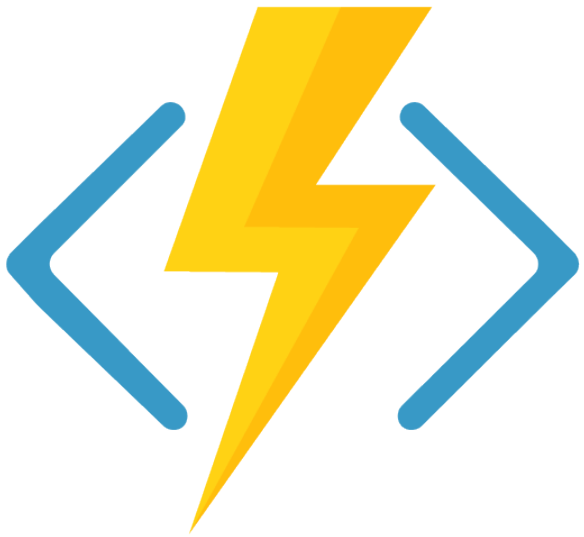
Microsoft Azure Developer: Create Serverless Functions

INTRODUCING AZURE FUNCTIONS

Mark Heath
MICROSOFT AZURE MVP

@mark_heath www.markheath.net

Azure Functions



Speed up your development

Reduce your costs

Scale automatically

Functions as a Service (FaaS)

A platform for running “functions”, which are simply **your code** running in response to an **event**

Supported Languages

C#

JavaScript

F#

Java

Python

PowerShell

Triggers

Timer: run a function on a schedule

Message: listen for messages on a queue

HTTP Request: implement web APIs or webhooks

Many more: Azure Storage **blob** creation, **Cosmos DB** new row

Bindings

Queue	Cosmos DB	Blob Storage
Service Bus	Table Storage	SendGrid

Eliminate repetitive boilerplate code

Focus on core business logic

Serverless Architecture

Servers are managed for you

- Servers are abstracted away

Per-second billing model

- Only pay when your code runs
- Monthly free grant

Automatic scale

- Multiple servers meet demand

Simpler Cheaper More Scalable

Hosting Models

Consumption Plan (Serverless)

- Per-second billing
- 1,000,000 executions
- 400,000 GB-s

App Service Plan

- Reserved servers
- Predictable monthly cost

Premium Plan

- Pre-warmed instances
- VNet integration
- Longer run duration

Docker Container

- Run anywhere
- On premises
- Other cloud providers

Development Environments

Azure Portal	Visual Studio	Azure Functions Core Tools
https://portal.azure.com	Powerful IDE	Cross platform
Experiments	Azure Functions extensions	Visual Studio Code
Proof of concept	Debug and test locally	Azure Functions extensions

Additional Features

Security

API keys

Identity provider integration

Durable Functions

Define workflows

Run tasks in parallel

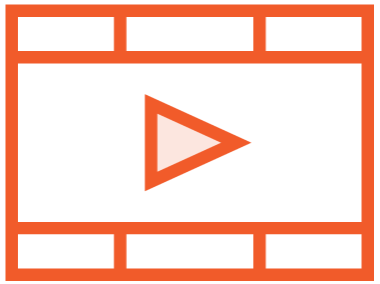
Retries and error handling

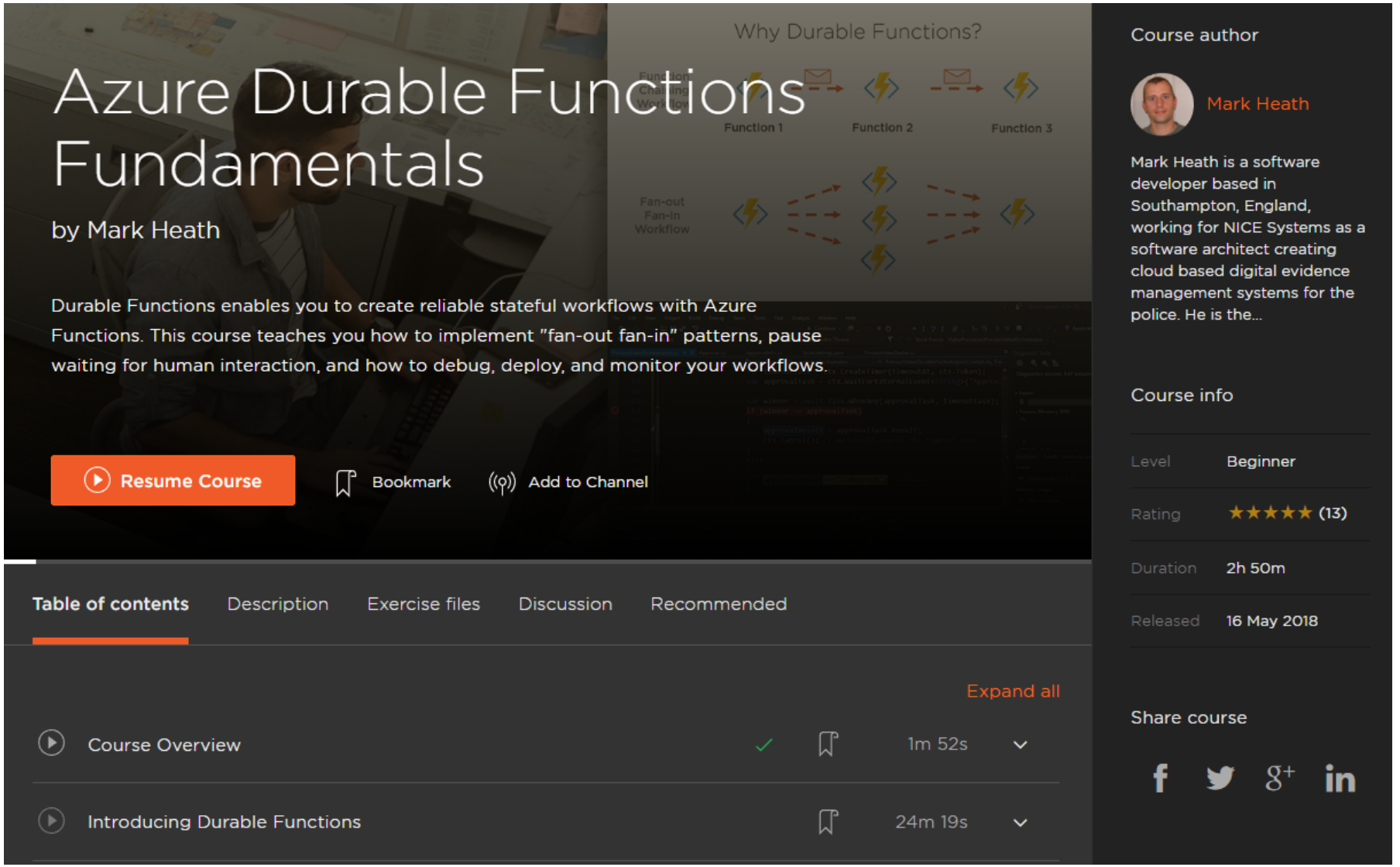
Proxies

Route incoming requests

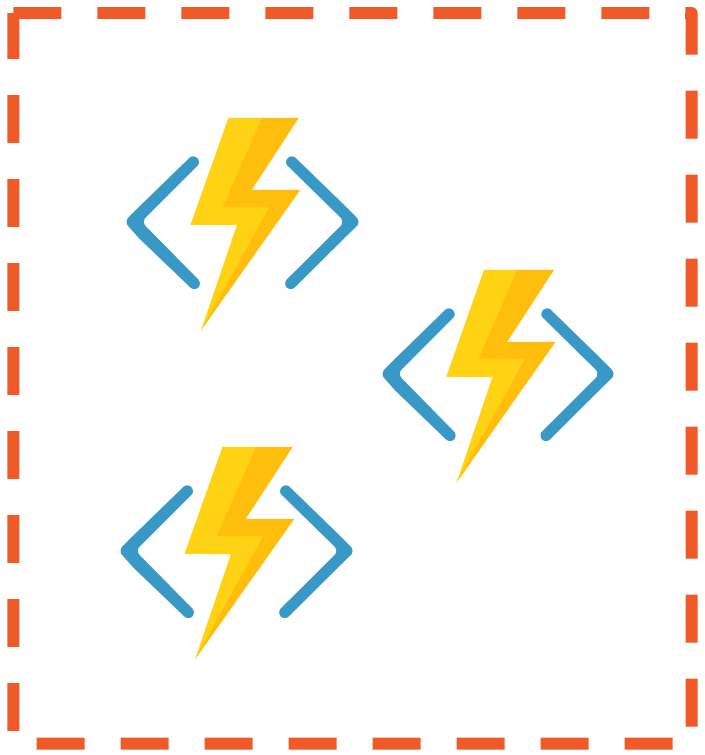
Static website

Transform requests and responses




Why Durable Functions?
Function 1
Function 2
Function 3
Fan-out
Fan-in
Workflow
Azure Durable Functions Fundamentals
by Mark Heath
Durable Functions enables you to create reliable stateful workflows with Azure Functions. This course teaches you how to implement "fan-out fan-in" patterns, pause waiting for human interaction, and how to debug, deploy, and monitor your workflows.
Resume Course
Bookmark
Add to Channel
Table of contents
Description
Exercise files
Discussion
Recommended
Expand all
Course Overview
1m 52s
Introducing Durable Functions
24m 19s
Course author
Mark Heath
Mark Heath is a software developer based in Southampton, England, working for NICE Systems as a software architect creating cloud based digital evidence management systems for the police. He is the...
Course info
Level
Beginner
Rating
(13)
Duration
2h 50m
Released
16 May 2018
Share course

Function Apps



Unit of deployment

Share common configuration

Scale together

Logically related

Summary

Functions as a Service (FaaS)

Serverless environment

- Consumption pricing model
- Automatic scaling

Triggers and bindings

Additional features

- Security, proxies & workflows

Development environments

- Portal, Visual Studio, VS Code

Up next:
Creating Azure Functions in the Azure Portal