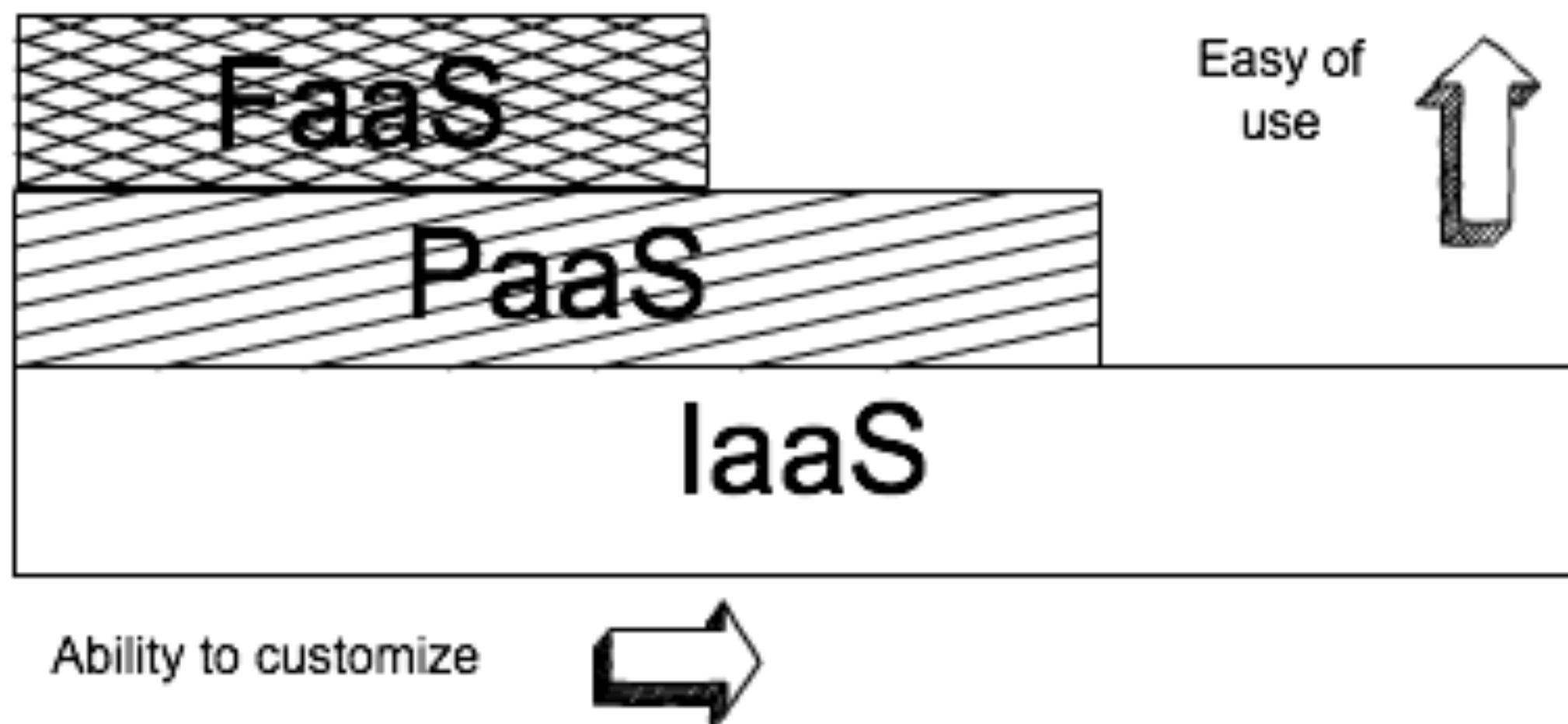


Serverless Computing with Azure Functions

Benefits of Serverless

- » Easier to operate and manage apps
- » Faster innovation and scaling
- » Great community

Cloud Computing



What is serverless?

Serverless is event driven logic (functions) in the cloud.



```
12     var rowCount = tableReaderBinding  
13         .Where(m => m.PartitionKey == "Test")  
14             .ToList().Count;  
15     var message = new Message()
```

How to Do Serverless



Disclaimer

Serverless still has servers but they are abstracted (hidden) from developers/end users like us.

Metaphor

Serverless is a drive-through model: "You pay for the hamburger, not the cow". (Tofu not soy field for vegetarians.)

Serverless Use Cases

- >> IoT
- >> Rarely/infrequently used processes (cut costs)
- >> Analytics, big data
- >> Microservices
- >> Batch jobs, e.g., image or video processing

Example: Surveys

- >> Web page form hosted on a static web server
- >> DocumentDB or Azure Table storage to store data
- >> Azure Function to respond to POST /survey (stateless) by saving the data to DB

Example: Archiving

- >> Cloud storage for file uploads (CSV)
- >> Event listener to trigger Azure Function
- >> Azure Function to compress (gzip) CSV into zip archive

Example: Bot

- >> Bot which sends a message
- >> Bot Framework to trigger Azure Function
- >> Azure Function which calls Cortana Analytics service to answer a message

Example: Firewall

- >> CloudFlare API to block IPs
- >> Aruze Function to store IPs to block in Azure table
- >> Azure Function on time to remove blocked IPs after 1h

Azure Functions is Serverless environment from Microsoft Azure cloud computing family of services

Aruze Functions UX is Open Source

You can run it on your own infrastructure! HUGE benefits!

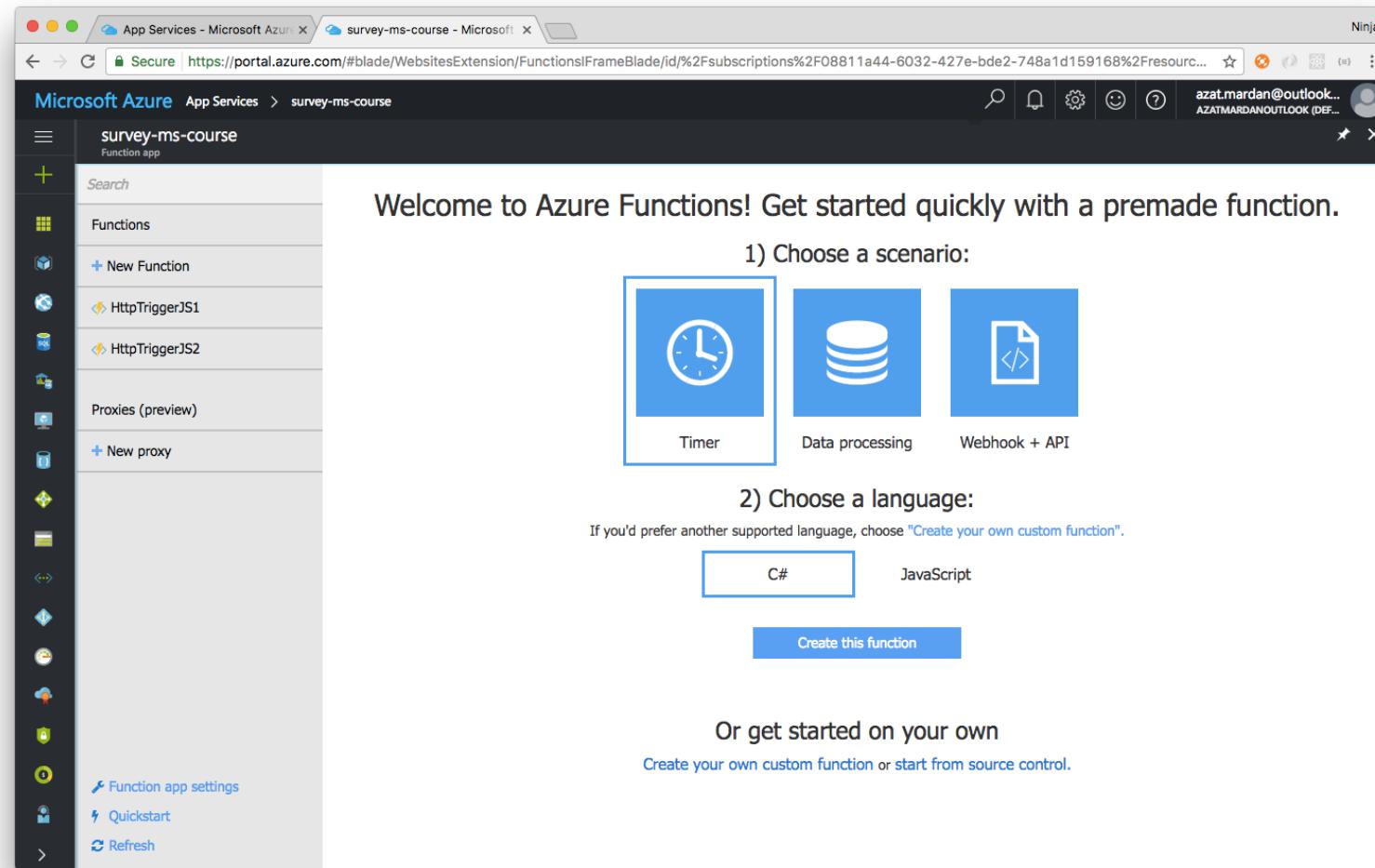
- » No vendor lock in.
- » Transparency of up-coming features and existing bugs.

<https://github.com/projectkudu/AzureFunctionsPortal>

Main Parts of Azure Functions

Azure Functions Portal

<https://functions.azure.com>



Types

- >> Web API
- >> WebHook
- >> Code triggered by event
- >> Code triggered by time

Azure Functions Languages

» JavaScript

» C#

» F#

» Python

» PHP

» Bash

» Batch

» PowerShell

Azure Functions Templates

Templates are pre-defined functions that demonstrate a working scenario and could be used as a starting point for more complex ones

Azure Functions Template Format

- >> Binding.json
- >> Function.json
- >> Code file
- >> Metadata.json
- >> Sample.dat

Binding.json

Function.json

Code file

Metadata.json

Sample.dat

Azure Functions Endpoint

WebHook + API functions have an HTTP(S) endpoint

Let's Create Our First Function!

Azure Functions—Serverless A X

Ninja

Microsoft Corporation [US] | https://azure.microsoft.com/en-us/services/functions/

Microsoft Azure

SALES 1-800-867-1389 ▾ MY ACCOUNT PORTAL Search

Why Azure Solutions Products Documentation Pricing Partners Blog Resources Support FREE ACCOUNT >

Azure Functions

Process events with a serverless code architecture

An event-based serverless compute experience to accelerate your development. It can scale based on demand and you pay only for the resources you consume.

Try It For Free >

Login to your account >

Create your free account >

Take the Azure Functions Code Challenge!

- Get started with Serverless
- Ramp up on event based paradigm
- Learn about triggers and bindings

Click to get started!



Explore Azure Functions: Pricing details Documentation Community

 Serverless

Don't worry about the infrastructure and provisioning of servers, especially when your Functions call rate scales up.

 Accelerate development

Write your code in the Functions editor and click run for immediate execution.

 Bind into services

Easily click and add bindings to Azure services and external services (Box, DropBox, OneDrive, SendGrid, ...) to get input into or output from Functions.

> JS php

Develop your way

Create Functions in a variety of languages, including JavaScript, C#, F#, as well as scripting options such as Python, PHP, Bash, Batch, and PowerShell. And do it all in an easy-to-use web-based interface. Or upload and trigger pre-

Everything - Microsoft Azure

Secure https://portal.azure.com/#blade/Microsoft_Azure_Marketplace/GalleryFeaturedMenuItemBlade/selectedMenuItemId/home/searchQuery/Function%20App

Microsoft Azure New > Marketplace > Everything

Search resources

azat.mardan@outlook... AZATMARDANOUTLOOK (DEF...)

Marketplace

Everything

Filter

Function App

Results

NAME	PUBLISHER	CATEGORY
Function App	Microsoft	
eperi Gateway for Cloud Apps VM	eperi	Compute
Citrix XenApp 7.12 Trial	Citrix	Compute
Official MEAN stack by Linnovate (MEAN.IO 0.5.0)	Mean.io	Compute
Redmine	Bitnami	Compute
Backendless Pro	Bitnami	Compute
Crossbar.io	Tavendo	Compute
myFidoc	Cordis Solutions	Compute
RemoteApp	Microsoft	
Tracks	Bitnami	Compute
OpenGeo Suite from Boundless	Boundless Spatial	Compute
Mahara	Bitnami	Compute
rethinkdb	Docker	Compute
MediaRich® All Media Server	Equilibrium	Compute
ManageEngine Mobile Device Manager Plus	ManageEngine	Compute

Related to your search ▾

CakePHP Cake Software Foundation

DocAve for Office 365 Sustainable Adoption AvePoint Inc.

SlashDB Cloud Edition vt.enterprise

This screenshot shows the Microsoft Azure Marketplace interface. The left sidebar lists various service categories like Compute, Networking, Storage, etc. The main search bar at the top has 'Function App' typed into it. Below the search bar is a table of search results, which includes items such as 'Function App' (published by Microsoft), 'eperi Gateway for Cloud Apps VM', 'Citrix XenApp 7.12 Trial', and 'Official MEAN stack by Linnovate (MEAN.IO 0.5.0)'. The results are sorted by Name, Publisher, and Category. At the bottom of the page, there are sections for 'Related to your search' featuring links to 'CakePHP' and 'DocAve for Office 365 Sustainable Adoption'.

Function app - Microsoft Azure X

Secure | https://portal.azure.com/#blade/WebsitesExtension/FunctionsFrameBlade/id/%2Fsubscriptions%2FO8811a44-6032-427e-bde2-748... ☆

Microsoft Azure All resources > Function app

Function app survey-ms-course

Search

Functions

+ New Function

The faster way to functions

Write any function in minutes - whether to run a simple job that cleans up a database or to build a more complex architecture. Creating functions is easier than ever before, whatever your chosen OS, platform, or development method. No install required.

Get started quickly with a premade function

1) Choose a scenario:

Timer Data processing Webhook + API

2) Choose a language:

If you'd prefer another supported language, choose "Create your own custom function".

C# JavaScript

Create this function

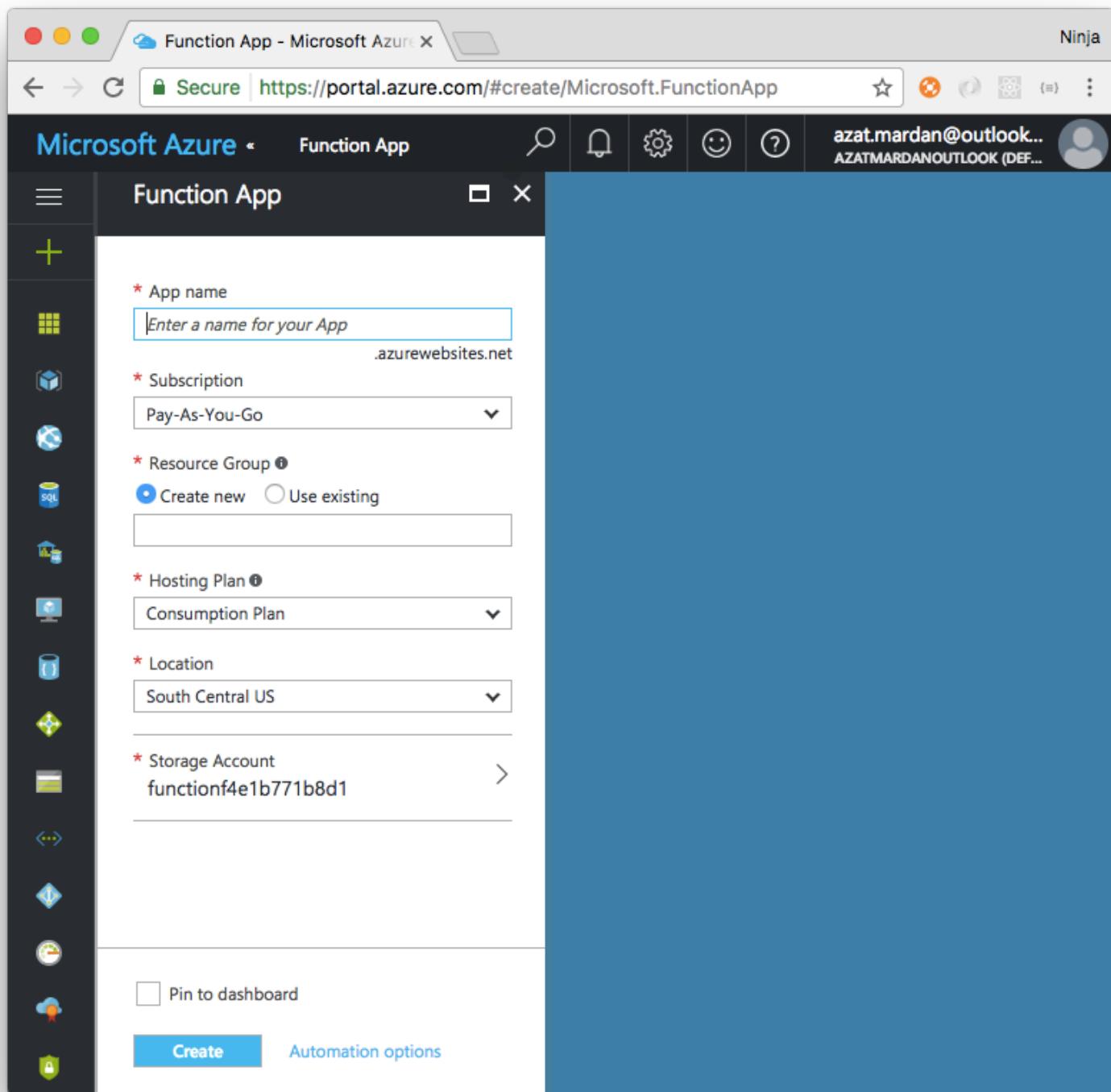
Or get started on your own

Create your own custom function or start from source control.

Function app settings

Quickstart

Refresh



Example of the Function App parameters

Name	Value
App name	ms-course-test-123
Subscription	Pay-As-You-Go
Resource Group	default (Use existing)
Hosting Plan	Consumption Plan
Location	South Central US
Storage Account	functionf4e1b771b8d1

Congratulations - Your First
Function App is Created!

Functions App Can have multiple function.

Examples:

HttpTriggerJS1

HttpTriggerJS2

HttpTriggerJS3

TimerTriggerJS1

survey-ms-course - Microsoft

Secure | https://portal.azure.com/#blade/WebsitesExtension/FunctionsFrameBlade/id/%2Fsubscriptions%2F08811a44-... ☆

Ninja

Microsoft Azure App Services > survey-ms-course

survey-ms-course Function app

index.js Save ▶ Run

```
1 module.exports = function (context, myTimer) {  
2     var timeStamp = new Date().toISOString();  
3  
4     if(myTimer.isPastDue)  
5     {  
6         context.log('JavaScript is running late!');  
7     }  
8     context.log('JavaScript timer trigger function ran!', timeStamp);  
9  
10    context.done();  
11};
```

View files Test Keys

Functions

+ New Function

HttpTriggerJS1

HttpTriggerJS2

HttpTriggerJS3

TimerTriggerJS1

</> Develop

Integrate

Manage

Monitor

Proxies (preview)

+ New proxy

Function app settings

Quickstart

Refresh

Logs

If you selected WebHoot+API, then your first function in your newly created app will be `HttpTriggerJS1`.

Next, four main screens to work
with a function.

Function app - Microsoft Azure X Ninja

Secure | https://portal.azure.com/#blade/WebsitesExtension/FunctionsFrameBlade/id/%2Fsubscriptions%2FO881a44-6032-427e-bde2-748... ☆

Microsoft Azure All resources > Function app

Function app survey-ms-course

Search

Functions

+ New Function

HttpTriggerJS1

</> Develop Integrate Manage Monitor

index.js Save Run Logs View Files Test Keys

Function Url: https://survey-ms-course.azurewebsites.net/api/HttpTriggerJS1?code=GpaRR1x5KuYd8wcivYQwFLpkZCjoJPHM5dqWgYdeQYXrDiGP1

```
1 module.exports = function (context, req) {  
2     context.log('JavaScript HTTP trigger function processed a request.');//  
3  
4     if (req.query.name || (req.body && req.body.name)) {  
5         res = {  
6             // status:  
7             body: "Hello"  
8         };  
9     }  
10    else {  
11        res = {  
12            status: 400,  
13            body: "Please provide a name."  
14        };  
15    }  
16    context.done(null, res);  
17};
```

Develop 1 of 4

The fastest way to edit code is with the code editor, but you can also use Git. This example uses NodeJS, but many other languages are also supported.

This page also includes a log stream and a test console for helping you debug your function.

Next Skip the tour and start coding

Function app settings Quickstart Refresh

Function app - Microsoft Azure

Secure | https://portal.azure.com/#blade/WebsitesExtension/FunctionsFrameBlade/id/%2Fsubscriptions%2FO8811a44-6032-427e-bde2-748...

Microsoft Azure All resources > Function app

Function app survey-ms-course

Search

Triggers ⓘ

HTTP (req)

Inputs ⓘ

+ New Input

Outputs ⓘ

+ New Output

HTTP trigger (req) delete

Allowed HTTP methods ⓘ

All methods

Request parameter name ⓘ

req

Authorization level ⓘ

Function

Integrate

2 of 4

Integrating your functions with other services and data sources is easy.

Set up automated actions based on external triggers, include other input data sources, and send the output to multiple targets.

Next

Skip the tour and start coding

Cancel

+ Documentation

Function app settings

Quickstart

Refresh

Function app - Microsoft Azure

Secure | https://portal.azure.com/#blade/WebsitesExtension/FunctionsFrameBlade/id/%2Fsubscriptions%2FO8811a44-6032-427e-bde2-748...

Microsoft Azure All resources > Function app

Function app survey-ms-course

Search

Functions

+ New Function

HttpTriggerJS1

Develop

- App Service Editor In portal editor with an integrated console and streaming logs [Go to App Service Editor](#)
- Application settings Manage environment variables and connection strings for your function app [Configure app settings](#)
- Dev Console In-portal console for accessing your function app's file system [Open dev console](#)

Deploy

- Continuous Integration Deploy your function code from GitHub, Bitbucket, or more [More](#)
- Kudu Access advanced functionality of your function app's file system, processes, and more

Manage

- App Service Settings Advanced Features. Access all the features of your function app's App Service
- CORS Allow your HTTP-triggered functions to be consumed by a browser [Configure authentication](#)
- Authentication/Authorization For functions that use the HTTP trigger, you can require calls to be authenticated [Configure API metadata](#)
- API definition Allow clients to more easily consume your HTTP-triggered functions

Daily Usage Quota (GB-Sec)

Enter value in GB-sec Set quota

Runtime version

Runtime version: latest (~1)

Function app settings >

Quickstart

Refresh

Function App Settings 3 of 4

Your functions are designed to run within Azure App Service as a function app, and this is where you modify the features of that app.

Changes made here will affect all of the functions within your function app.

Next

Skip the tour and start coding

Function app - Microsoft Azure X

Secure | https://portal.azure.com/#blade/WebsitesExtension/FunctionsFrameBlade/id/%2Fsubscriptions%2FO8811a44-6032-427e-bde2-748... Ninja

Microsoft Azure All resources > Function app

Function app survey-ms-course

Search

Functions

+ New Function

HttpTriggerJS1

</> Develop

Integrate

Manage

Monitor

index.js

Save Run Logs View Files Test Keys

Function Url: https://survey-ms-course.azurewebsites.net/api/HttpTriggerJS1?code=GpaRR1x5KuYd8wcivYQwFLpkZCjoJPHM5dqWgYdeQYXrDiGXp1

```
1 module.exports = function (context, req) {  
2     context.log('JavaScript HTTP trigger function processed a request.');  
3  
4     if (req.query.name || (req.body && req.body.name)) {  
5         res = {  
6             // status:  
7             body: "Hello"  
8         };  
9     }  
10    else {  
11        res = {  
12            status: 400,  
13            body: "Please provide a name."  
14        };  
15    }  
16    context.done(null, res);  
17};
```

Next Steps 4 of 4

Now that you've seen a bit of what Azure Functions can do, try it for yourself.

Tweak this sample function - Make it yours on the easy-to-use dashboard

Create a brand-new function - Get started with one of the prebuilt function templates

Dive into the documentation - Explore all of the Azure Functions features [here](#)

Close

Function app settings

Quickstart

Refresh

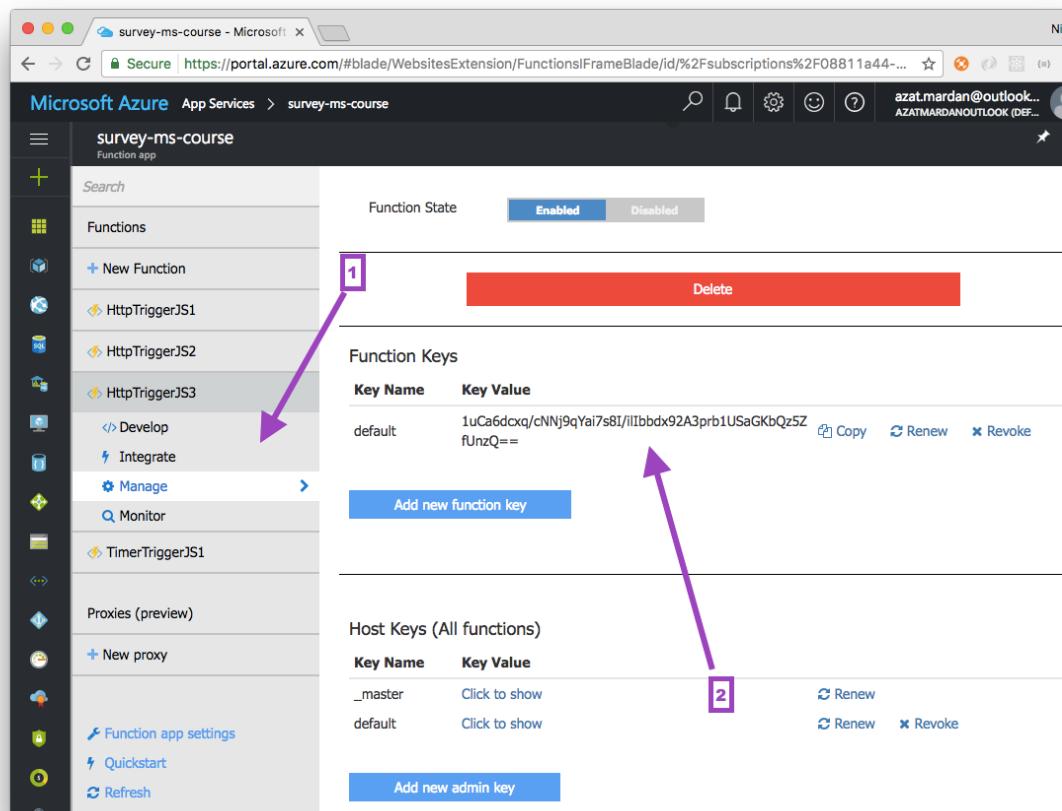
Seeing Your Web Functions Live

Go to URL: `https://{{func-app-name}}.azurewebsites.net/api/{{func-name}}?code={{function-key}}`

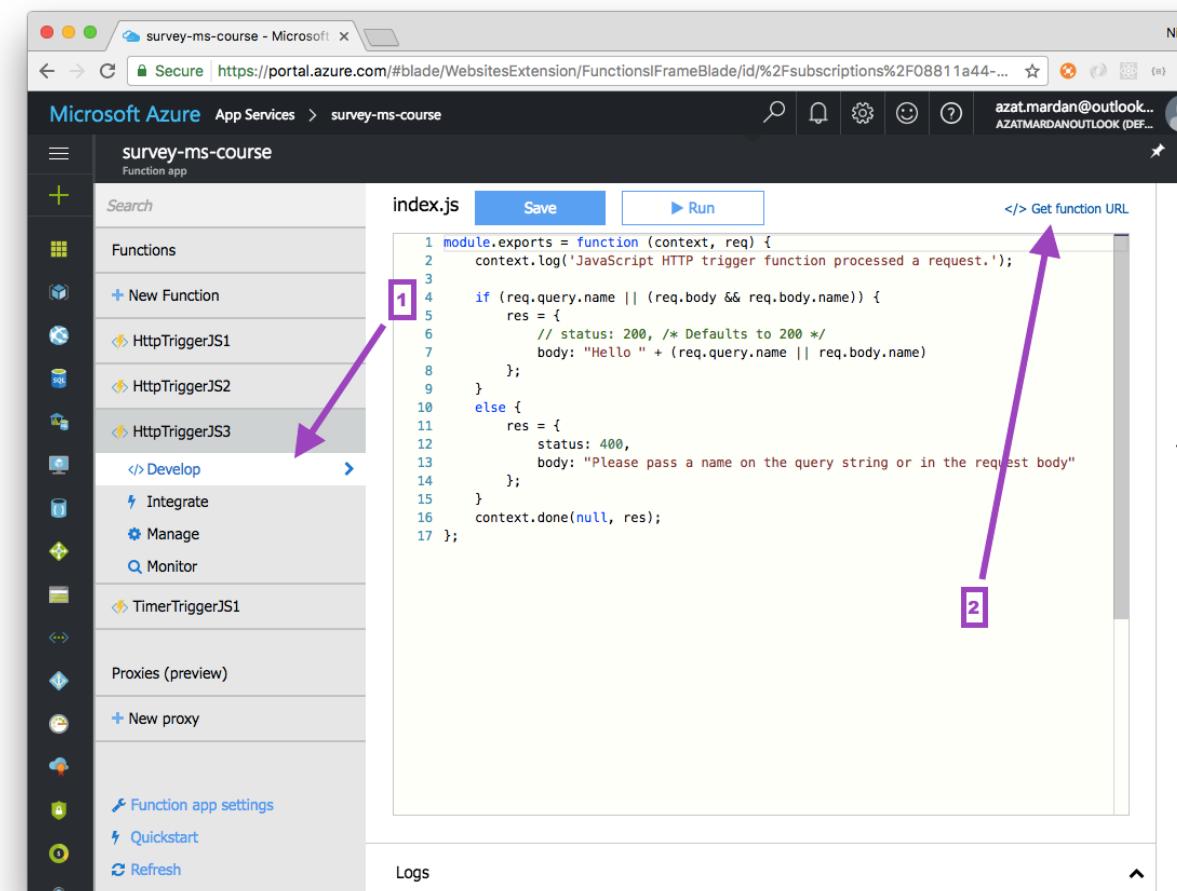
For example, for survey-ms-course <https://survey-ms-course.azurewebsites.net/api/HttpTriggerJS3?code=1uCa6dcxqcNNj9qYai7s8I/iIbbdx92A3prb1USaGKbQz5ZfUnzQ==>

Managing Keys

The function key(s) is in the "Manage" pane: Manage | Click to show | Copy



A Quicker way to Access an HTTP Function



Integrate

Azure Functions can have triggers, inputs and outputs

Triggers

Examples:

- » Time (Schedule)
- » HTTP (REST request or a webhook)
- » Events

Inputs

Examples:

- >> Azure Blob Storage
- >> External File or Table
- >> Azure Table Storage
- >> Azure Document Storage
- >> Azure Mobile Table Record
- >> Bot Framework

Outputs

Examples:

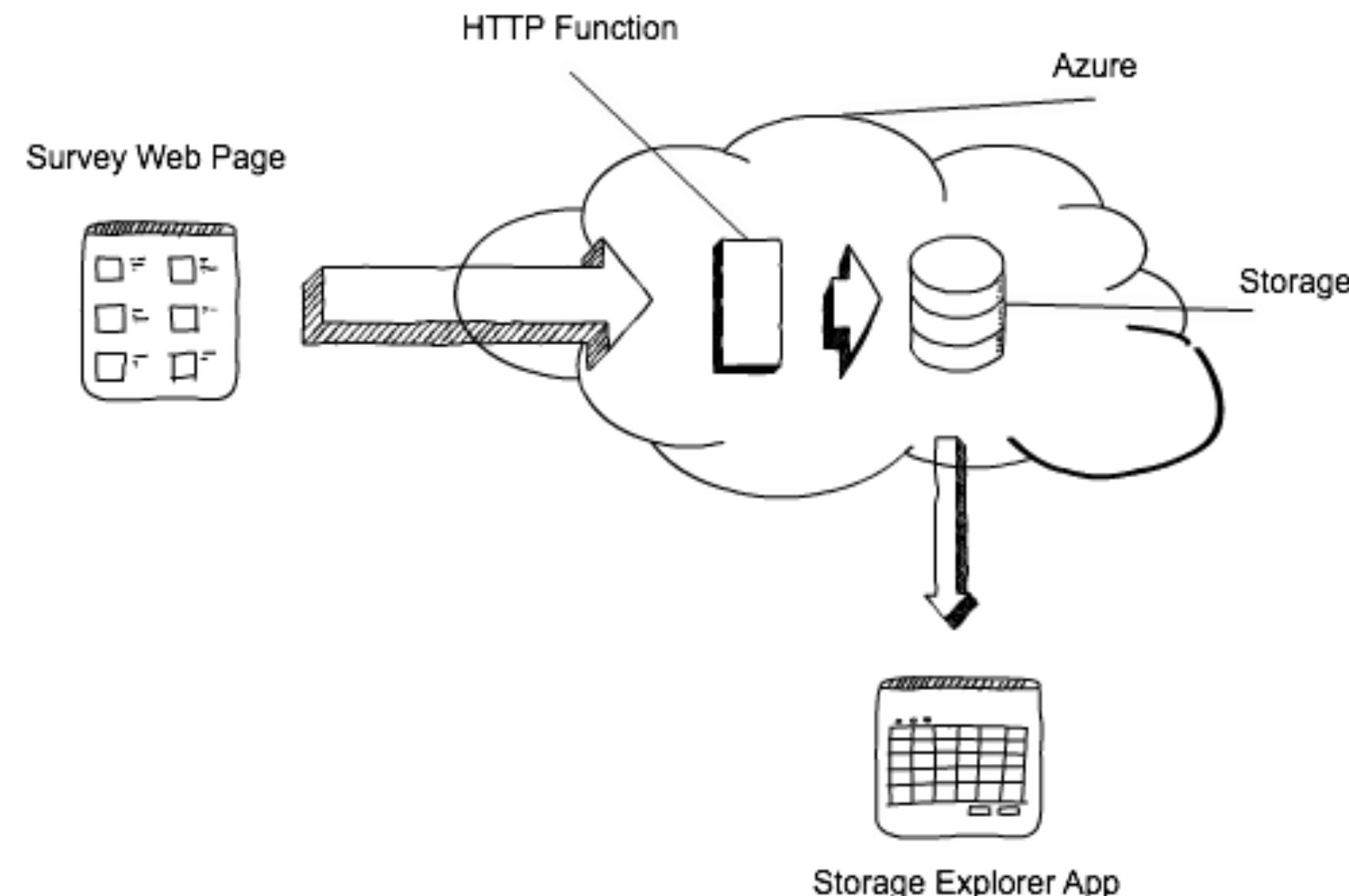
- >> Azure Event Hub
- >> Azure Table Storage
- >> Twilio
- >> SendGrid

Type	Service	Trigger	Input	Output
Schedule	Azure Functions	✓		
HTTP (REST or webhook)	Azure Functions	✓		✓*
Blob Storage	Azure Storage	✓	✓	✓
Events	Azure Event Hubs	✓		✓
Queues	Azure Storage	✓		✓
Queues and topics	Azure Service Bus	✓		✓
Tables	Azure Storage	✓	✓	✓
Tables	Azure Mobile Apps	✓	✓	
No-SQL DB	Azure DocumentDB	✓	✓	
Push Notifications	Azure Notification Hubs			✓
Twilio SMS Text	Twilio		✓	

Code Along: Serverless Survey

1. Create Azure Storage
2. Create a survey HTTP function which saves data from an HTML page to Azure Table
3. Create HTTP function to serve the HTML page (Optional)

Serverless Survey Capture



localhost:8080/func/code/surv X Ninja

localhost:8080/func/code/survey.html

HOTEL CALIFORNIA STAY SURVEY

Johnny Appleseed

Johnny.Appleseed@outlook.com

555-111-2233

RATE YOUR STAY

1
 2
 3
 4
 5

Great location and service. Highly recommend.

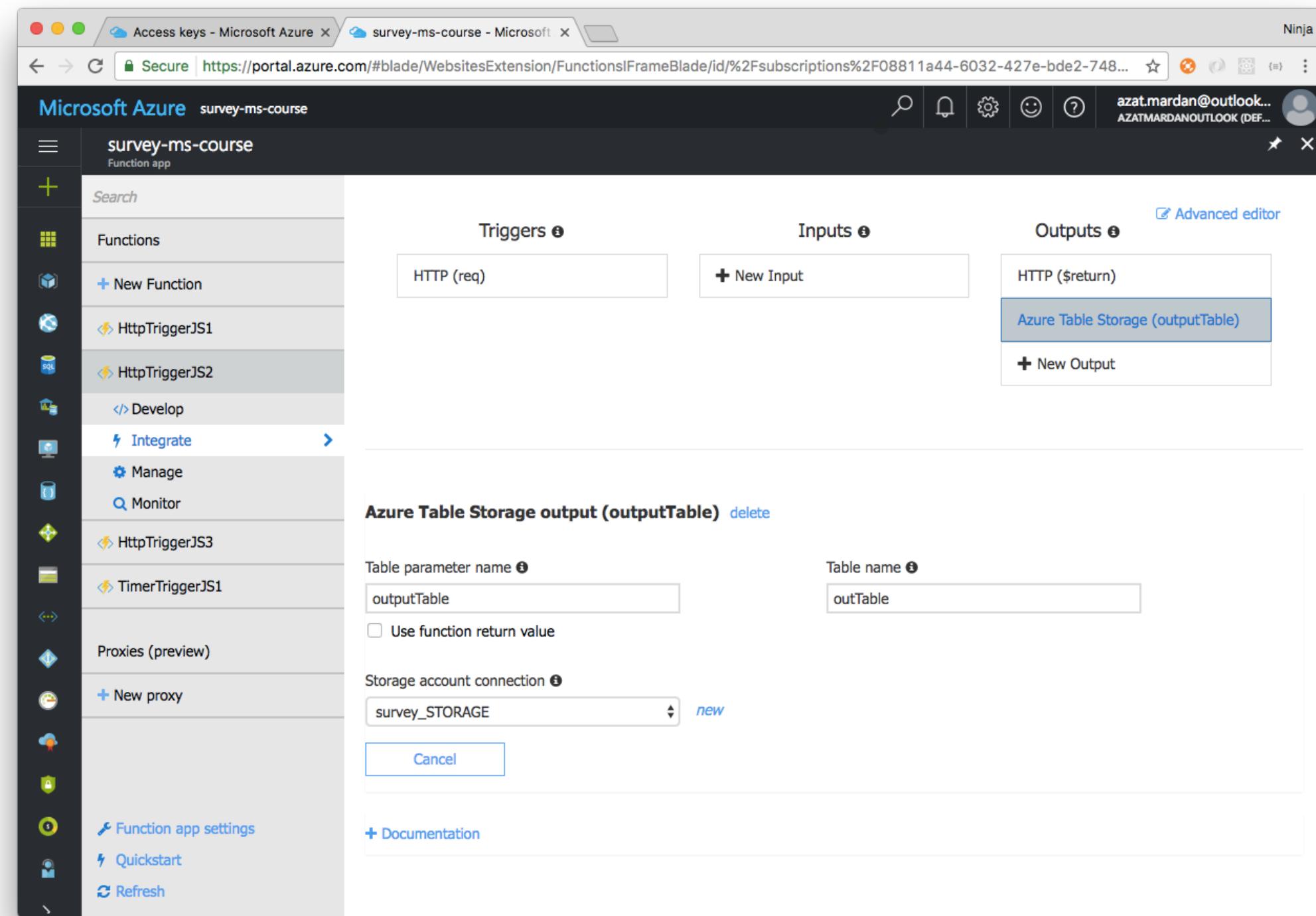
Send

Let's go! 

Use the function you created or create a new one.

Input

1. Select Integrate | Outputs | Azure Table Storage
2. Select new in "Storage account connection"
3. Select existing (or create a new) Storage account



Storage Account



Storage Accounts are filtered to the same region as your app



Create New



functionccc6cbb6a8e6

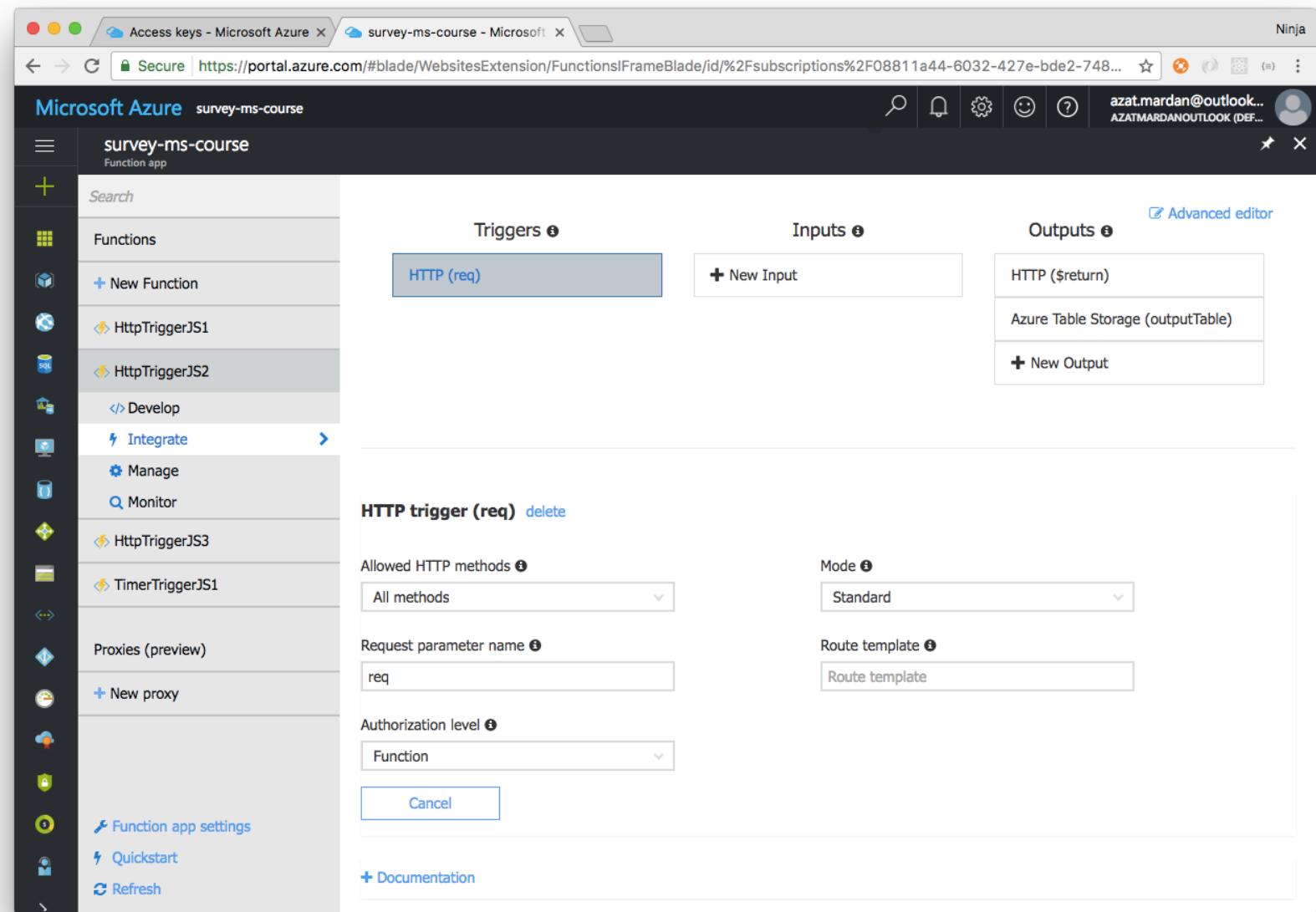
South Central US



survey

South Central US

Check HTTP trigger



Develop: index.js

```
module.exports = function (context, req) {
  const error = (err)=> {
    let res = {
      status: 400,
      body: err
    }
    context.done(null, res)
  }
  context.log('JavaScript HTTP trigger function processed a request.', req.body)
```

Develop: index.js (Cont)

```
if (!req.body) return error('no body')
const data = require('querystring').parse(req.body)
```

Develop: index.js (Cont)

```
if (!data.name) return error('no name')
if (!data.phone) return error('no phone')
if (!data.rating) return error('no rating')
if (!data.comments) return error('no comments')
if (!data.email) return error('no email')
```

Develop: index.js (Cont)

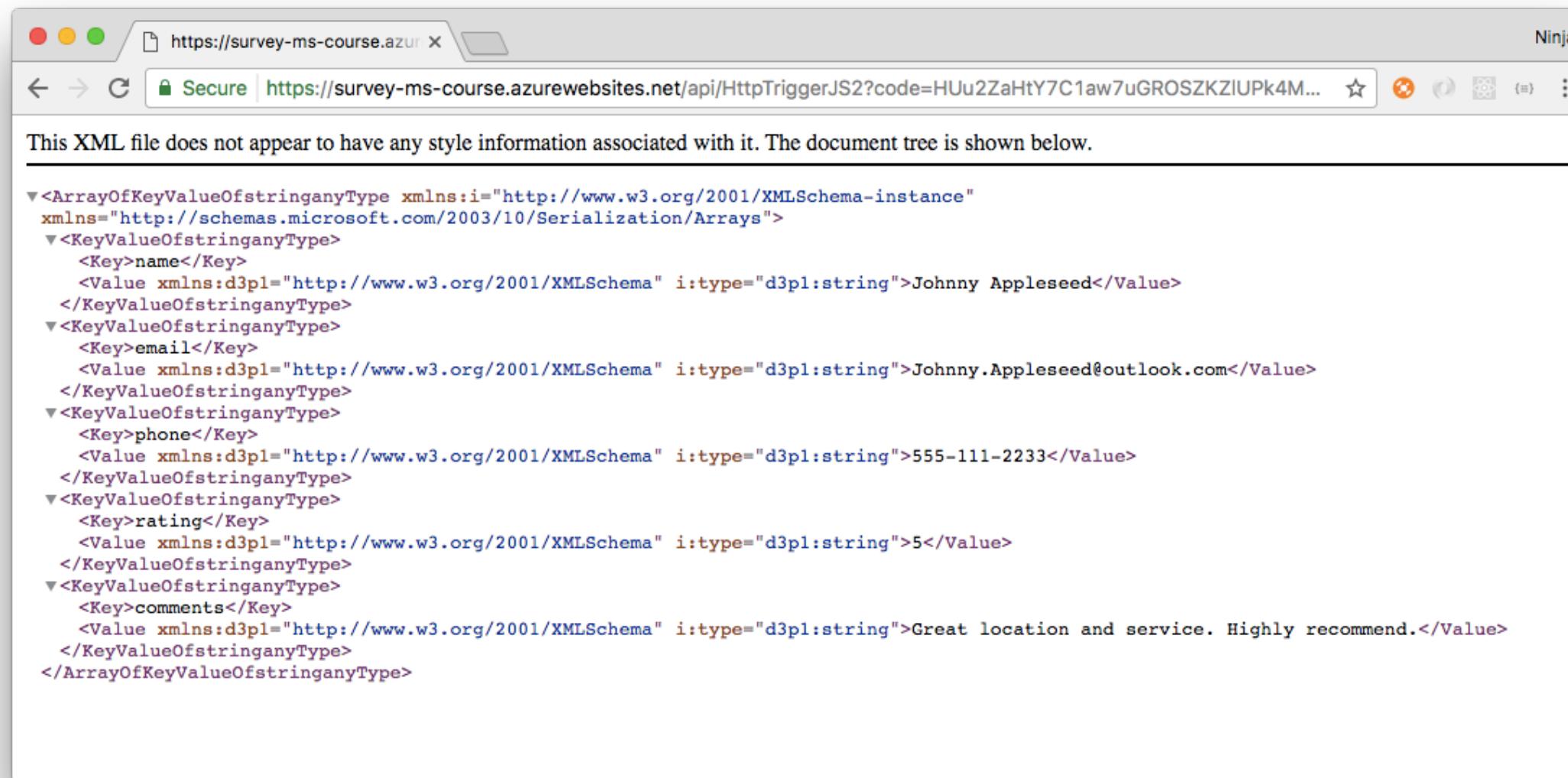
```
context.bindings.outputTable = [{

    PartitionKey: 'surkeypat',
    RowKey: Date.now(),
    name: data.name,
    phone: data.phone,
    comments: data.comments,
    email: data.email,
    rating: data.rating
}]
let res = {
    status: 201,
    body: data
}
context.done(null, res)
}
```

Send Survey

1. Open survey.html
2. Type data
3. Select "Send"
4. Observe response (must be error-free)

Survey Function Response



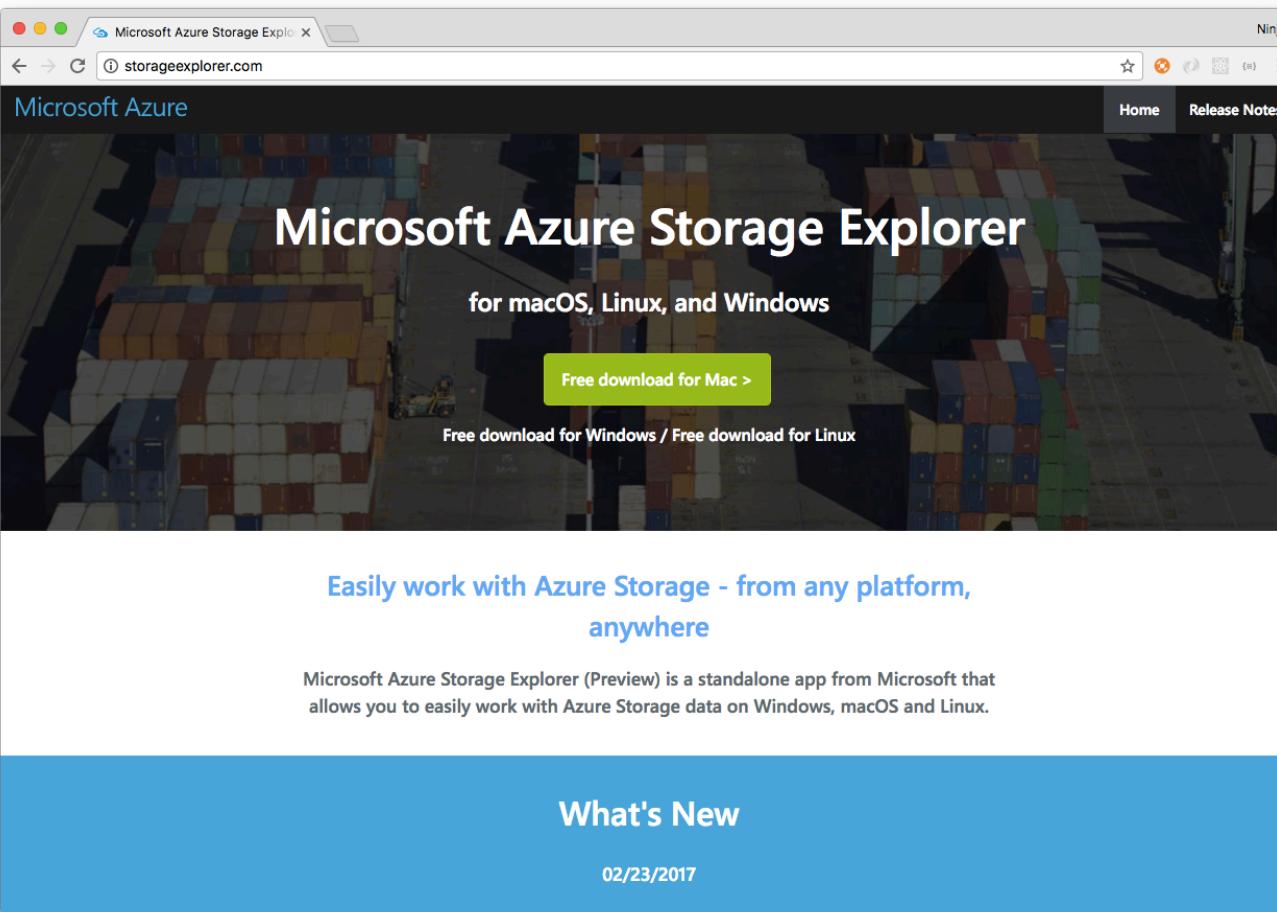
This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<ArrayOfKeyValueOfstringanyType xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <KeyValueOfstringanyType>
    <Key>name</Key>
    <Value xmlns:d3p1="http://www.w3.org/2001/XMLSchema" i:type="d3p1:string">Johnny Appleseed</Value>
  </KeyValueOfstringanyType>
  <KeyValueOfstringanyType>
    <Key>email</Key>
    <Value xmlns:d3p1="http://www.w3.org/2001/XMLSchema" i:type="d3p1:string">Johnny.Appleseed@outlook.com</Value>
  </KeyValueOfstringanyType>
  <KeyValueOfstringanyType>
    <Key>phone</Key>
    <Value xmlns:d3p1="http://www.w3.org/2001/XMLSchema" i:type="d3p1:string">555-111-2233</Value>
  </KeyValueOfstringanyType>
  <KeyValueOfstringanyType>
    <Key>rating</Key>
    <Value xmlns:d3p1="http://www.w3.org/2001/XMLSchema" i:type="d3p1:string">5</Value>
  </KeyValueOfstringanyType>
  <KeyValueOfstringanyType>
    <Key>comments</Key>
    <Value xmlns:d3p1="http://www.w3.org/2001/XMLSchema" i:type="d3p1:string">Great location and service. Highly recommend.</Value>
  </KeyValueOfstringanyType>
</ArrayOfKeyValueOfstringanyType>
```

Validate Data in the DB (Optional)

1. Install Microsoft Azure Storage Explorer
2. Put name and key
3. Validate your sent data in the database

1. Explorer



<http://storageexplorer.com>

2. Storage Name and Keys

The screenshot shows the Microsoft Azure portal interface. The left sidebar has a dark theme with various icons for different services. The main content area is titled "Access keys - Microsoft Azure" and shows the URL <https://portal.azure.com/#resource/subscriptions/08811a44-6032-427e-bde2-748a1d159168/resourceGroups/default/providers/Microsoft.Storage/storageAccounts/survey>. The top navigation bar includes "Microsoft Azure", "Storage accounts", and "survey - Access keys". The left pane lists "Storage accounts" under "azatmardanoutlook (Default Directory)" with a "NAME" column showing "azattest", "function5dca988ebc0e", "functioncc6cbb6a8e6", "microblog", and "survey". The right pane is titled "survey - Access keys" and contains sections for "Overview", "Activity log", "Access control (IAM)", "Tags", "Diagnose and solve problems", "SETTINGS" (with "Access keys" selected), "Configuration", "Shared access signature", "Properties", "Locks", "Automation script", and "BLOB SERVICE" (with "Containers", "CORS", "Custom domain", and "Encryption"). Under "SETTINGS", there is a table for "Access keys" with two entries: "key1" and "key2". The "key1" row shows the key value as `ZliofttNllhrDBml5UvXU3QbC0/XUELO90vQZk6DkU9sqA0/KYbv`. The "key2" row shows the key value as `ynVf0by+NGoELzVi9wolghAqNpSst31Qy6l+7pflikOjZlkfp7aFj4Z`.

3. Validate

Microsoft Azure Storage Explorer

An update to Storage Explorer is available.

Open View Release Notes Later Close

Search for resources

Query Import Export Add Edit Select all Column Options Delete Refresh

Collapse All

Quick Access

(Local and Attached)

- Storage Accounts
 - (SAS-Attached Services)
 - survey (External)
 - Blob Containers
 - File Shares
 - Queues
 - Tables
 - \$MetricsCapacityBlob
 - \$MetricsHourPrimaryTransa
 - \$MetricsHourPrimaryTransa
 - \$MetricsHourPrimaryTransa
 - outTable

PartitionKey	RowKey	Timestamp	name	phone	comments	email	rating
surkeypat	1489030228747	2017-03-09T03:30:28.830Z	Johnny Appleseed	555-111-2233	Great location and service. Highly recommend.	Johnny.Appleseed@outlook.com	5

Showing 1 to 1 of 1 cached items

Actions Properties

URL https://survey.table.core.windows.net:443/out
Type Table

Activities

What do you like about this tool?
What don't you like or feel is missing?

The screenshot shows the Microsoft Azure Storage Explorer interface. On the left, there's a navigation pane with 'Quick Access' and a tree view for '(Local and Attached)' storage accounts, including 'Storage Accounts' (with 'survey' as a child), 'Tables' (with several metrics tables listed), and 'BLOB Containers', 'File Shares', and 'Queues'. The main area displays a table named 'outTable' with one row. The columns are PartitionKey, RowKey, Timestamp, name, phone, comments, email, and rating. The row data is: surkeypat, 1489030228747, 2017-03-09T03:30:28.830Z, Johnny Appleseed, 555-111-2233, Great location and service. Highly recommend., Johnny.Appleseed@outlook.com, 5. Below the table, it says 'Showing 1 to 1 of 1 cached items'. At the bottom, there are sections for 'Activities' and feedback questions: 'What do you like about this tool?' and 'What don't you like or feel is missing?'. The top bar has a message 'An update to Storage Explorer is available.' and buttons for 'Open', 'View Release Notes', 'Later', and 'Close'.

Congrats! You implemented cost effective and almost maintenance-free survey.

Debug

<https://{{func-app-name}}.scm.azurewebsites.net/DebugConsole>

For example, <https://survey-ms-course.scm.azurewebsites.net/DebugConsole>

Functions Files: \site\wwwroot

Logs: LogFiles\Application\Functions\function

Install npm Packages

<https://survey-ms-course.scm.azurewebsites.net/DebugConsole>

cd site

cd wwwroot

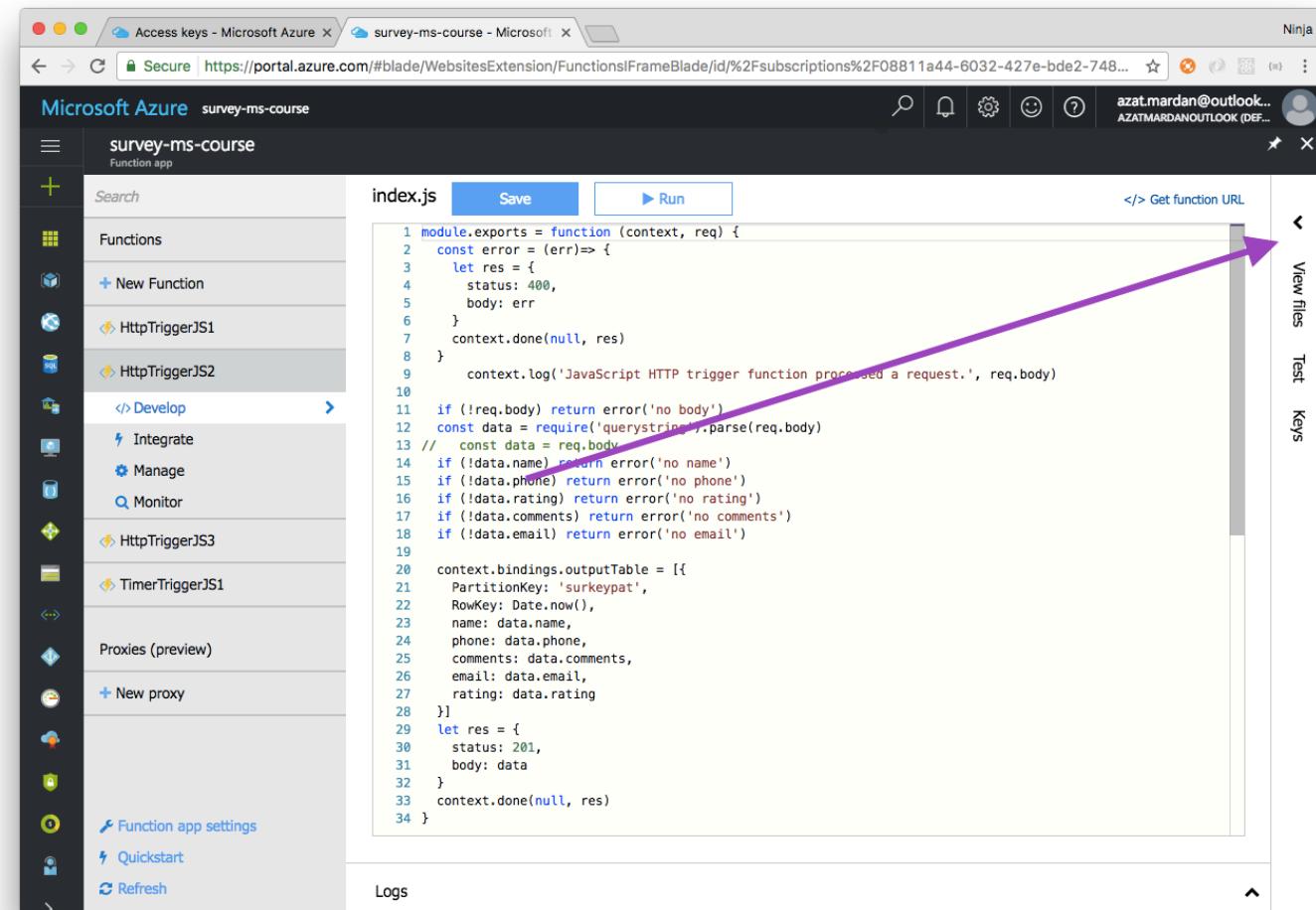
cd HttpTriggerJS1

npm init -y

npm i {package-name} -SE

Files

You can view, add/upload, delete files which belong to a function.



The screenshot shows the Microsoft Azure portal interface for a Function app named "survey-ms-course". On the left, the sidebar lists various functions: "HttpTriggerJS1" (selected), "HttpTriggerJS2", "HttpTriggerJS3", and "TimerTriggerJS1". The main area displays the code for "index.js". The code is as follows:

```
1 module.exports = function (context, req) {
2     const error = (err)=> {
3         let res = {
4             status: 400,
5             body: err
6         }
7         context.done(null, res)
8     }
9     context.log('JavaScript HTTP trigger function processed a request.', req.body)
10
11    if (!req.body) return error('no body')
12    const data = require('querystring').parse(req.body)
13 //    const data = req.body
14    if (!data.name) return error('no name')
15    if (!data.phone) return error('no phone')
16    if (!data.rating) return error('no rating')
17    if (!data.comments) return error('no comments')
18    if (!data.email) return error('no email')
19
20    context.bindings.outputTable = [
21        PartitionKey: 'surkeypat',
22        RowKey: Date.now(),
23        name: data.name,
24        phone: data.phone,
25        comments: data.comments,
26        email: data.email,
27        rating: data.rating
28    ]
29    let res = {
30        status: 201,
31        body: data
32    }
33    context.done(null, res)
34 }
```

Test

View files **Test** Keys >

HTTP method
POST

Query
code HUu2ZaHtY7C1aw7uGR! ✖
[+ Add parameter](#)

Headers
There are no headers
[+ Add header](#)

Request body
1 name=Azat

Output ✖ Status: 400 Bad Request
"no phone"

▶ Run

Logs

Logs

■■ Pause

■ Clear

□ Copy logs

↗ Expand



```
2017-03-09T04:10:17 No new trace in the past 2 min(s).
2017-03-09T04:11:17 No new trace in the past 6 min(s).
2017-03-09T04:12:17 No new trace in the past 7 min(s).
2017-03-09T04:13:17 No new trace in the past 8 min(s).
2017-03-09T04:14:17 No new trace in the past 9 min(s).
2017-03-09T04:14:20.939 Function started (Id=c5af5b1d-f4a2-46eb-a6ee-a6ab1adc79f1)
2017-03-09T04:14:21.267 JavaScript HTTP trigger function processed a request. { name: 'Azure' }
2017-03-09T04:14:21.316 Function completed (Success, Id=c5af5b1d-f4a2-46eb-a6ee-a6ab1adc79f1)
2017-03-09T04:14:35.688 Function started (Id=d9c37fb2-0f92-4061-a1ef-403cc4d54406)
2017-03-09T04:14:35.688 JavaScript HTTP trigger function processed a request. name=Azat
2017-03-09T04:14:35.688 Function completed (Success, Id=d9c37fb2-0f92-4061-a1ef-403cc4d54406)
```

Azure CLI

>> <https://docs.microsoft.com/en-us/azure/xplat-cli-install>

>> <https://github.com/Azure/azure-cli>

The End 🏁