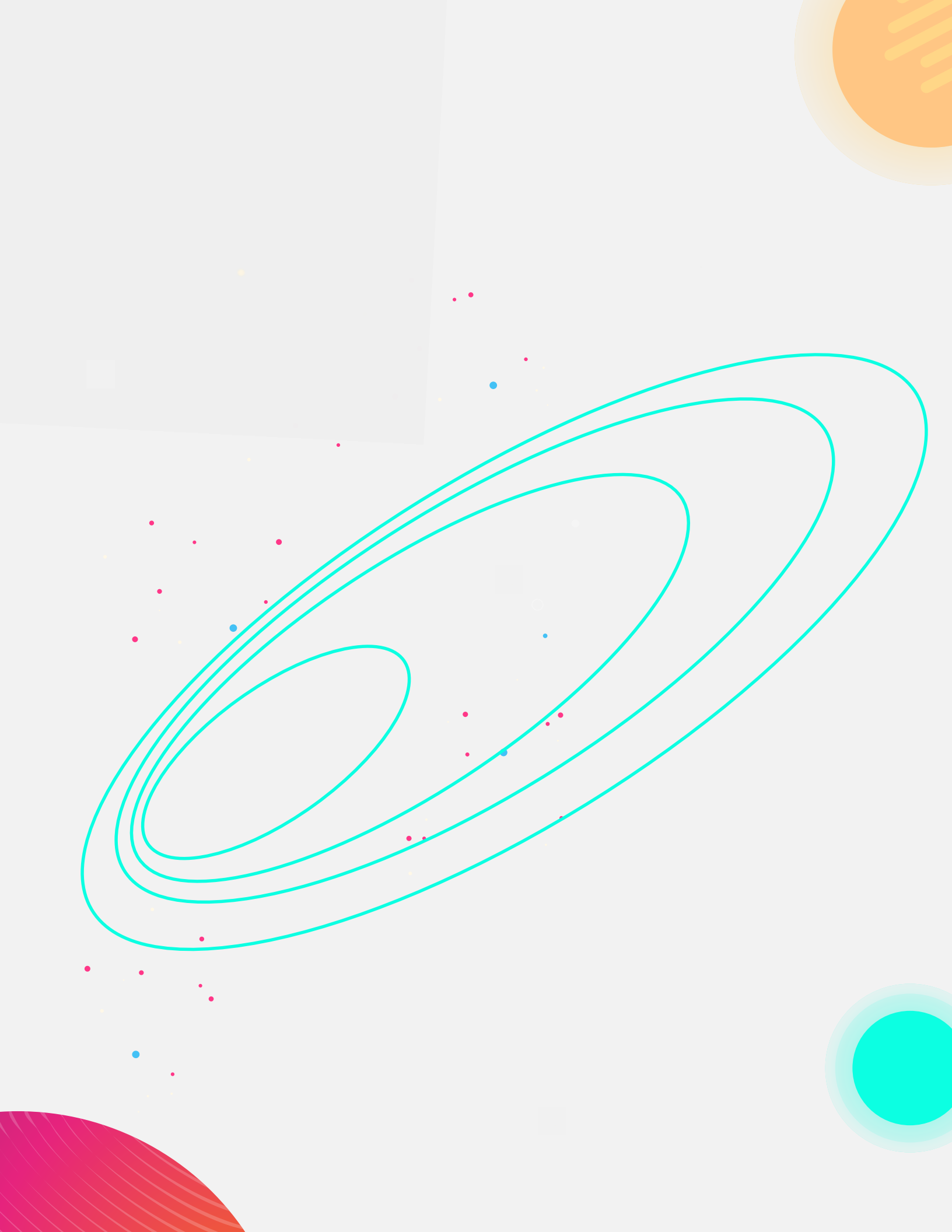


Azure Tips and Tricks: Productivity

azuredev.tips



Introduction

Hi, folks!



When I reflect back on Azure Tips and Tricks a year ago, I was only thinking that I'd write a couple of posts and move on. Fast-forward to today, the collection has grown to over 150+ tips, as well as videos, conference talks, and now an e-book spanning the entire universe of the Azure platform. What you are currently reading is a special collection of tips based on page views of the entire series over the last year. Before we dive in, you'll notice my pixelated form as you turn each page.

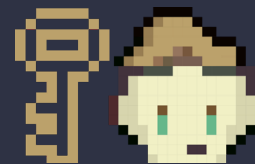
These represent:



Something I found interesting and you may too.



Additional resources to get the most out of this tip.



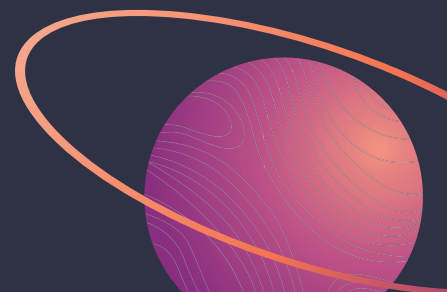
A key takeaway from the tip.

You can stay up to date with the latest Azure Tips and Tricks at:

- Blog - azuredev.tips
- Videos - videos.azuredev.tips
- eBook - ebook.azuredev.tips
- Survey - survey.azuredev.tips

I hope you enjoy reading the eBook as much as I did writing it.

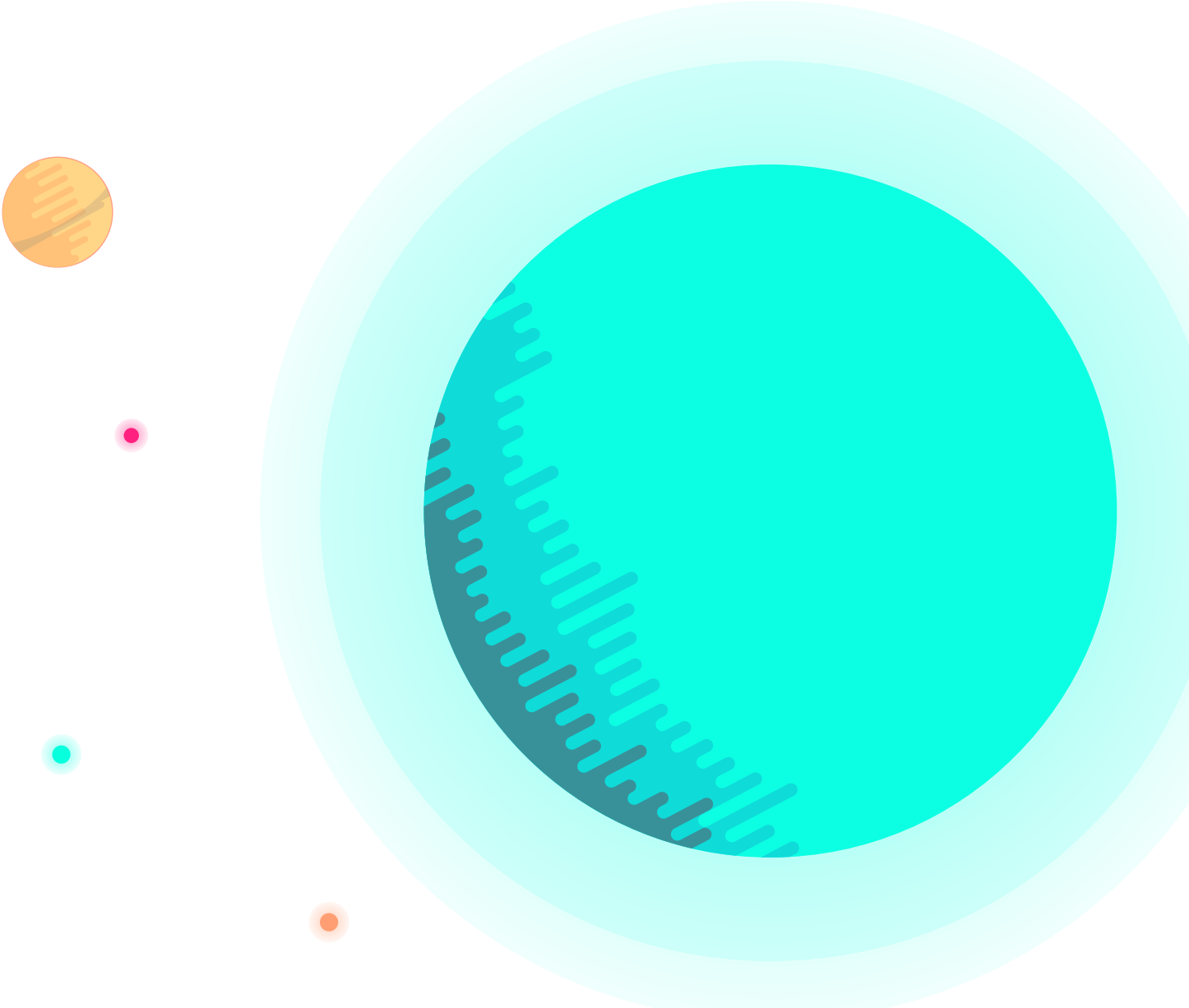
Thanks,
Michael Crump (@[mbcrump](https://twitter.com/mbc Crump))





Productivity

If you're reading this e-book, then you certainly understand the spirit of what I originally wanted to achieve with Azure Tips and Tricks - simply to be more productive with Azure. We'll start by revisiting the first tip that I ever wrote describing how you can use keyboard shortcuts within the Azure Portal to navigate more effectively. We'll also cover how you can apply tags to your Azure resources to logically organize them by categories. We'll wrap up with using Azure Cloud Shell, which provides an interactive, browser-accessible shell for managing Azure resources, and how you can quickly take advantage of it with Visual Studio Code in the browser or on your local development machine.



Azure Portal Keyboard Shortcuts

Developers love keyboard shortcuts and there are plenty of keyboard shortcuts in the Azure platform. You can see a list by logging into the Azure Portal, clicking on the question mark (or help icon), and selecting Keyboard Shortcuts.

You will see that you have the following keyboard shortcuts available:

Actions

CTRL+/ 	Search blade menu items
ALT+SHIFT+Up	Move favorites up
ALT+SHIFT+Down	Move favorites down
G+/ 	Search resources (global)
G+N	Create a new resource
G+B	Open the 'More services' pane

Navigation

G+,	Move focus to command bar
G+.	Toggle focus between top bar and side bar

Go to

G+D	Go to dashboard
G+A	Move favorites up
G+R	Move favorites down
G+number	Search resources (global)



Continue checking the site as new ones are being added all the time!

Use Tags to Quickly Organize Azure Resources

You can utilize tags to quickly organize Azure Resources. For example, if you'd like to have a set of Resources for "Production" and another for "Dev", then you can quickly do that.



Remember this!

Tags are user-defined key/value pairs which can be placed directly on a resource or a resource group.

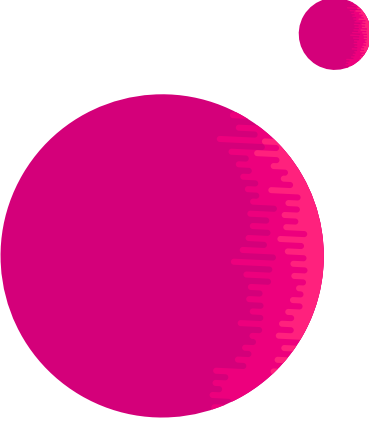
Head over to the Azure Portal and select a service. In my example, I'm going to select a Web App that I want to tag as a Production App. Select the Tags menu and provide a Name and Value as shown below.

The screenshot shows the 'Tags' page for the resource 'mvcappdemovslive' in the Azure Portal. The left sidebar contains navigation links: Overview, Activity log, Access control (IAM), Tags (highlighted), and Diagnose and solve problems. The main area has a 'Save' button and an information box stating: 'Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. [Learn more](#)'. Below this, there are two input fields: '* Name' with the value 'Environment' and '* Value' with the value 'Production'. A 'No tags' section is visible at the bottom.

I selected **Environment** and gave it the value of **Production**. I then clicked **Save**. I could also do this for other Production resources, and even tag the appropriate ones with **Dev**. I can now take advantage of this ability by going to **More Services**, typing **Tags**, and clicking on the Environment: Production as shown below.

The first screenshot shows the 'Tags' page for 'mvcappdemovslive' with a dropdown menu showing '2 subscriptions'. The second screenshot shows the 'Environment: Production' tag page with a table of resources and subscriptions. The table has columns for 'NAME' and 'SUBSCRIPTION'. The resources listed are 'mc-webstarter' and 'mvcappdemovslive'. The subscriptions listed are 'Visual Studio Enterprise' and 'Michael's Internal Subscription'. The 'Environment: Production' tag is highlighted with a red '2'.

NAME	SUBSCRIPTION
mc-webstarter	Visual Studio Enterprise
mvcappdemovslive	Michael's Internal Subscription

- 
1. Results from searching "Tags"
 2. Our Production Environment we just setup
 3. List all the Web Apps with the Production Environment Tag
 4. Pin the Blade to our Azure Portal Main Page

If you pin the blade (by pressing the pin in step 4) you'll see the following on your Azure Portal dashboard:



Recap: Make your life easier by applying tags to your Azure resources to logically organize them by categories.



Environment : Production

You can even interact with **Tags** using Azure CLI 2.0. For example, I can type **az tag list -o json** to list all the tags associated with an account.

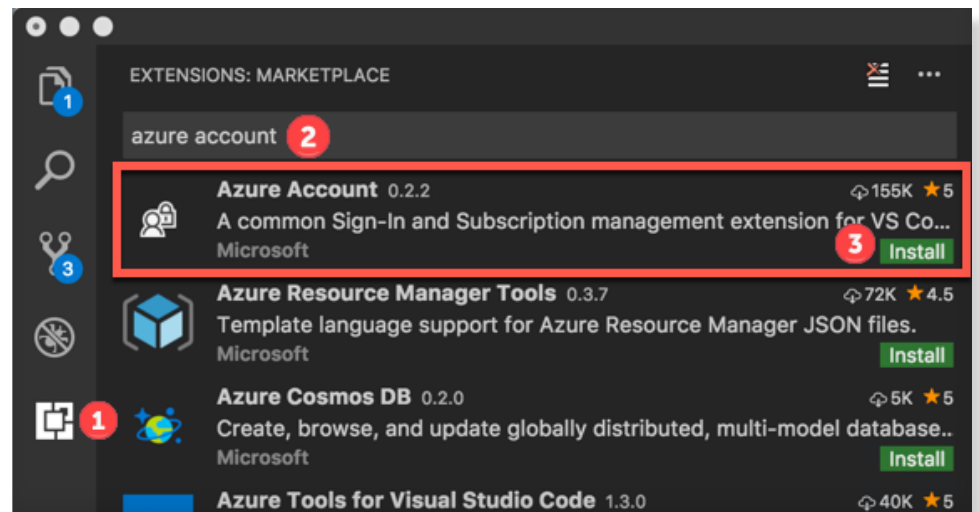
```
michael@Azure:~$ az tag list
[
  {
    "count": {
      "type": "Total",
      "value": 2
    },
    "id":
"/subscriptions/c0e5fb0f-7461-4b04-9720-63fe407b1bdb/tagNames/Environment",
    "tagName": "Environment",
    "values": [
      {
        "count": {
          "type": "Total",
          "value": 1
        },
        "id":
"/subscriptions/c0e5fb0f-7461-4b04-9720-63fe407b1bdb/tagNames/Environment/tagValues
/Dev",
        "tagValue": "Dev"
      },
      {
        "count": {
          "type": "Total",
          "value": 1
        },
        "id":
"/subscriptions/c0e5fb0f-7461-4b04-9720-63fe407b1bdb/tagNames/Environment/tagValues
/Production",
        "tagValue": "Production"
      }
    ]
  }
]
```



Azure Cloud Shell is an interactive, browser-accessible shell for managing Azure resources. Linux users can opt for a Bash experience, while Windows users can opt for PowerShell.

Add Azure Cloud Shell to Visual Studio Code

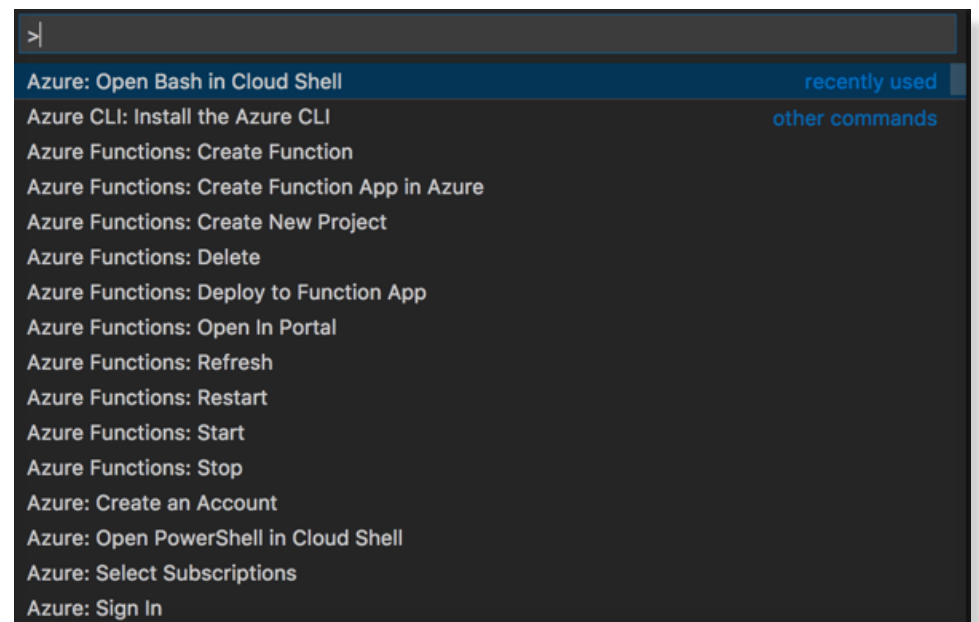
To add Azure Cloud Shell to VS Code, click on Extensions and search for [azure account](#). Install it as shown below.



Once installed, go to View -> Command Palette and type [Open Bash in Cloud Shell](#).



Note: You can also open PowerShell in Cloud Shell with this extension!



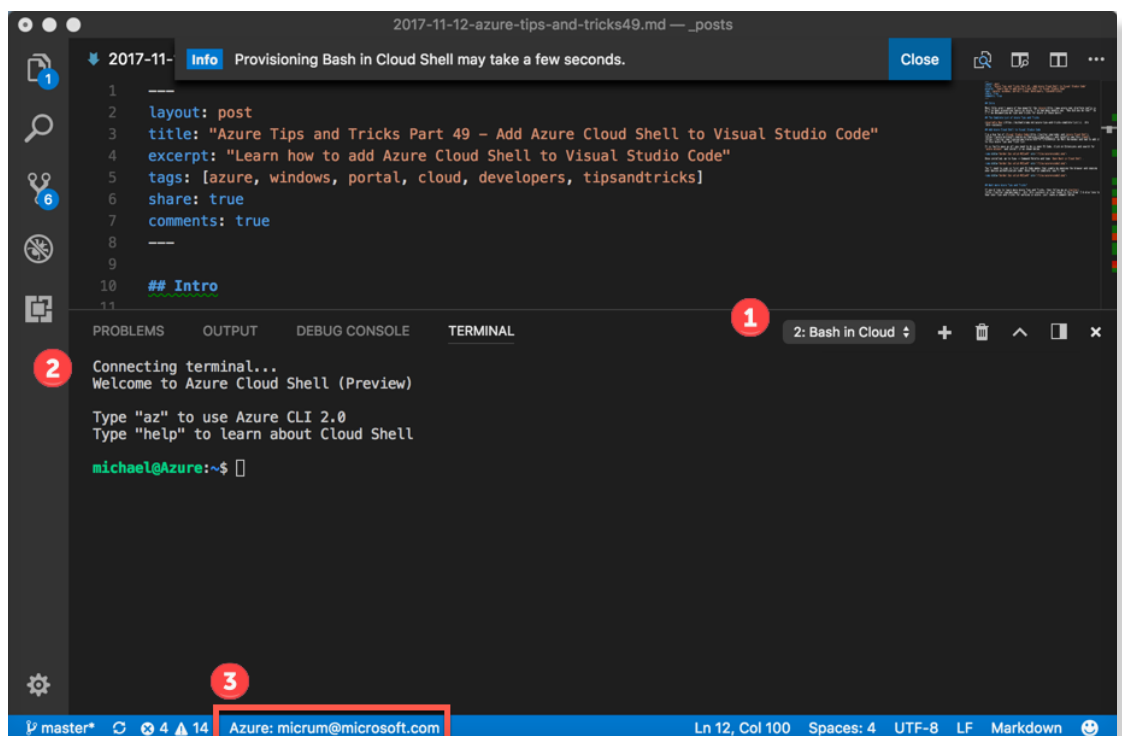


You'll need to sign in first, and Visual Studio Code makes that simple by opening the browser and copying your device authentication code. Once that is complete, you'll see:

Visual Studio Code

You have signed in to the Visual Studio Code application on your device. You may now close this window.

Go back to View -> Command Palette and select **Open Bash in Cloud Shell** again and it should spin up as shown below.

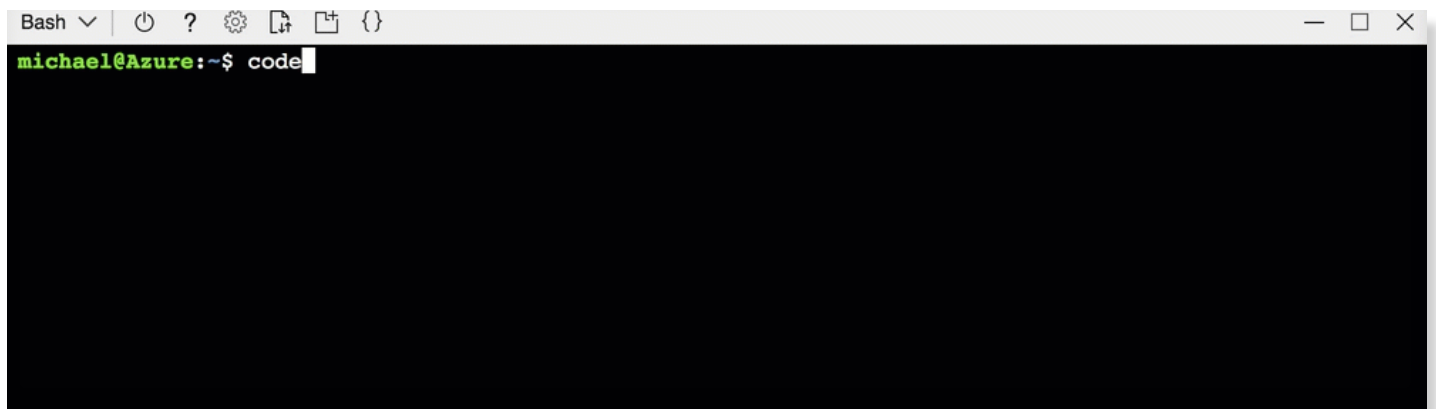


Very cool! If you want to see the source code for the app it can be found [here](#).

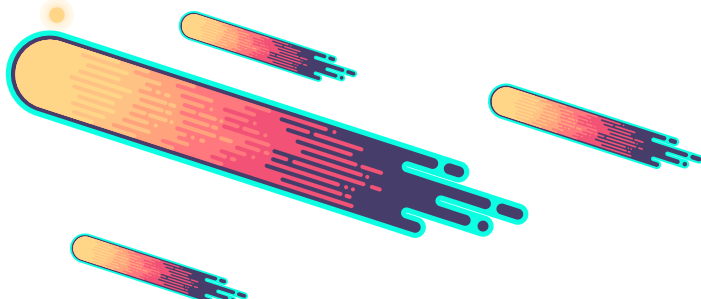
Quickly Edit Files Within Azure Cloud Shell Using Visual Studio Code That You Know and Love

Did you know that you can access Visual Studio Code within a Cloud Shell instance?

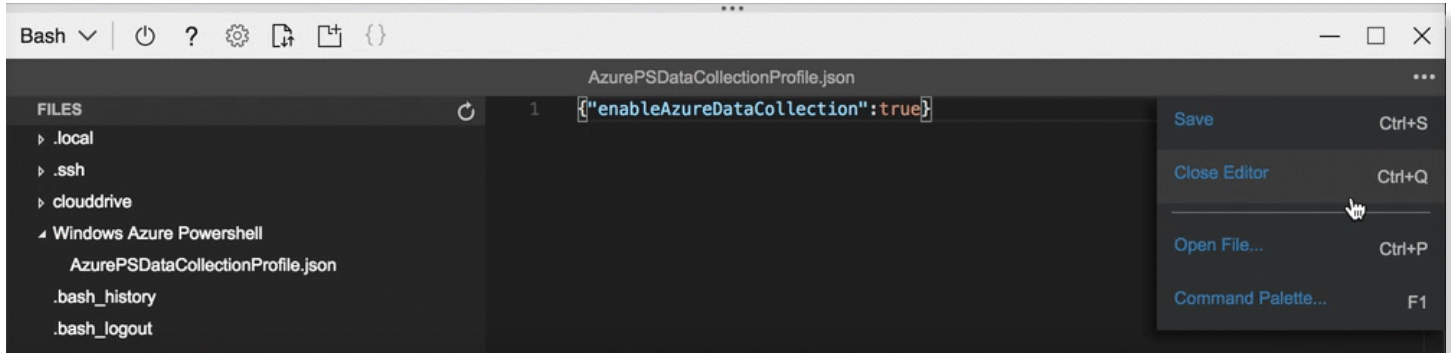
I'm sure by now everyone has used the lovely [Visual Studio Code editor](#) in some application before, but you may not be aware that you can use the editor within Cloud Shell without installing anything. To give this a spin, open up Cloud Shell and type `code` . and you'll see the following:

A screenshot of the Azure Cloud Shell terminal window. The window has a title bar with 'Bash' and several icons. The terminal content shows the prompt 'michael@Azure:~\$' followed by the command 'code' entered. The background of the terminal is black, and the text is green and white.

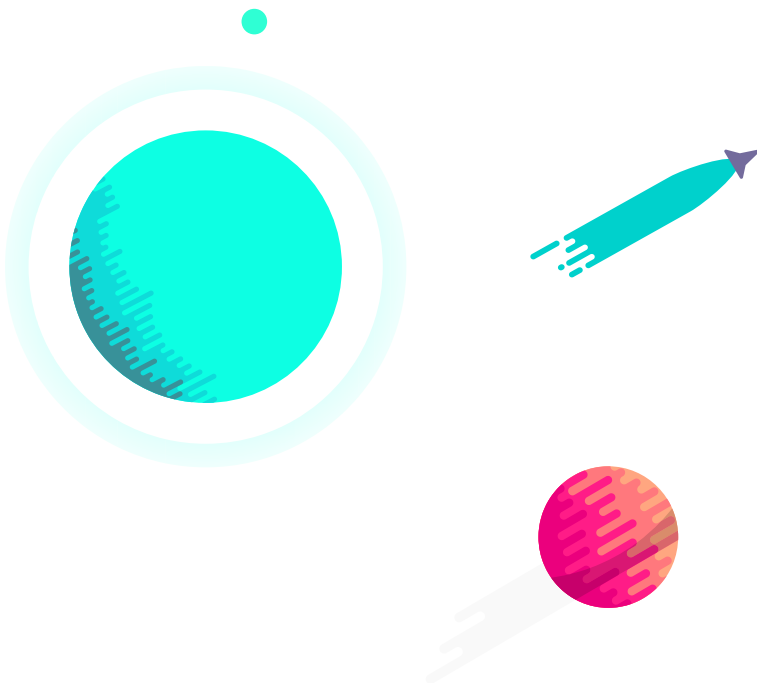
Notice that you can do things such as navigate directories, and also view files with the same syntax used in VS Code. You can easily save and close the editor, open a file outside the current working directory, and open the command palette.



If you open the command palette you'll see a very familiar list of commands that you've probably used in the editor on your desktop.



And since this is based upon the open-source Monaco project that powers Visual Studio Code, you can expect we'll see more features added over time. As of the publication time of this eBook, it automatically includes authorization for pre-installed open source tools like Terraform, Ansible, and InSpec. So what are you waiting for? [Go check out now!](#)



Conclusion

There are 130+ additional tips waiting on you that cover additional topics such as :

- App Services
- CLI
- Cloud Shell
- Cognitive Services
- Containers
- Cosmos DB
- Functions
- IoT
- Logic Apps
- Portal
- PowerShell
- Productivity
- Storage
- SQL and Search

Find all of these and more at azuredev.tips

Don't forget that if you are modernizing an existing application or building a new app, you can get started Azure for free and get:

- \$200 credit toward use of any Azure service
- 12 months of free services—including compute, storage, network, and database
- 25+ always-free services—including serverless, containers, and artificial intelligence

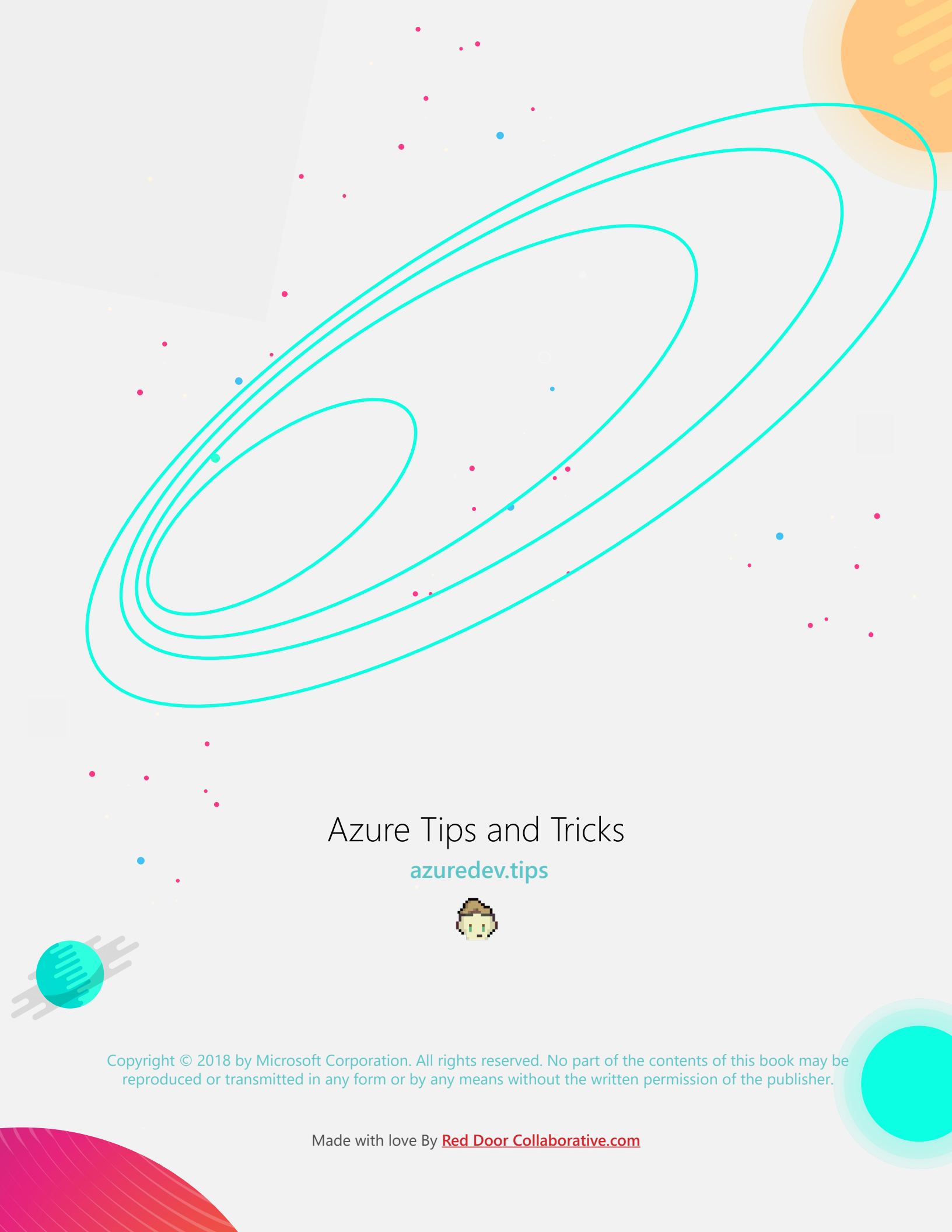
[Start free](#)

Until next time,

Michael Crump [@mbcrump](#)

signing off...





Azure Tips and Tricks

azuredev.tips



Copyright © 2018 by Microsoft Corporation. All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

Made with love By [Red Door Collaborative.com](https://RedDoorCollaborative.com)