azure Storage queue.



Timer

Function that runs on a schedule

Azure Functions Deployment Options



Manual Deployment

- Kudu, FTP, Web Deploy
- Invoke tool manually when you're ready to deploy
- Deploy from Visual Studio

Continuous Deployment

- Push to Git repository
- Code is automatically deployed
- Mercurial, DropBox also supported

Summary



- Serverless programming focus on the business needs
- Azure Functions reside in a Function App
- Azure Function = Event + Code
- As of now Azure functions can be create using the languages like C#, JavaScript and F#. Support to other languages like Java, Python, PHP are in experimental mode
- Azure functions can be learned easily from the sample templates





People matter, results count.

This message contains information that may be privileged or confidential and is the property of the Capgemini Group.
Copyright © 2017 Capgemini. All rights reserved.
Rightshore® is a trademark belonging to Capgemini.

About Capgemini

With more than 190,000 people, Capgemini is present in over 40 countries and celebrates its 50th Anniversary year in 2017. A global leader in consulting, technology and outsourcing services, the Group reported 2016 global revenues of EUR 12.5 billion. Together with its clients, Capgemini creates and delivers business, technology and digital solutions that fit their needs, enabling them to achieve innovation and competitiveness. A deeply multicultural organization, Capgemini has developed its own way of working, the *Collaborative Business Experience*™, and draws on *Rightshore*®, its worldwide delivery model.

Learn more about us at
www.capgemini.com

This message is intended only for the person to whom it is addressed. If you are not the intended recipient, you are not authorized to read, print, retain, copy, disseminate, distribute, or use this message or any part thereof. If you receive this message in error, please notify the sender immediately and delete all copies of this message.