

AAISIG | AWS Cloud Computing Workshop | Start-Up Guide



AY 2023/2024 | 19 July 2023 | Workshop Taught By: Luke Thomas Lim

 Guide Prepared By: Poh Jie Ren Luke



IMPORTANT NOTES - PLEASE READ!!:

- Please use Windows and not Mac. This workshop's content is only suitable for Windows users.
- Project codes will be shared at the beginning of the workshop.
- Please prepare your credit/debit card details ONLY for the registering of the AWS Account.
- **NO CHARGES WILL BE INCURRED** as you follow along and do the 2 projects presented **UNLESS** your existing AWS account is more than 12 months old (Free Tier Expiration, No guarantee that charges will be not be incurred)

AWS Set Up



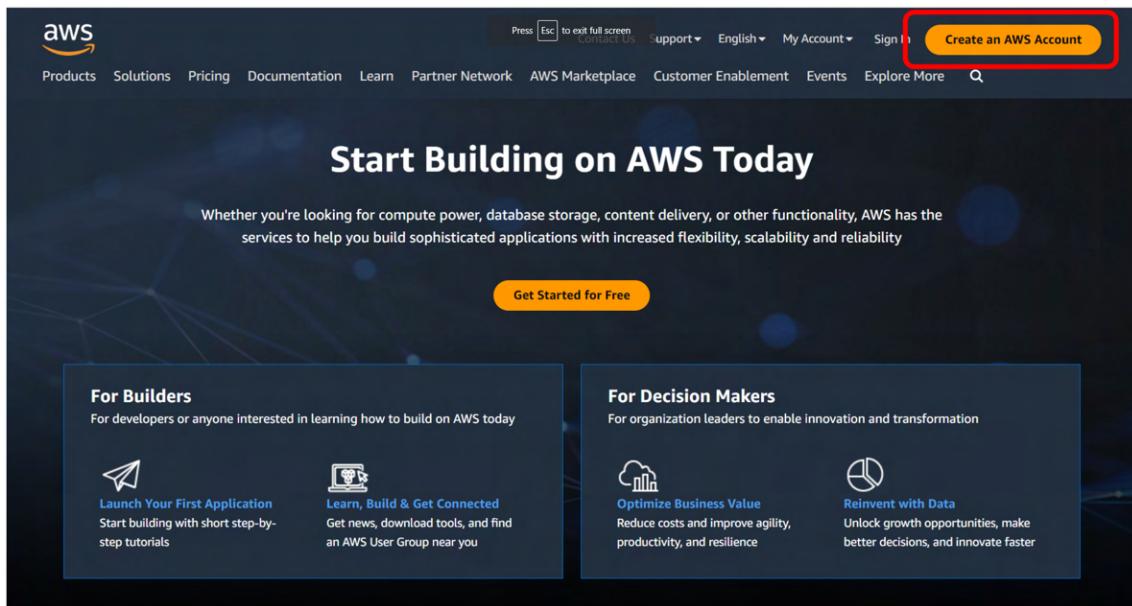
Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud, offering over 200 fully featured services from data centers globally. Millions of customers—including the fastest-growing startups, largest enterprises, and leading government agencies—are using AWS to lower costs, become more agile, and innovate faster.



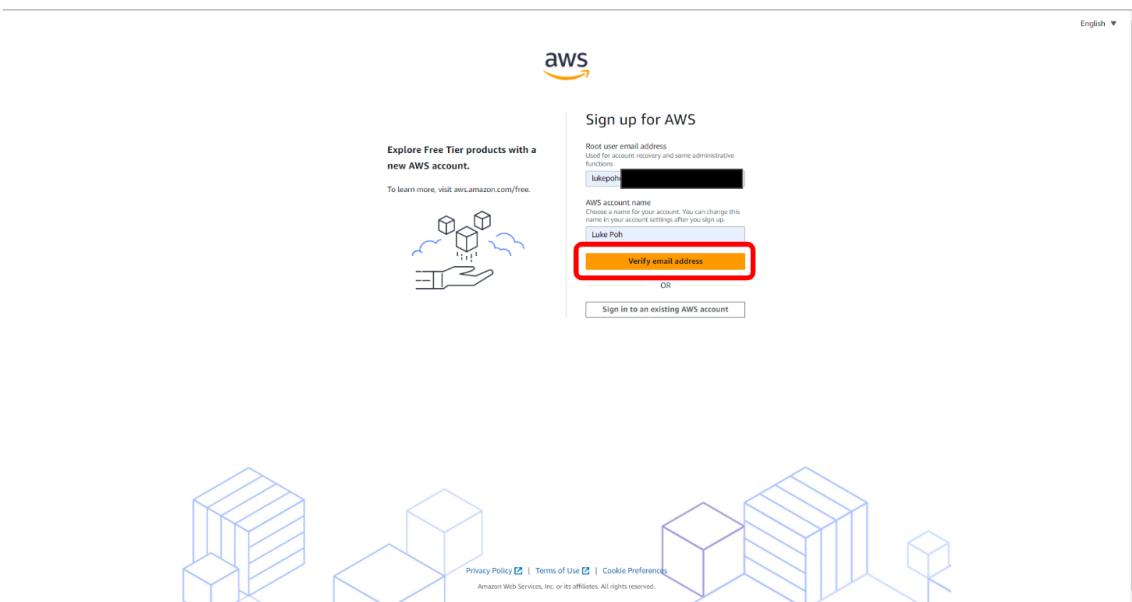
Cloud computing is the on-demand delivery of IT resources over the Internet with pay-as-you-go pricing. Instead of buying, owning, and maintaining physical data centers and servers, you can access technology services, such as computing power, storage, and databases, on an as-needed basis from a cloud provider like Amazon Web Services (AWS). Organizations of every type, size, and industry are using the cloud for a wide variety of use cases, such as data backup, disaster recovery, email, virtual desktops, software development and testing, big data analytics, and customer-facing web applications. For example, healthcare companies are using the cloud to develop more personalized treatments for patients. Financial services companies are using the cloud to power real-time fraud detection and prevention. And video game makers are using the cloud to deliver online games to millions of players around the world.

▼ Create AWS Account

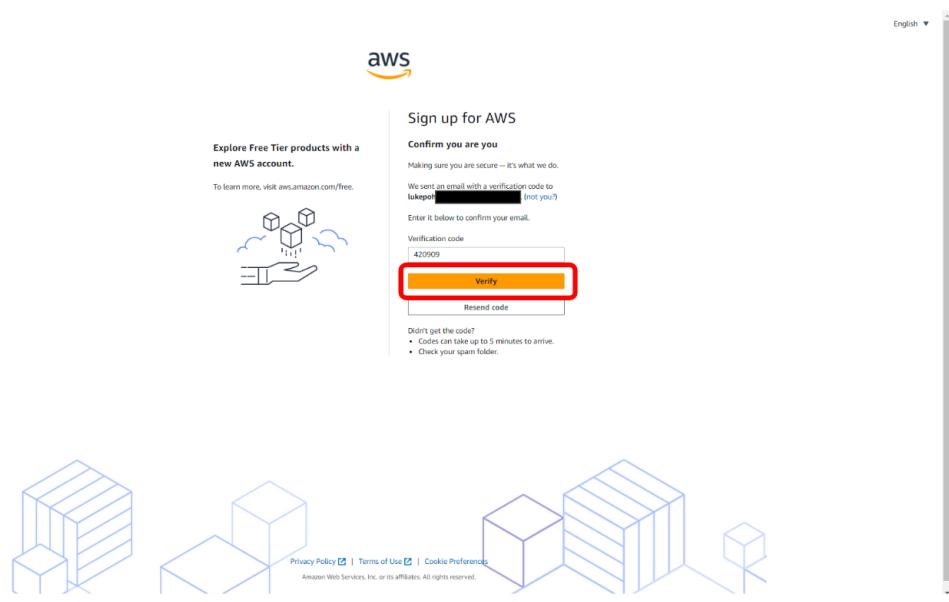
1. In your browser, open the [AWS Home Page](#)
2. Choose **Create an AWS account**



3. Enter your account information, and then choose **Verify email address**.
This will send a verification code to your specified email address.



4. Enter your verification code, and then choose **Verify**.



5. Enter a strong password for your root user, confirm it, and then choose Continue. AWS requires that your password meet the following conditions:
 - a. It must have a minimum of 8 characters and a maximum of 128 characters.
 - b. It must include a minimum of three of the following mix of character types: uppercase, lowercase, numbers, and ! @ # \$ % ^ & * () <> [] {} | _+= symbols.
 - c. It must not be identical to your AWS account name or email address.
6. Between Business or Personal, choose Personal. The difference between these options is the information that AWS asks you for. Both account types have the same features and functions.
Enter your personal information. Refer to the recommendations in the [Prerequisites](#) section about the email address and phone number.
Read and accept the [AWS Customer Agreement](#). Be sure that you read and understand the terms of the AWS Customer Agreement.
There will be an option to access a Basic Support plan at no cost later on.
Choose Continue.

Sign up for AWS

Contact Information

How do you plan to use AWS?

Business - for your work, school, or organization
 Personal - for your own projects

Who should we contact about this account?

Full Name: Polle Ren Luke

Phone Number: +65 [REDACTED]

Country or Region: Singapore

Address: [REDACTED]
[REDACTED]

City: Singapore

State, Province, or Region: Singapore

Postal Code: [REDACTED]

I have read and agree to the terms of the AWS Customer Agreement [\[link\]](#)

Continue (Step 2 of 3)

- At this point, you'll receive an email message to confirm that your AWS account is ready to use.

You can sign in to your new account by using the email address and password you provided during sign up. However, you can't use any AWS services until you finish activating your account.

Enter information about your payment method. If you want to use a different address for billing purposes, choose **Use a new address**.

Choose **Verify and Continue**.

Sign up for AWS

Secure verification

We will not charge you for usage below AWS Free Tier limits. We may temporarily hold up to \$1 USD (or an equivalent amount in local currency) as a pending transaction for 2-5 days to verify your identity.

Verify and Continue (Step 3 of 3)

Billing Information

Credit or Debit card number: [REDACTED]

VISA

AWS accepts all major credit and debit cards. To learn more about payment options, review our FAQ.

Expiration date: [REDACTED]

Cardholder's name: [REDACTED]

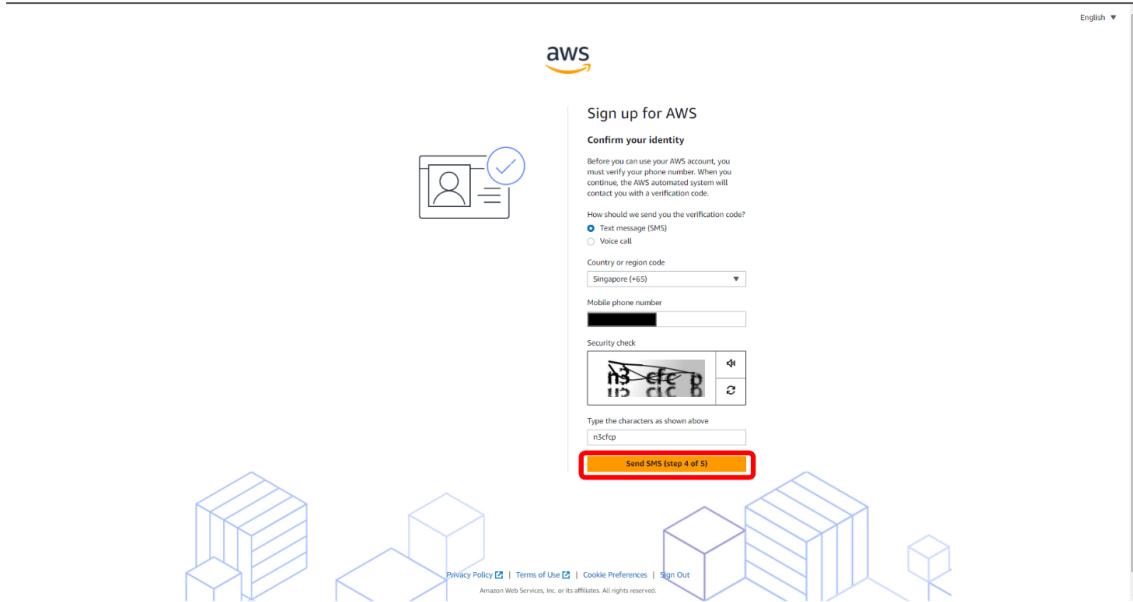
Billing address:

Use my contact address [REDACTED]
 Use a new address [REDACTED]

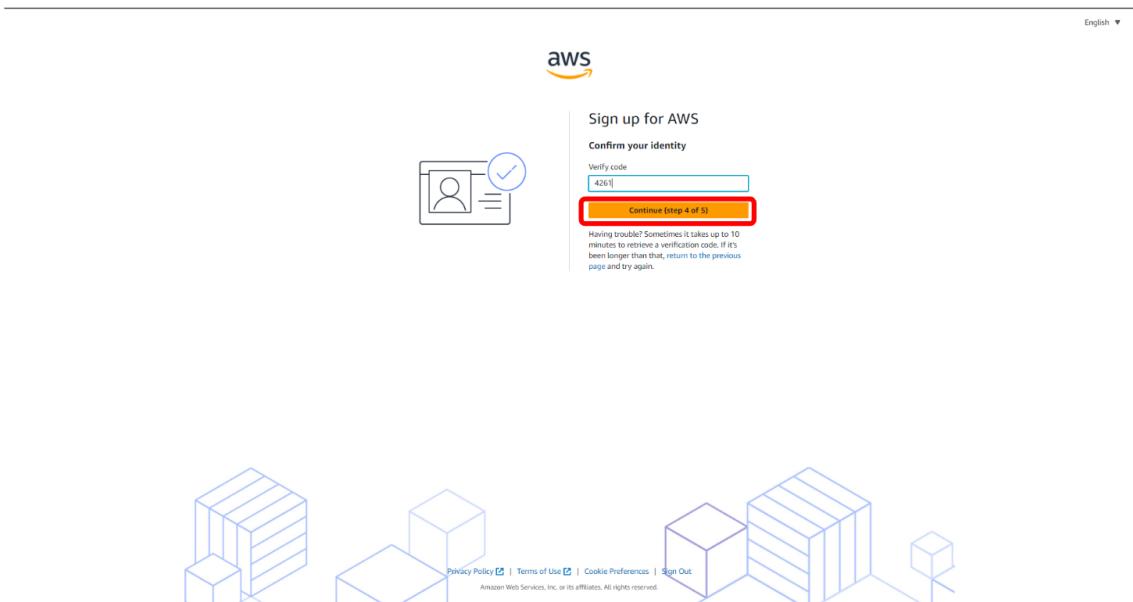
Privacy Policy [\[link\]](#) | Terms of Use [\[link\]](#) | Cookie Preferences [\[link\]](#) | Sign Out

Amazon Web Services, Inc. or its affiliates. All rights reserved.

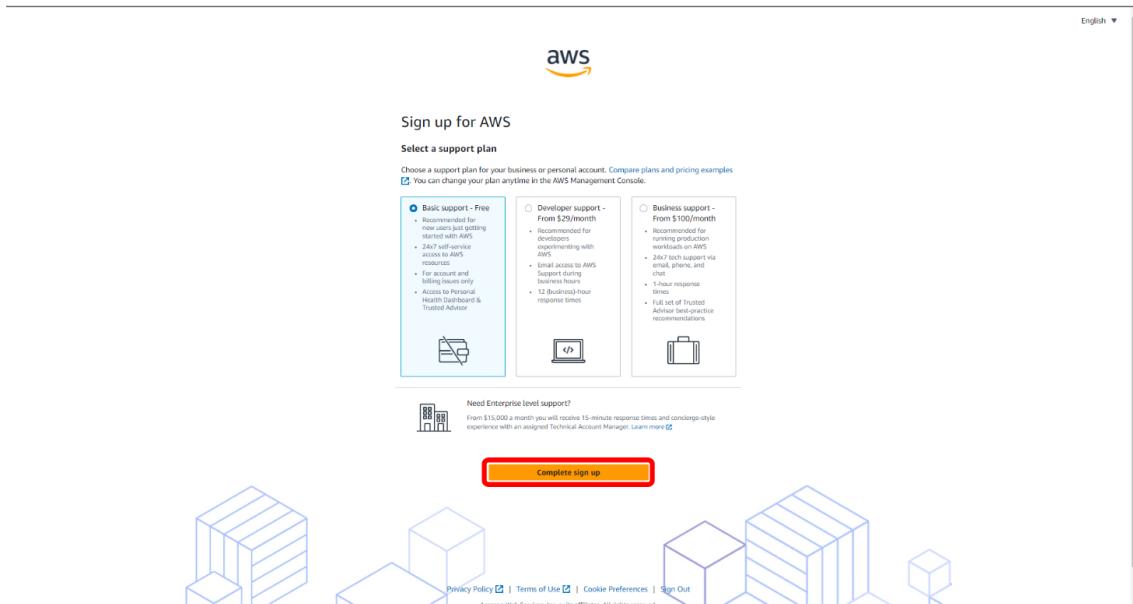
8. Enter your country or region code from the list, and then enter a phone number where you can be reached in the next few minutes. Enter the CAPTCHA code, and **Send SMS**.



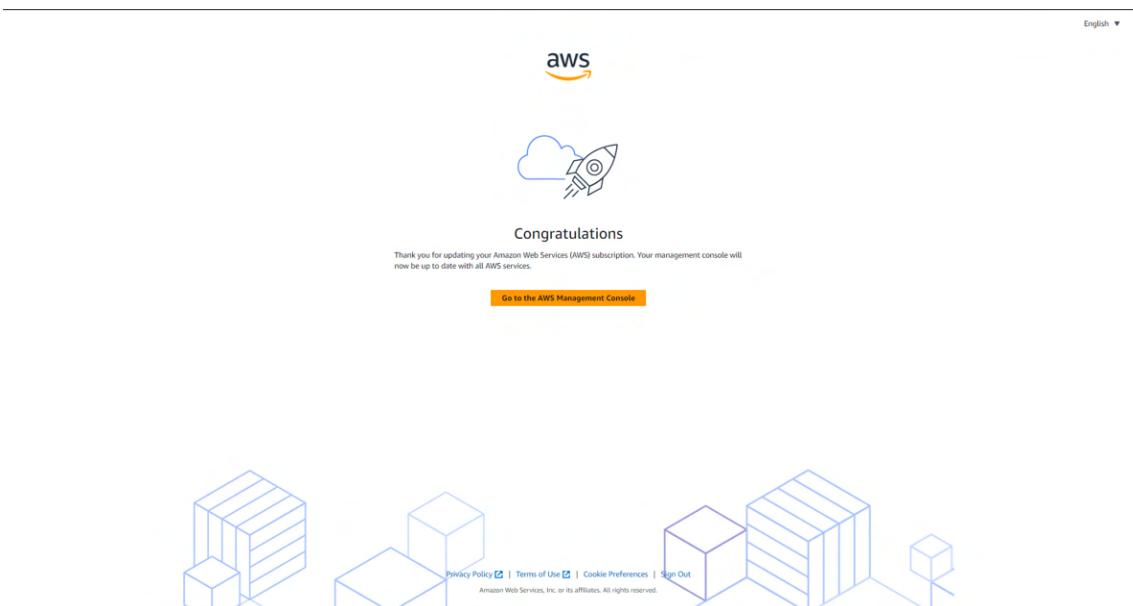
9. When the automated system contacts you, enter the PIN you receive and then submit.



10. Select your AWS Support plan. For a description of the available plans, see [Compare AWS Support plans](#).
Choose **Complete sign up**.



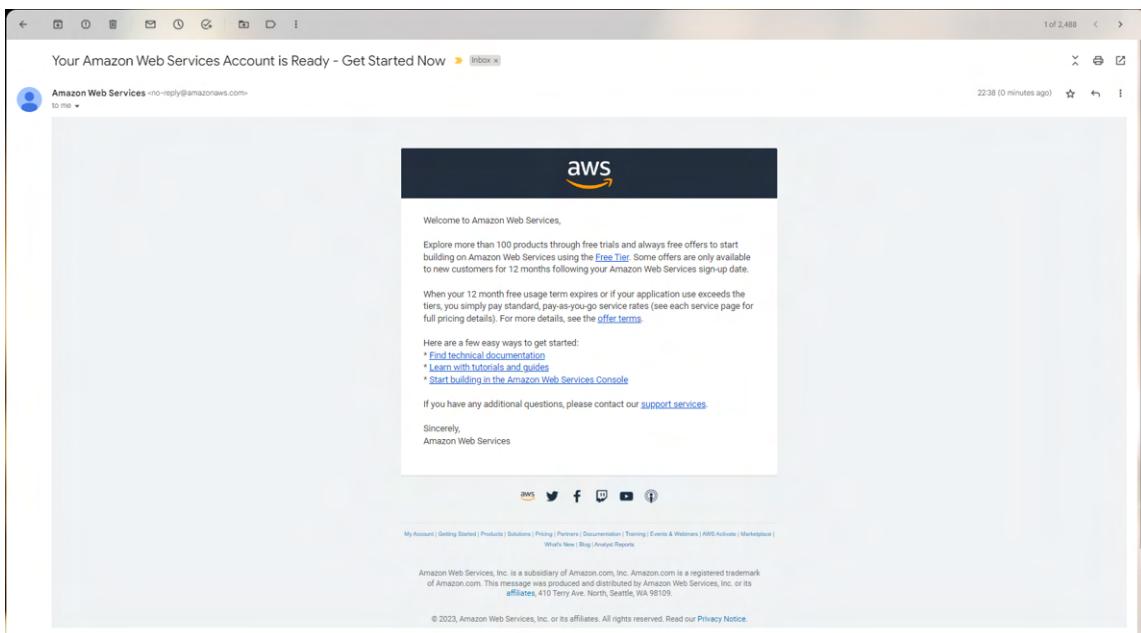
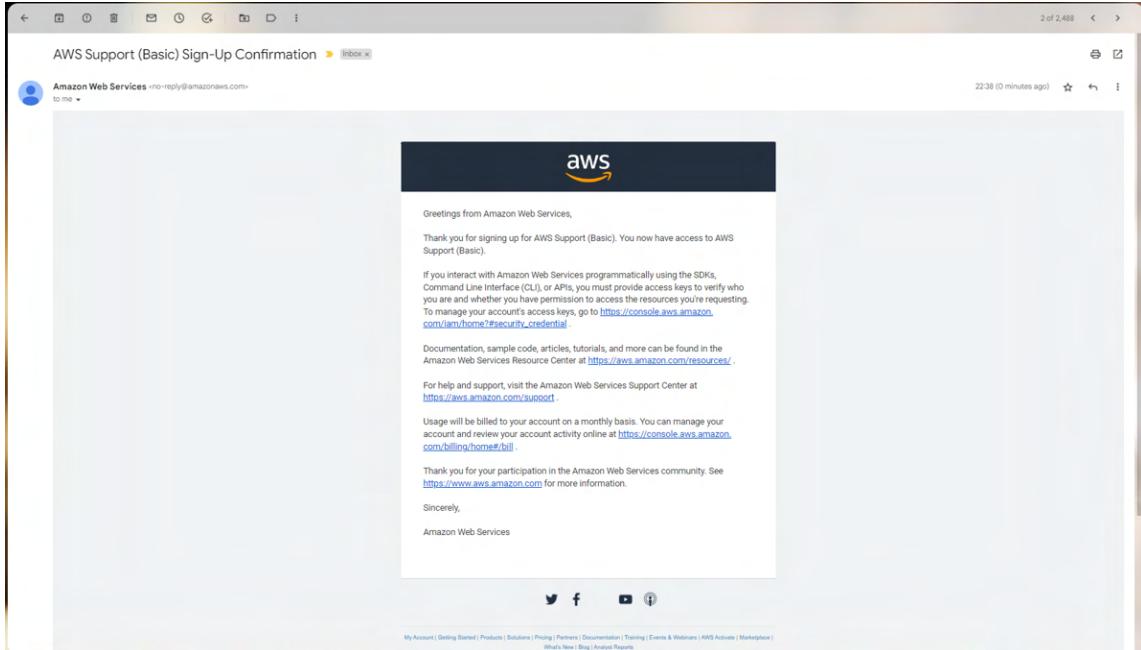
11. A confirmation page appears that indicates that your account is being activated.



12. Check your email and spam folder for an email message that confirms your account was activated. Activation usually takes a few minutes but can

sometimes take up to 24 hours.

After you receive the activation message, you have full access to all AWS services.



13. If you are having trouble with account activation, see [Troubleshooting issues with AWS account creation.](#)

▼ Activate MFA for Root User

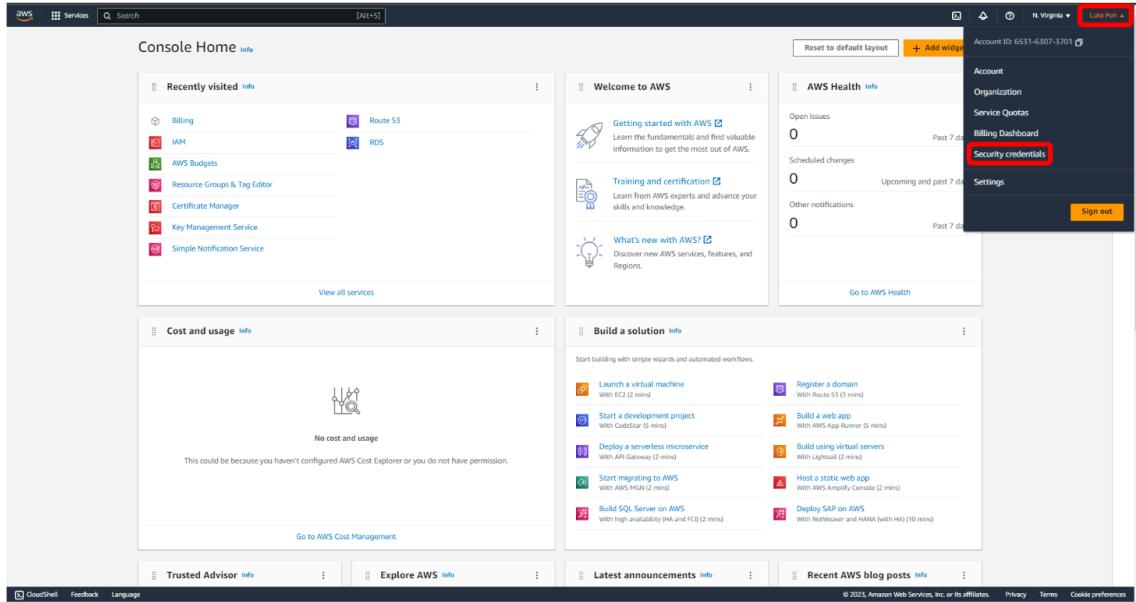


You can use the AWS Management Console to configure and enable a virtual MFA device for your root user. To enable MFA devices for the AWS account, you must be signed in to AWS using your root user credentials.

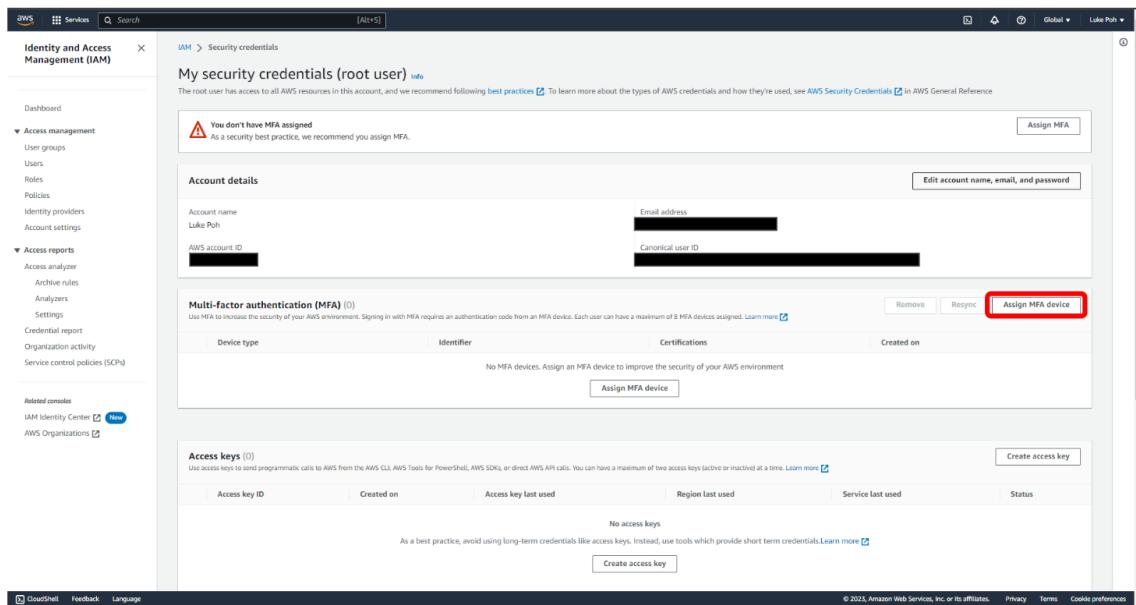


Before you enable MFA for your root user, review your account settings and contact information to make sure that you have access to the email and phone number. If your MFA device is lost, stolen, or not working, you can still sign in as the root user by verifying your identity using that email and phone number. To learn about signing in using these alternative factors of authentication, see [What if an MFA device is lost or stops working?](#)

1. Sign in to the [AWS Management Console](#) as the account owner by choosing Root user and entering your AWS account email address. On the next page, enter your password.
For help signing in using your root user, see [Sign in to the AWS Management Console as the root user](#) in the AWS Sign-In User Guide.
2. On the right side of the navigation bar, choose your account name, and choose **Security credentials**. If necessary, choose **Continue to Security** credentials.



3. In the Multi-Factor Authentication (MFA) section, choose Assign MFA device.



4. In the wizard, type a Device name, choose Authenticator app, and then choose Next.

For this workshop, specify the Device name to be “**Twilio_Authy**”, choose **Authenticator app**.

Then, click **Next**.

IAM > Security credentials > Assign MFA device

Step 1
Select MFA device

Step 2
Set up device

Select MFA device

Specify MFA device name

Device name
Enter a meaningful name to identify this device.
Twilio_Authy
Maximum 128 characters. Use alphanumeric and '+ = , . @ - ' characters.

Select MFA device Info

Select an MFA device to use, in addition to your username and password, whenever you need to authenticate.

Authenticator app
Authenticate using a code generated by an app installed on your mobile device or computer.

Security Key
Authenticate using a code generated by touching a YubiKey or other supported FIDO security key.

Hardware TOTP token
Authenticate using a code displayed on a hardware Time-based one-time password (TOTP) token.

Cancel **Next**

5. Go to the **App Store / Google Play** and download “**Twilio Authy**”



6. In “**Twilio Authy**”, fill in our **country code ‘+65’**, and fill up your **phone number**.

Add your **preferred email** too as a backup contact too.

Proceed to **verify your account** through one of the 3 methods provided.

7. Click on the **Red Plus Button / Add Account** on Twilio Authy

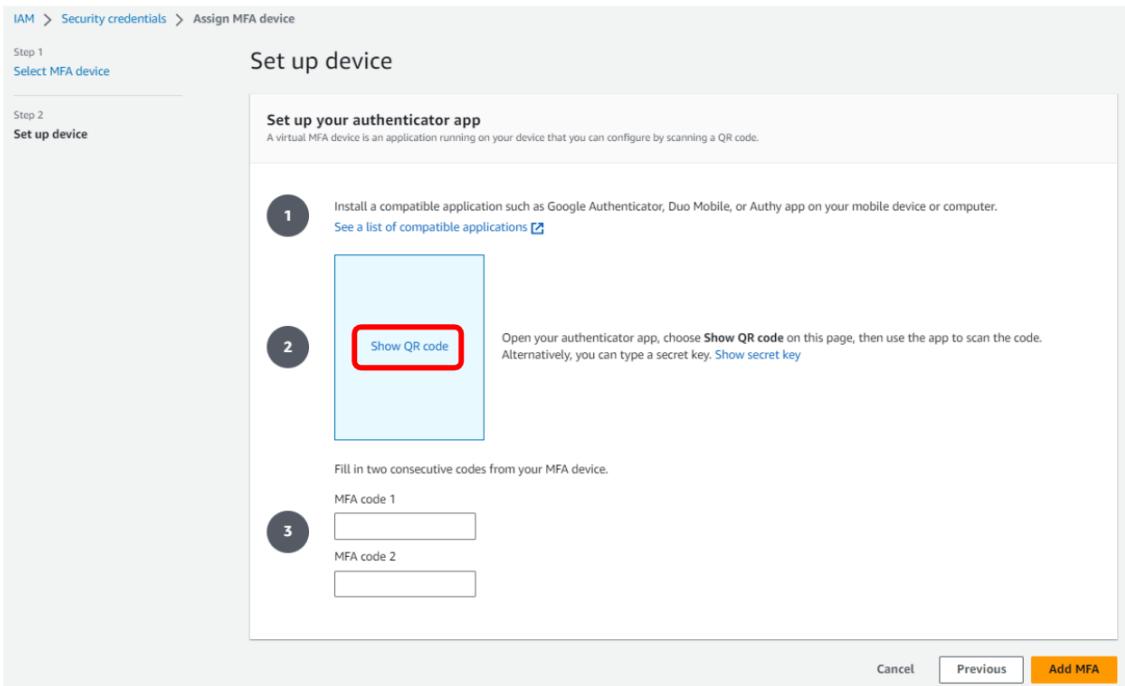
Get your camera ready to scan the QR Code on AWS.

You don't have any accounts yet.

Tap on the plus button below to Add your first authenticator account.



8. Click **Show QR code**



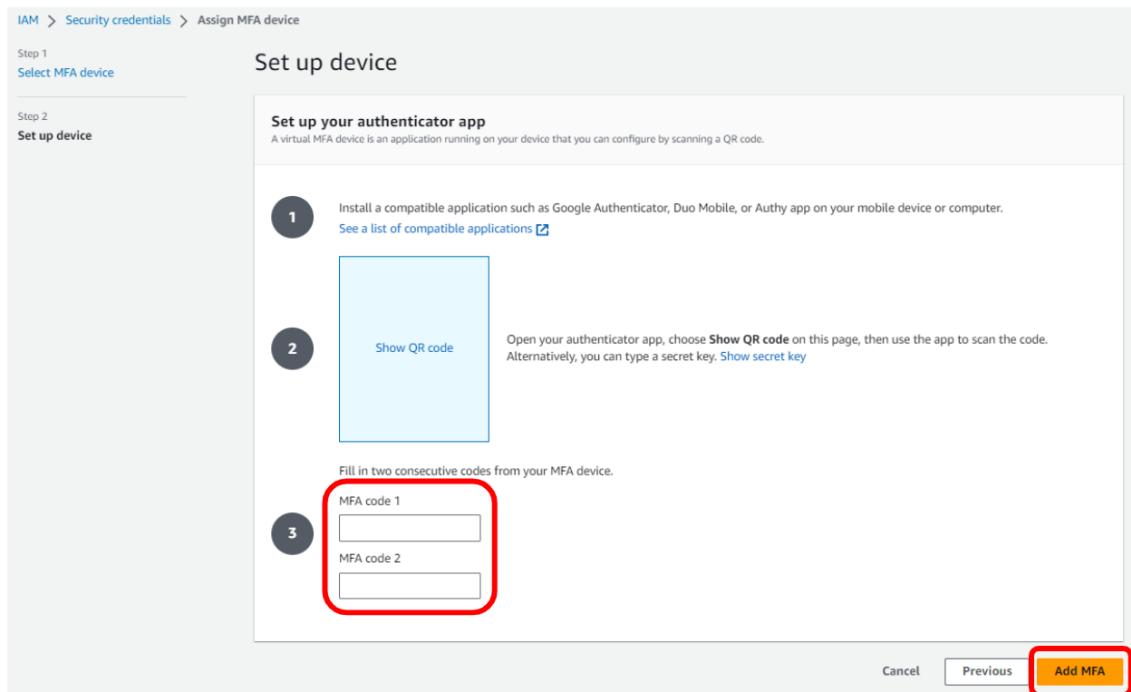
9. Scan the QR code using Twilio Authy

Proceed to save the **Account Nickname** (Customize as preferred)

10. Fill in two consecutive codes as seen in your Twilio Authy app.

In the wizard, in the **MFA code 1** box, type the one-time password that currently appears in the virtual MFA device. Wait up to 30 seconds for the device to generate a new one-time password. Then type the second one-time password into the **MFA code 2** box. Choose **Add MFA**.

Afterward, **Add MFA**



11. Afterward, it should be added under **Multi-Factor Authentication (MFA)** section.

Multi-factor authentication (MFA) (1)			
Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more [?]			
Device type	Identifier	Certifications	Created on
<input checked="" type="radio"/> Virtual	[REDACTED]mfa/Twilio_Authy	Not Applicable	Now

Visual Studio Code Set Up



Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages and runtimes (such as C++, C#, Java, Python, PHP, Go, .NET).

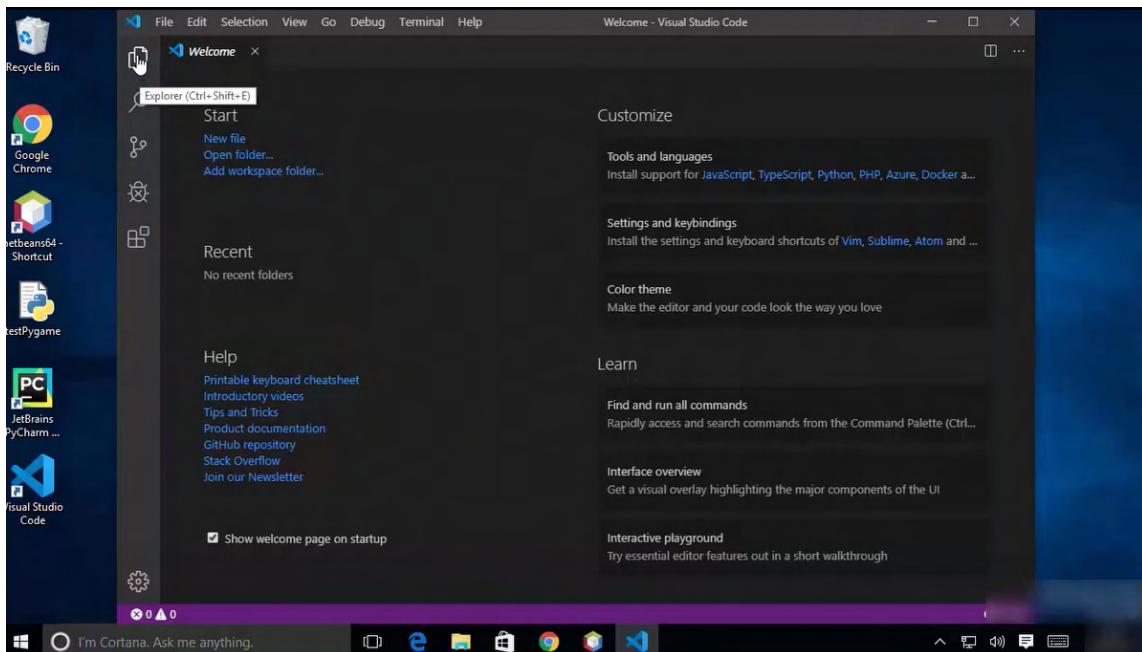
Here are some introductory videos for your own exploration →
<https://code.visualstudio.com/docs/getstarted/introvideos>

▼ Basic Download & Setup



Getting started basic document, for your own exploration →
<https://code.visualstudio.com/learn/get-started/basics>

1. Download VS Code from → <https://code.visualstudio.com/download>
2. Once it is downloaded, run the installer (VSCodeUserSetup-{version}.exe). This will only take a minute.
3. By default, VS Code is installed under `C:\Users\{Username}\AppData\Local\Programs\Microsoft VS Code`.
4. Launch VS Code. It should look something like this.



▼ Installation of Extensions

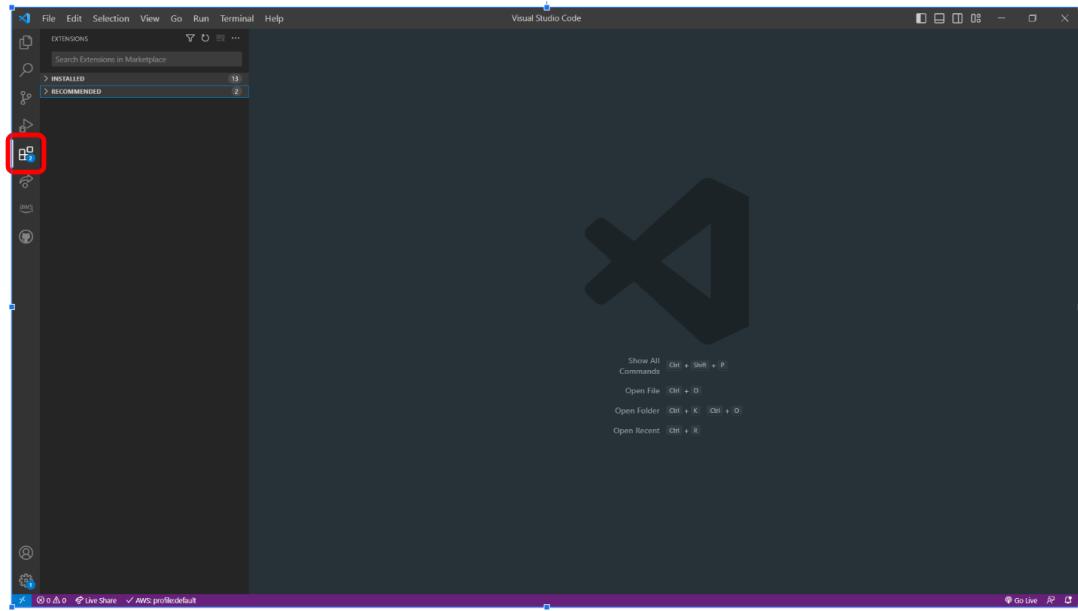


Note that because we are using our own VS Code, it might look different from yours. DO NOT PANIC.

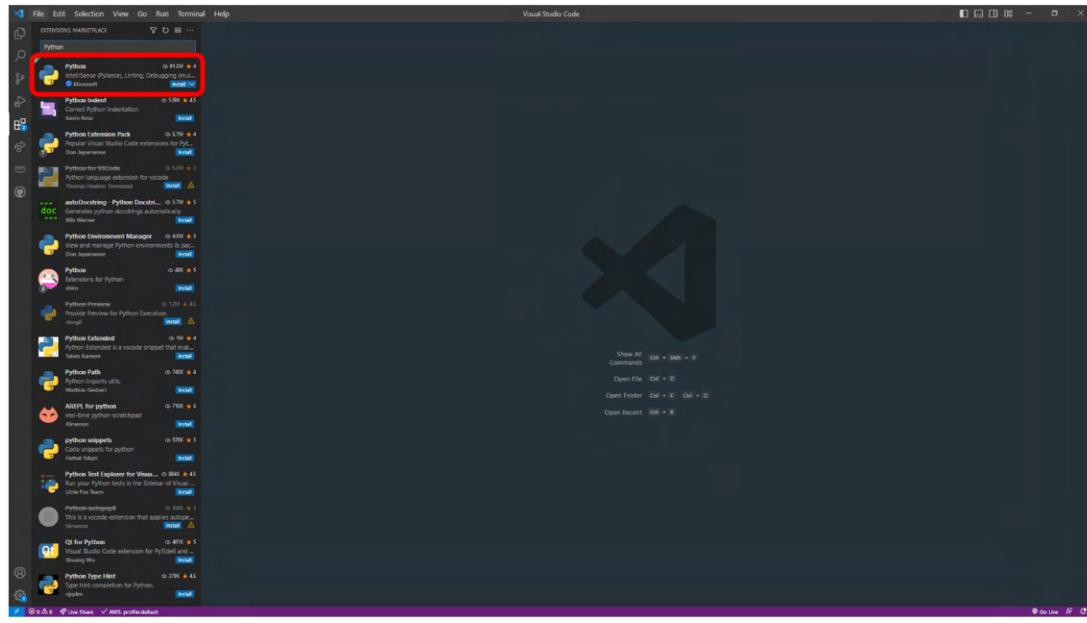
 If you already have the extensions downloaded, check that they are the correct ones installed and that they are updated.

▼ Python Extension

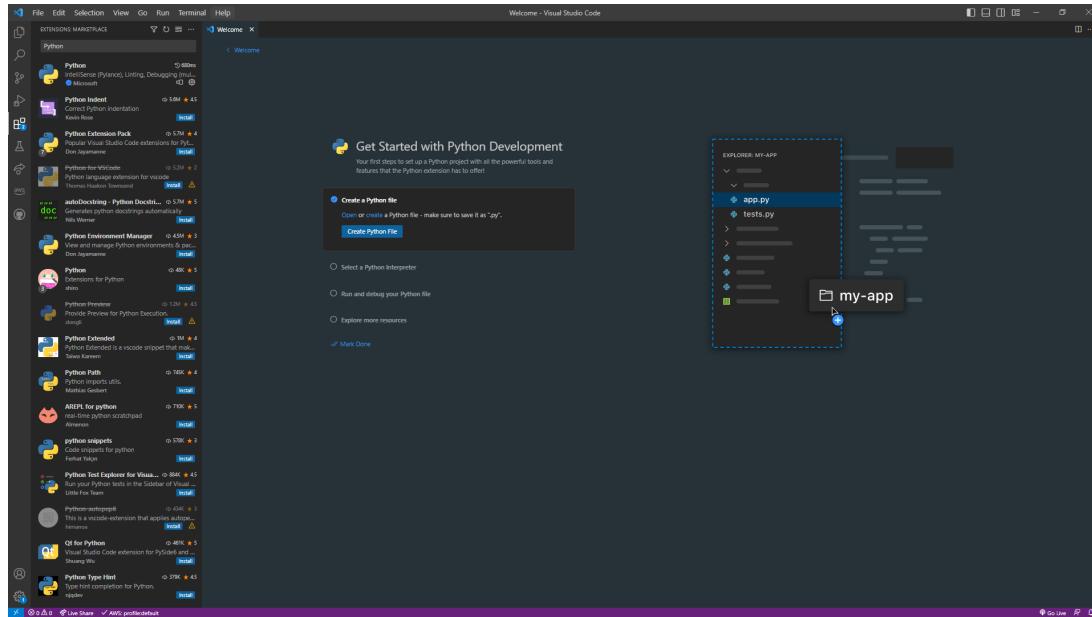
1. Open the **Extensions** activity. Alternatively, Ctrl+Shift+X to open it.



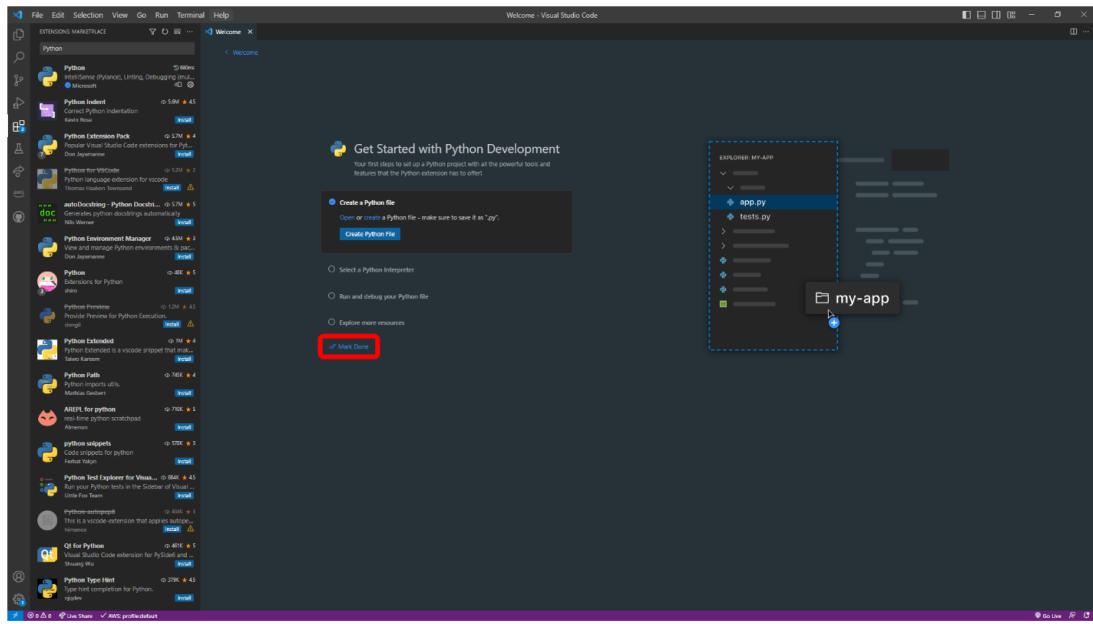
2. Search 'Python' in the search bar and download the **Python** extension.
Proceed to **Install**.



3. Wait for it to finish installing. This should take about a minute. This welcome tab should appear after.

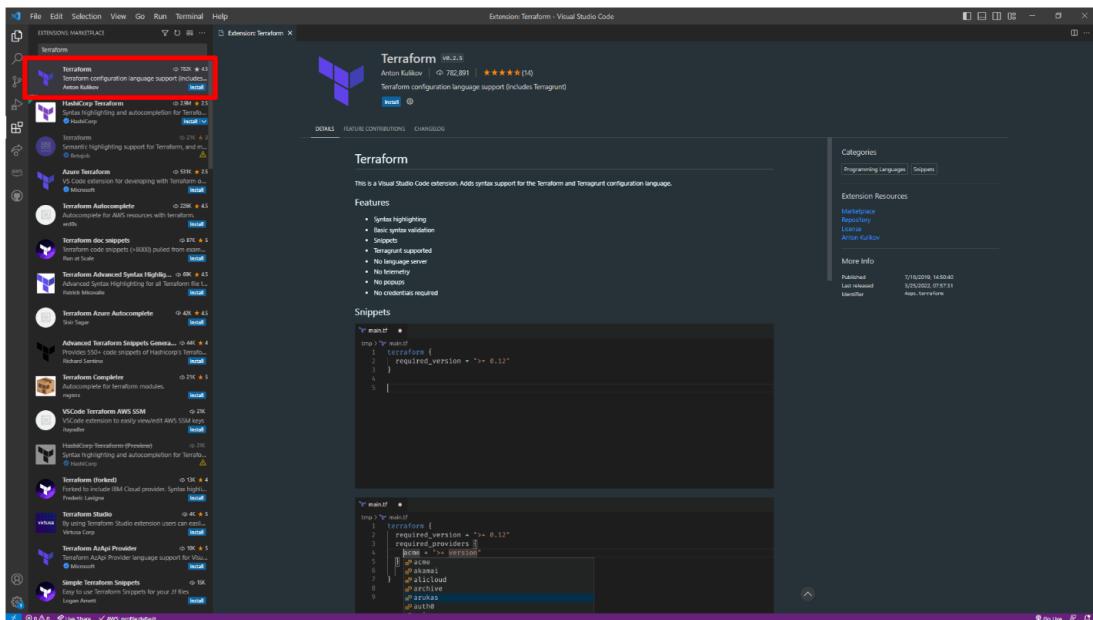


4. Proceed to **Mark Done** (Skip through the quick guide if you wish)

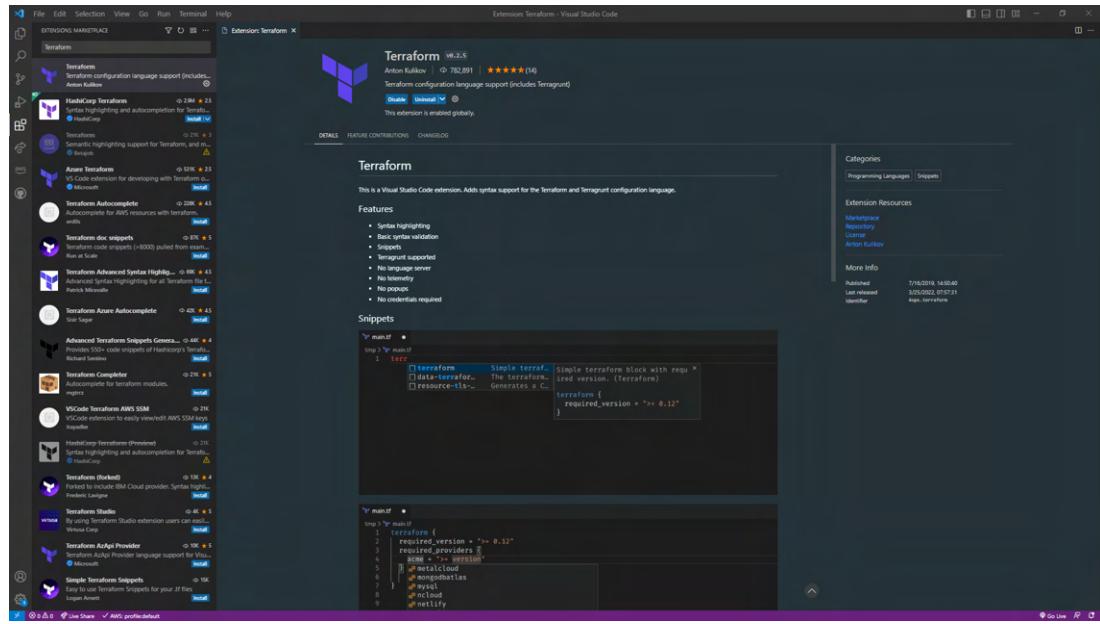


▼ Terraform Extension

1. Open the **Extensions** activity. Alternatively, **Ctrl+Shift+X** to open it.
2. Search ‘Terraform’ in the search bar and download the **Terraform** extension. Proceed to **Install**.



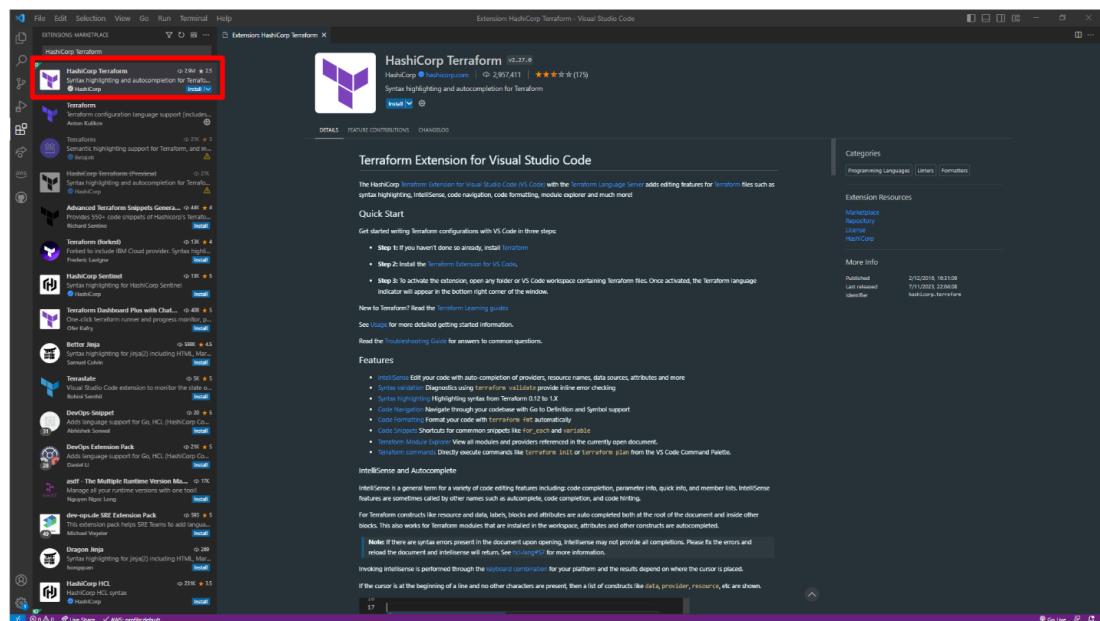
3. It should be installed instantly.



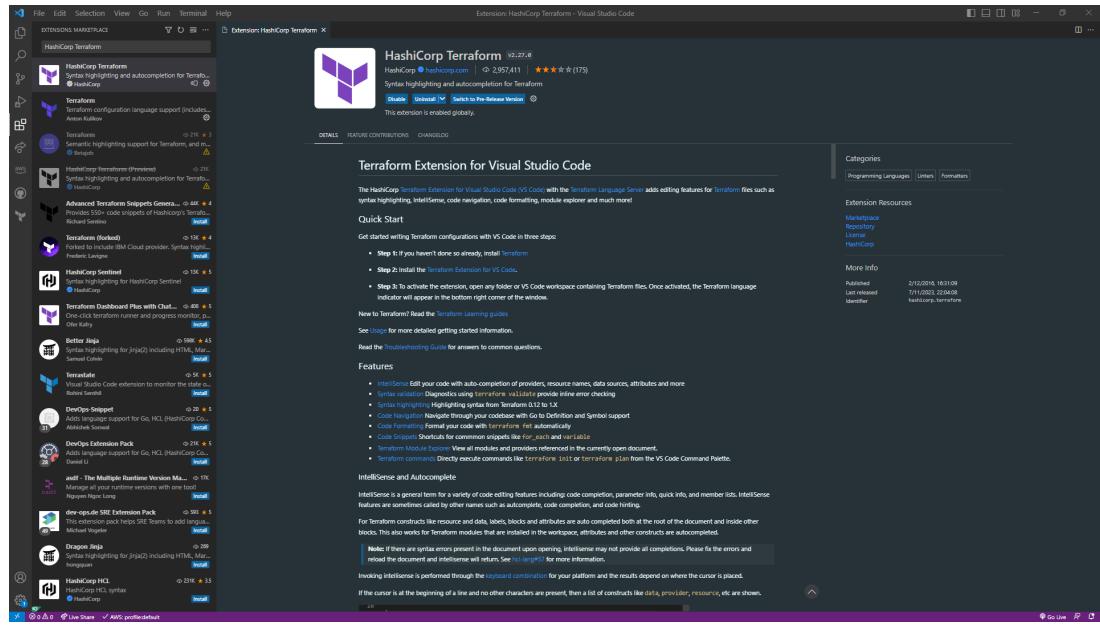
▼ HashiCorp Terraform Extension

1. Open the **Extensions** activity. Alternatively, **Ctrl+Shift+X** to open it.
2. Search ‘HashiCorp Terraform’ in the search bar and download the **HashiCorp Terraform** extension.

Proceed to **Install**.



3. It should be installed instantly.



▼ Terraform User Creation

- From your AWS Home Page, search 'IAM' in the search bar, and select the service of IAM.

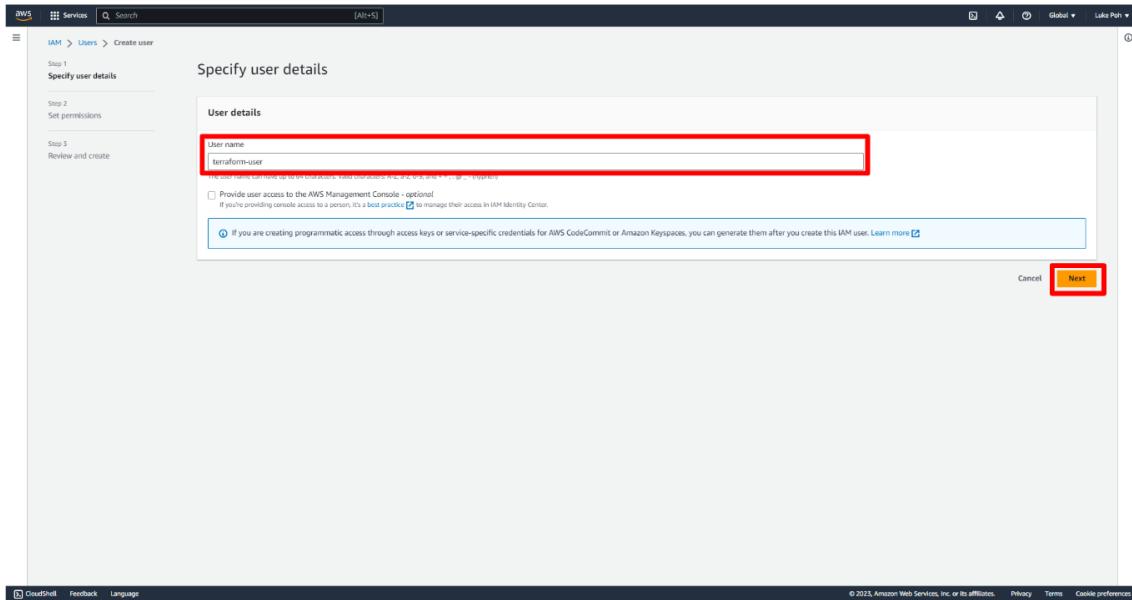
The screenshot shows the AWS Home Page. In the top search bar, 'IAM' is typed and highlighted with a red box. Below the search bar, the 'Services' section lists several services, with 'IAM' being the first item and highlighted by a red box. To the right of the search results, the 'Welcome to AWS' dashboard is visible, featuring sections like 'Getting started with AWS', 'Training and certification', 'What's new with AWS?', and 'Build a solution' with various AWS service links.

- Select Users from the side bar.

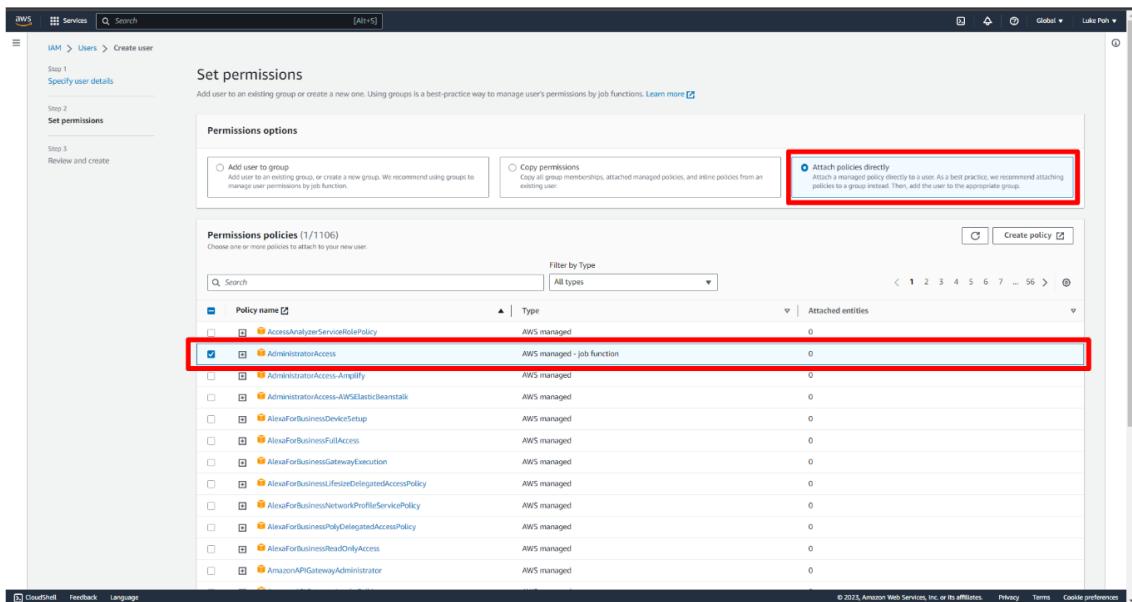
3. Select Add users.

4. Enter 'terraform-user' as the User name.

Then, select **Next**.



5. Select the **Attach policies directly** under Permissions options section. Then, tick the box of the Policy name of **AdministratorAccess** under Permission policies section.



6. Scroll down, and then select **Next**.

The screenshot shows the AWS IAM Policies search results. The 'AdministratorAccess' policy is selected and highlighted with a blue border. The table lists various AWS managed policies, each with a checkbox, name, type, and attached entities count. A red box highlights the 'Next' button at the bottom right of the search results.

7. Select **Create user**.

The screenshot shows the 'Review and create' step of the IAM user creation wizard. It displays user details (username: terraform-user, console password type: None, require password reset: No), a permissions summary (AdministratorAccess policy assigned), and a tags section (no tags). A red box highlights the 'Create user' button at the bottom right.

8. Upon returning back to this screen, select **terraform-user**.

The screenshot shows the AWS Identity and Access Management (IAM) service interface. On the left, there's a navigation sidebar with various options like Dashboard, Access management, and Access reports. The main area is titled 'Users (1) info' and shows a single user entry for 'terraform-user'. The user has an ARN (arn:aws:iam::653163073701:user/terraform-user), is assigned to a group ('Console access Disabled'), and has two access keys ('Access key 1 Not enabled' and 'Access key 2 Not enabled'). A red box highlights the 'terraform-user' entry in the list.

9. Select Security credentials.

This screenshot shows the detailed view for the 'terraform-user' account. The top navigation bar includes 'IAM > Users > terraform-user'. Below it, the 'Summary' section provides basic information like ARN, creation date, and access keys. The 'Security credentials' tab is highlighted with a red box. Under this tab, there's a table showing attached policies: 'AdministratorAccess' (AWS managed - job-function). The 'Permissions' tab is also visible. Other sections include 'Permissions policies (1)', 'Permissions boundary (not set)', and 'Generate policy based on CloudTrail events'.

10. Under the section of Access keys, select Create access key.

The screenshot shows the AWS IAM Security Credentials page. The left sidebar includes sections for Dashboard, Access management, Users, Roles, Identity providers, Account settings, Access reports, and Related consoles. The main content area is titled 'Summary' and shows the ARN as 'arn:aws:iam::653163073701:user/terraform-user'. It indicates 'Console access Disabled' and lists two access keys: 'Access key 1' (Not enabled) and 'Access key 2' (Not enabled). Below this is the 'Console sign-in' section with a link to the AWS sign-in console. The 'Multi-factor authentication (MFA)' section shows 'No MFA devices' assigned. The 'Access keys' section is highlighted with a red box around the 'Create access key' button.

11. Select **Third-party service**.

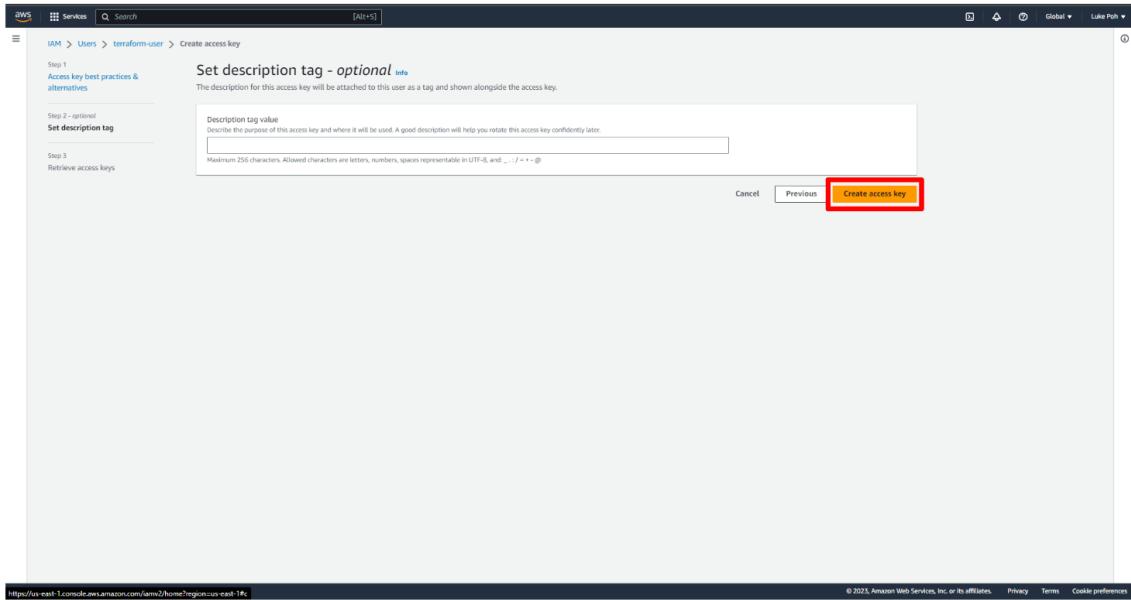
Then, tick the **Confirmation** that reads 'I understand the above recommendation and want to proceed to create an access key.'

Lastly, select **Next**.

The screenshot shows the 'Access key best practices & alternatives' step of the 'Create access key' wizard. It lists several use cases: Command Line Interface (CLI), Local code, Application running on an AWS compute service, Third-party service (which is selected and highlighted with a red box), Application running outside AWS, and Other. Below these is an 'Alternative recommended' section with a note about using temporary security credentials instead of long-term access keys. At the bottom, a checkbox for 'I understand the above recommendation and want to proceed to create an access key' is checked, and a red box highlights the 'Next' button.

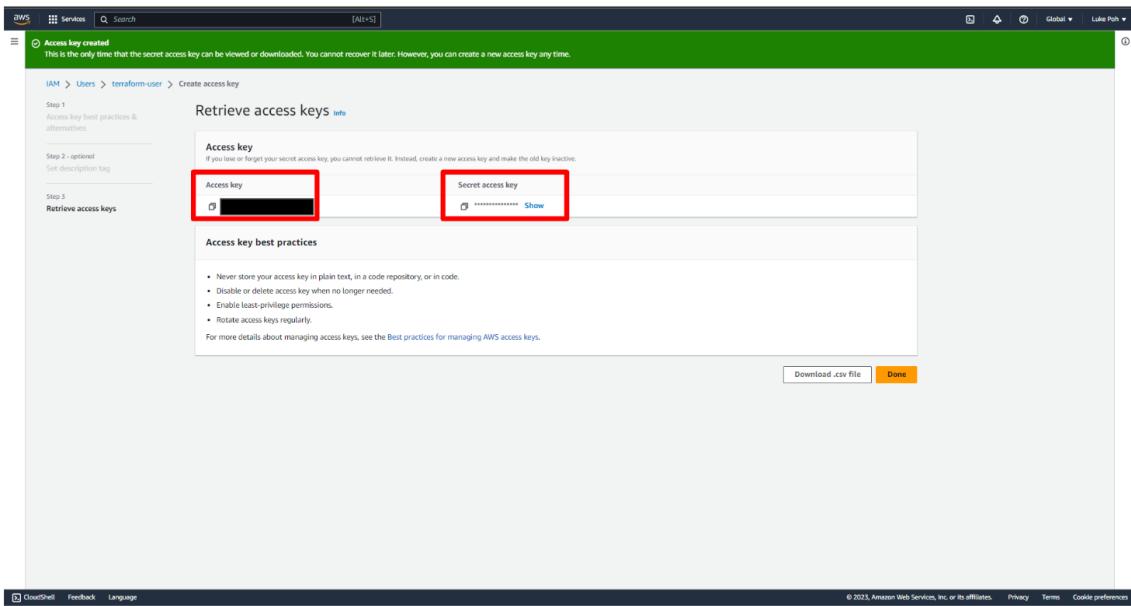
121. No need to enter a Description tag value.

Simply select **Create access key**.



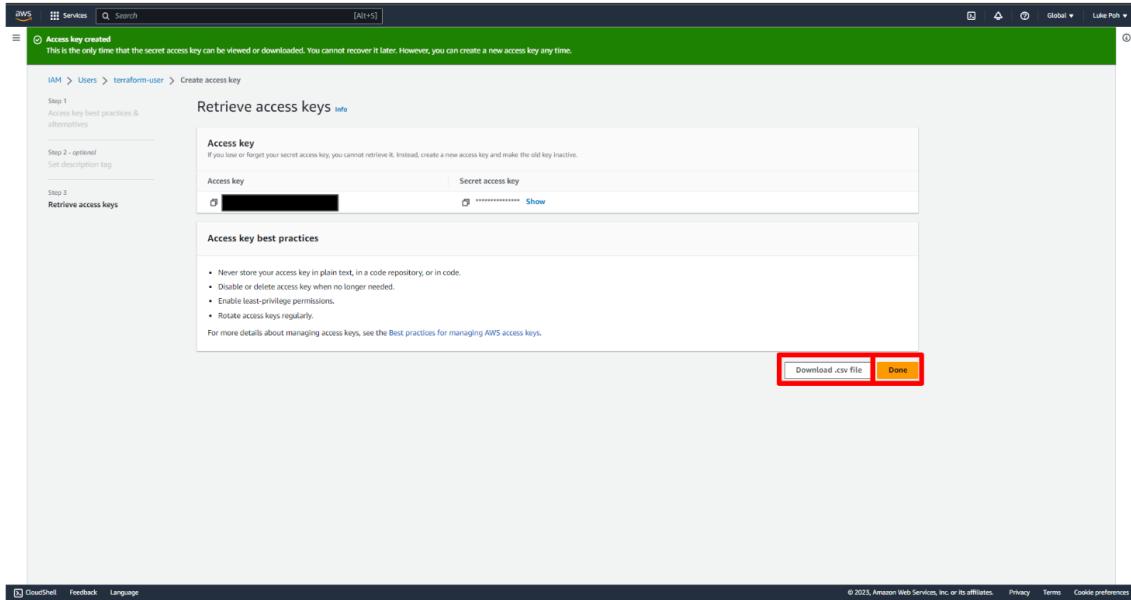
122. Upon entering this page, you would have to **save your access keys**.

**SAVE BOTH KEYS. ACCESS KEY & SECRET ACCESS KEY
PLEASE COPY AND STORE THE ACCESS KEYS SOMEWHERE. YOU
NEED THESE.**

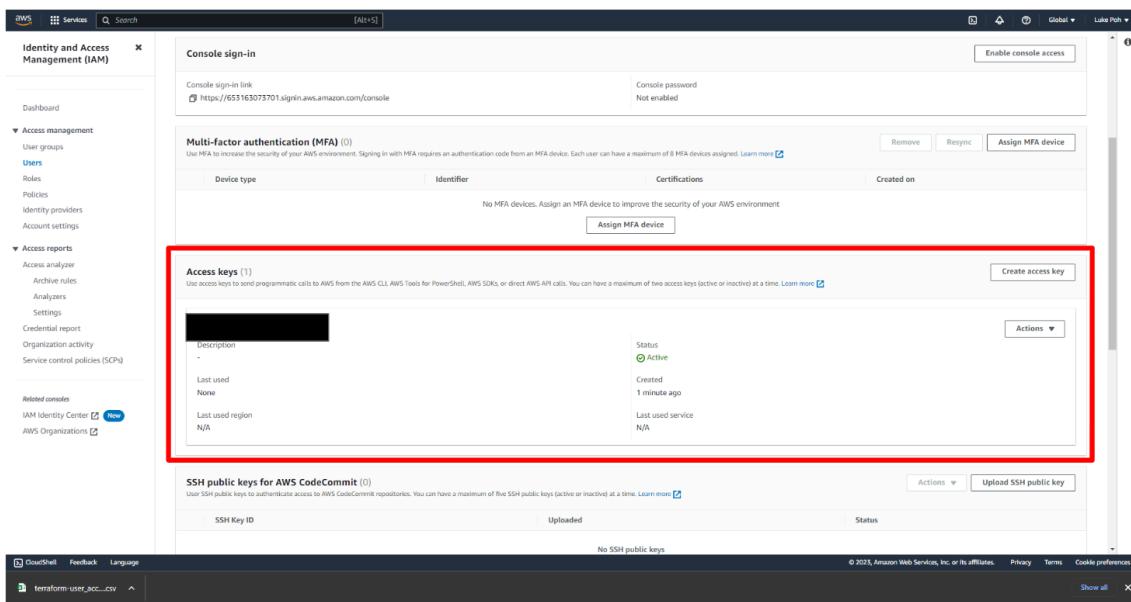


123. Alternatively, **Download .csv file**. The .csv file stores both access keys.
Do this in addition to Step 13 to ensure that you have the access keys.

Check that you can open the .csv file after downloading it.
After which, Select **Done**.



124. Lastly, you should end up at this screen, whereby you can now see the access key created.



Terraform Set Up



Terraform is an open-source Infrastructure as Code (IaC) tool developed by HashiCorp. It enables users to define and provision infrastructure resources across various cloud providers, including Amazon Web Services (AWS), in a declarative manner. Terraform uses a domain-specific language (DSL) called HashiCorp Configuration Language (HCL) to describe the desired state of the infrastructure.



We will be solely utilising Terraform to develop the 2 projects in the workshop. Please ensure that it is installed.

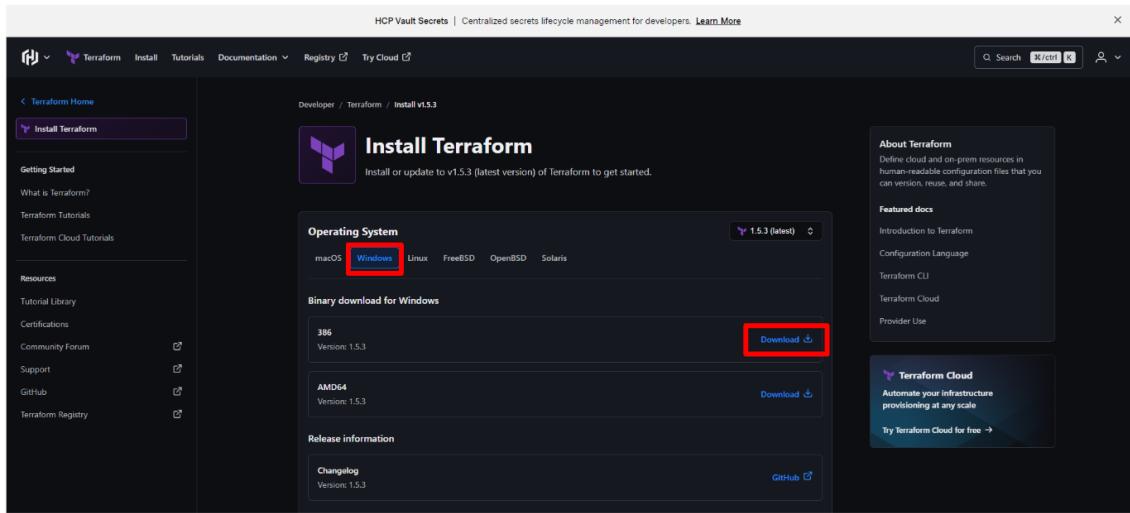
▼ Basic Download & Setup



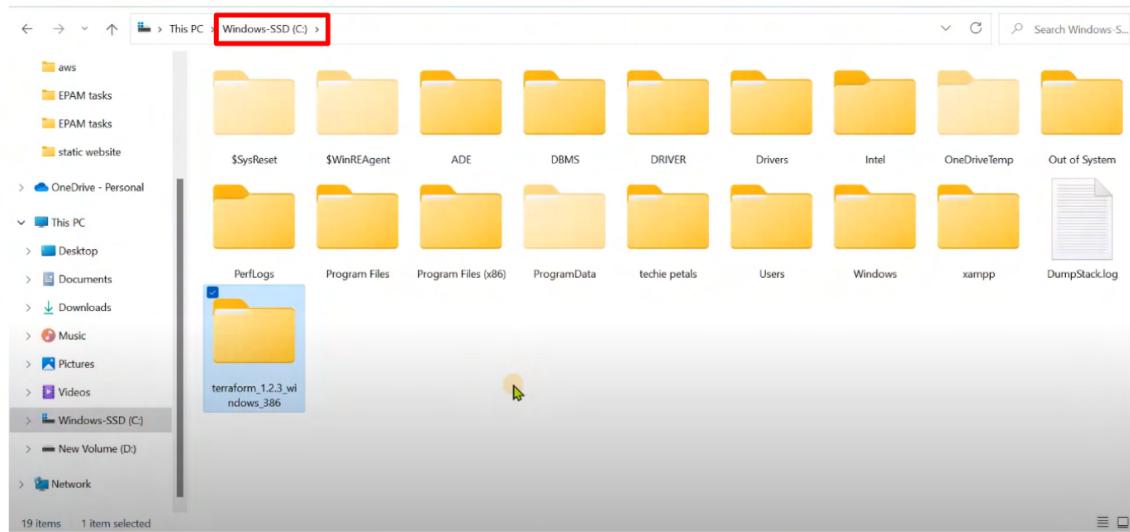
Here is the full Terraform Installation guide on YouTube →

[https://www.youtube.com/watch?
v=bMRBVYNhS5M&ab_channel=TechiePetals](https://www.youtube.com/watch?v=bMRBVYNhS5M&ab_channel=TechiePetals)

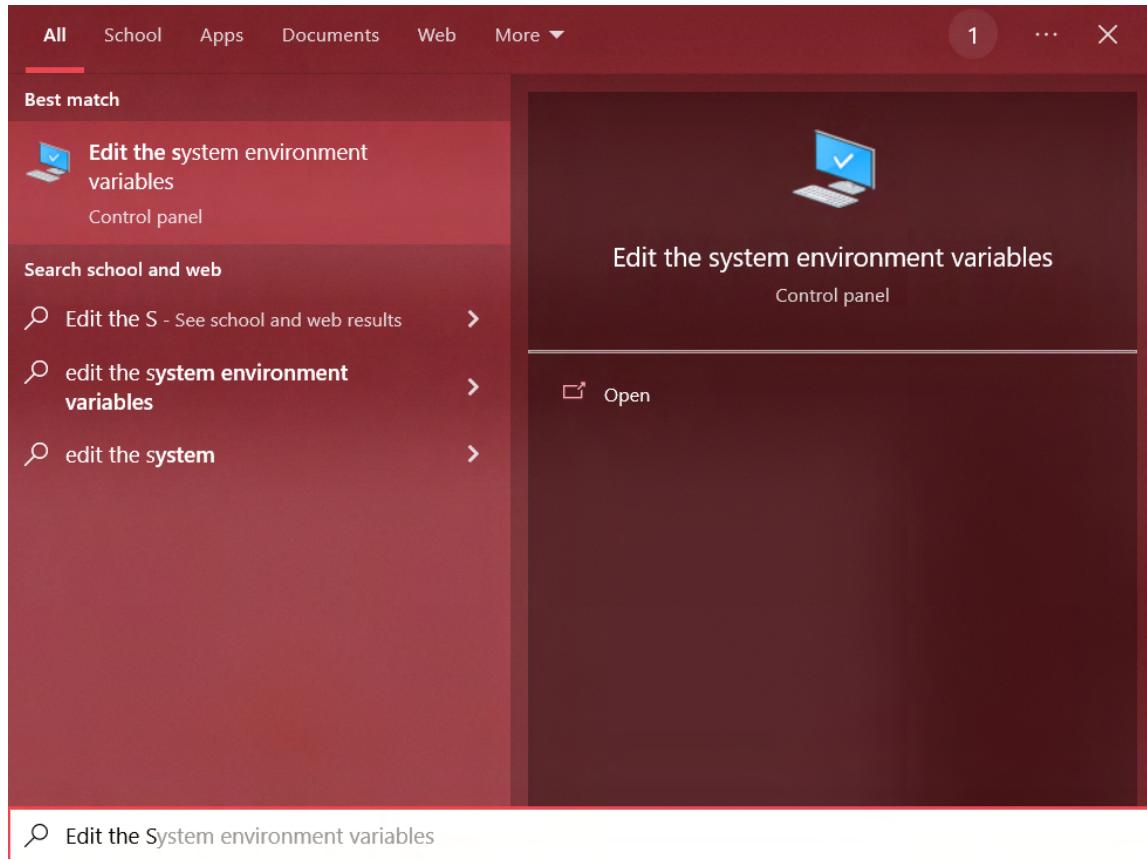
1. Head to the Terraform Download page →
https://developer.hashicorp.com/terraform/downloads?product_intent=terraform
2. Select the **Windows** category on the navigation bar and then download the 386 binary file → Select **Download**



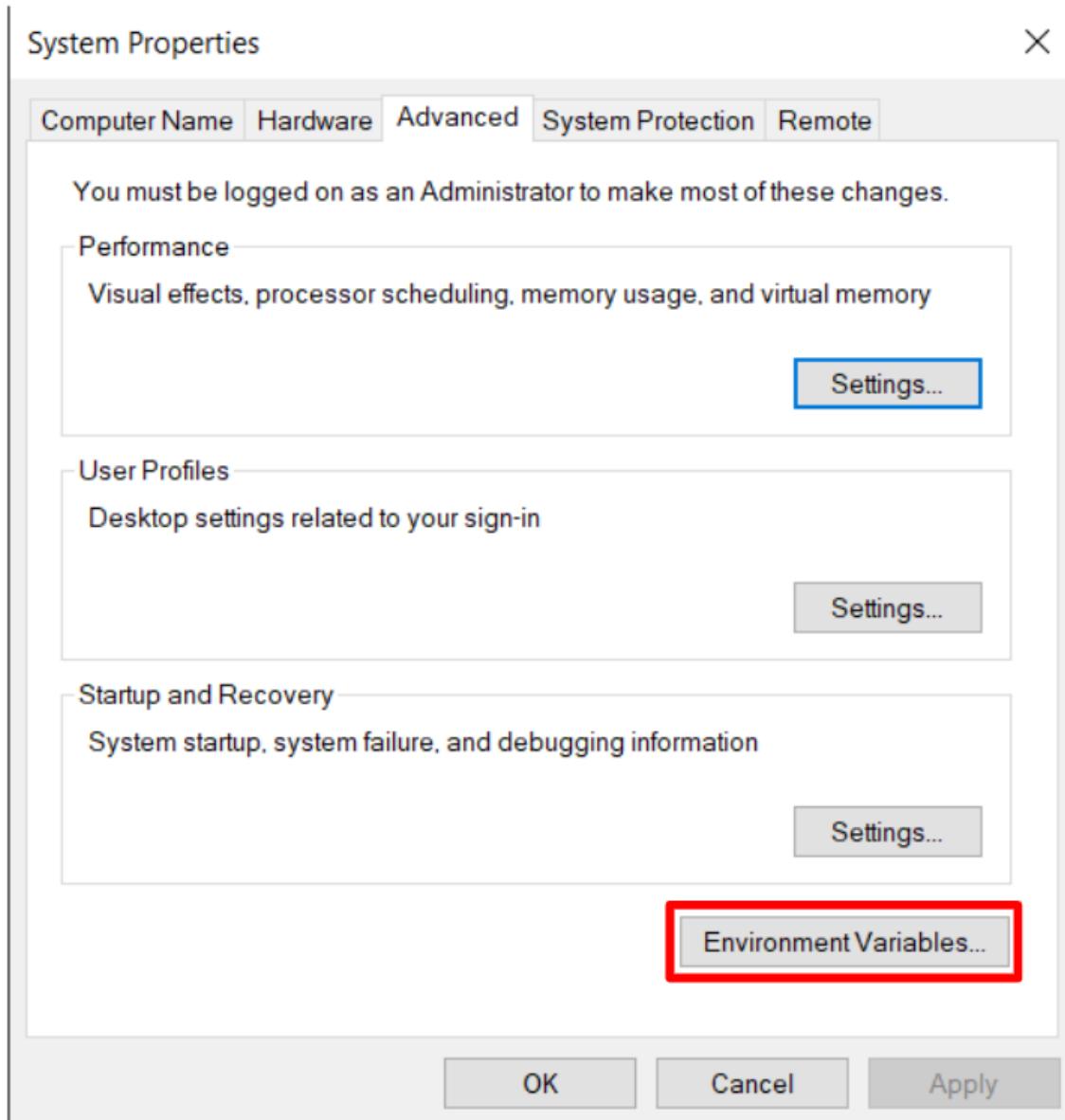
- Extract the zip file and place the contents in any directory you want. In this example, the zip was extracted to the C: Drive



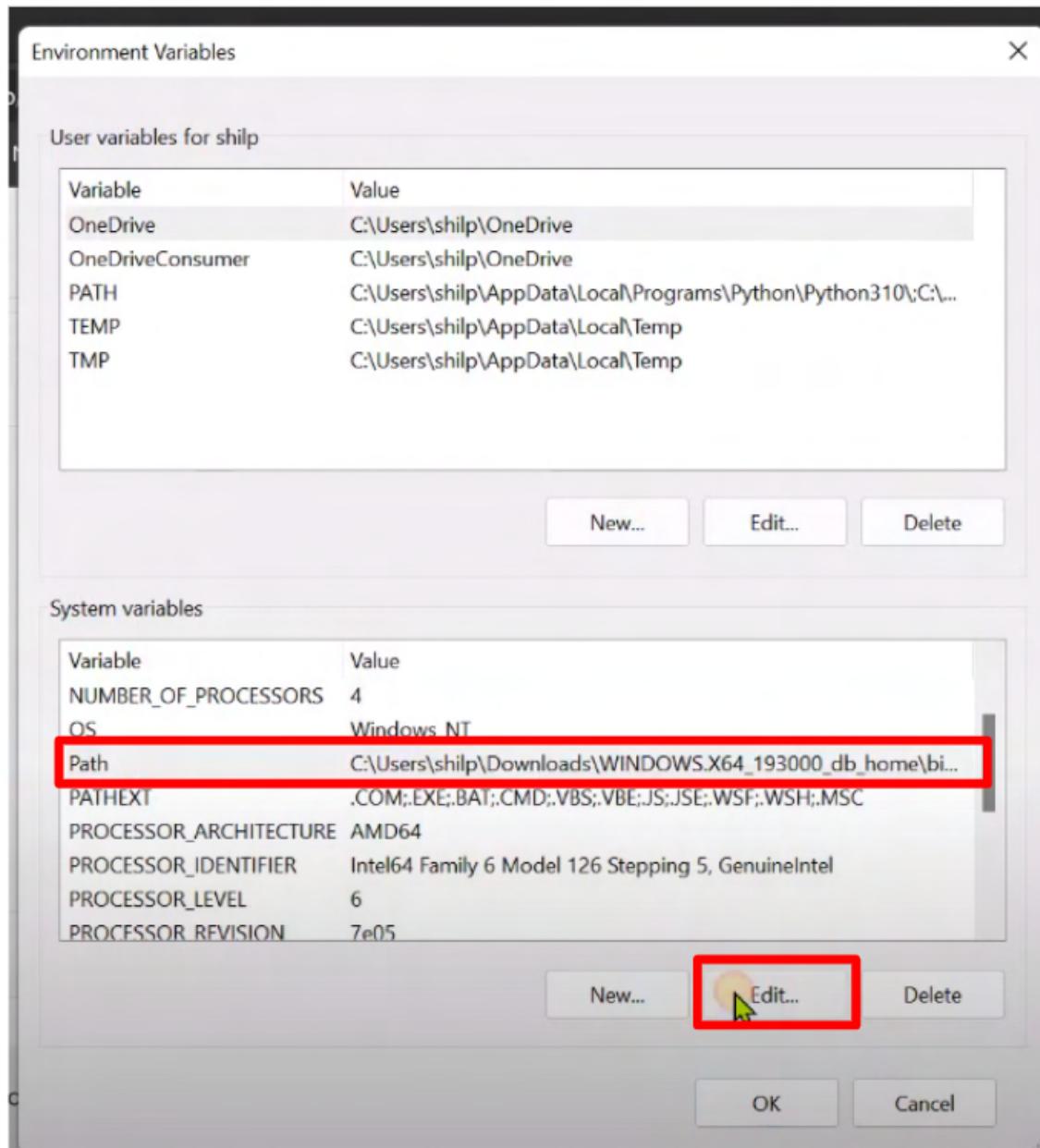
- Remember the file path** where the terraform folder is located in (In the example above for example, the file path is "C:\terraform_1.2.3_windows_386\")
- Search **Edit the system environment variables** in the Windows search bar
(Use shortcut → **Windows Key + S** to access the Windows search bar)



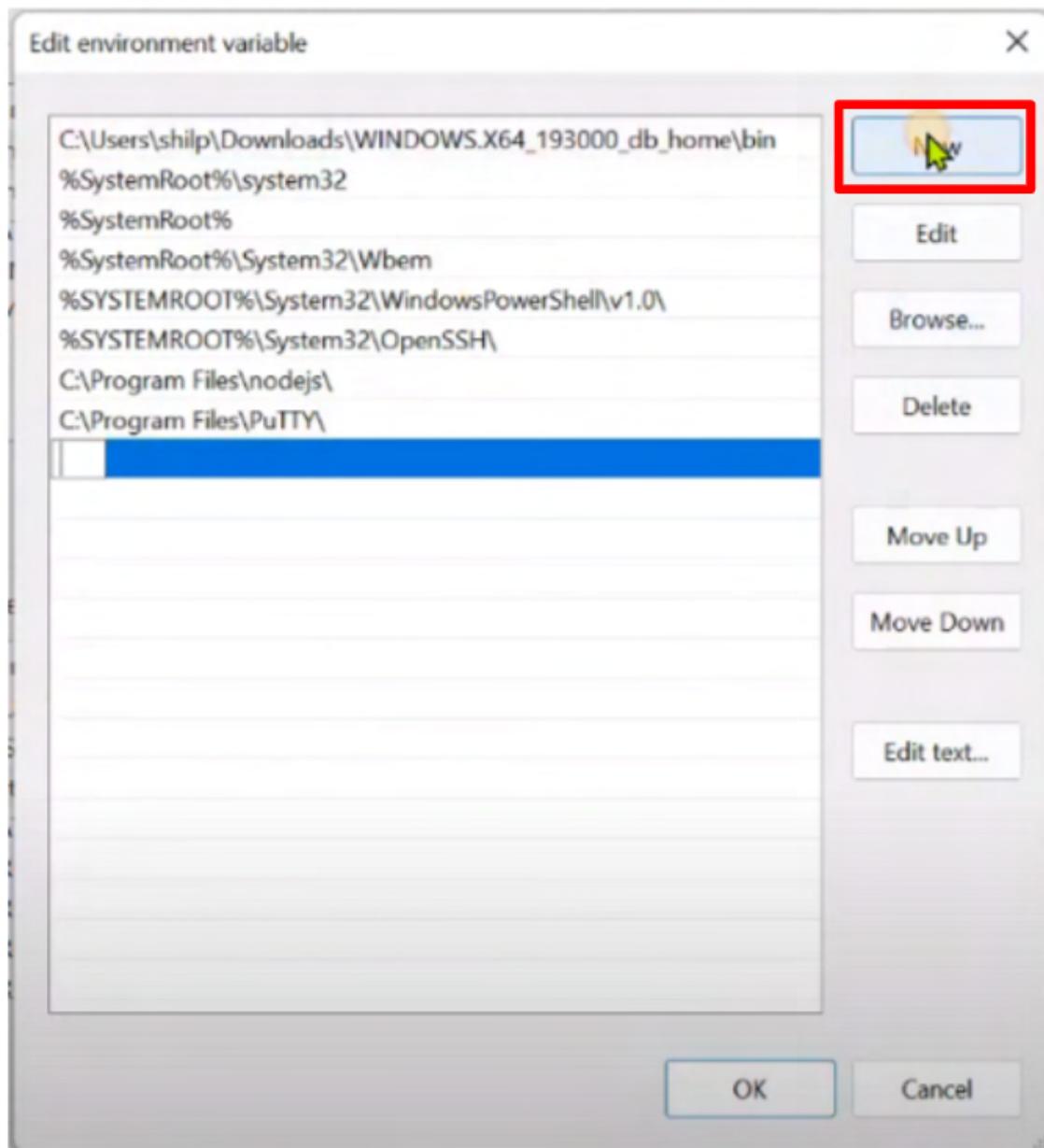
6. Select Environment Variables



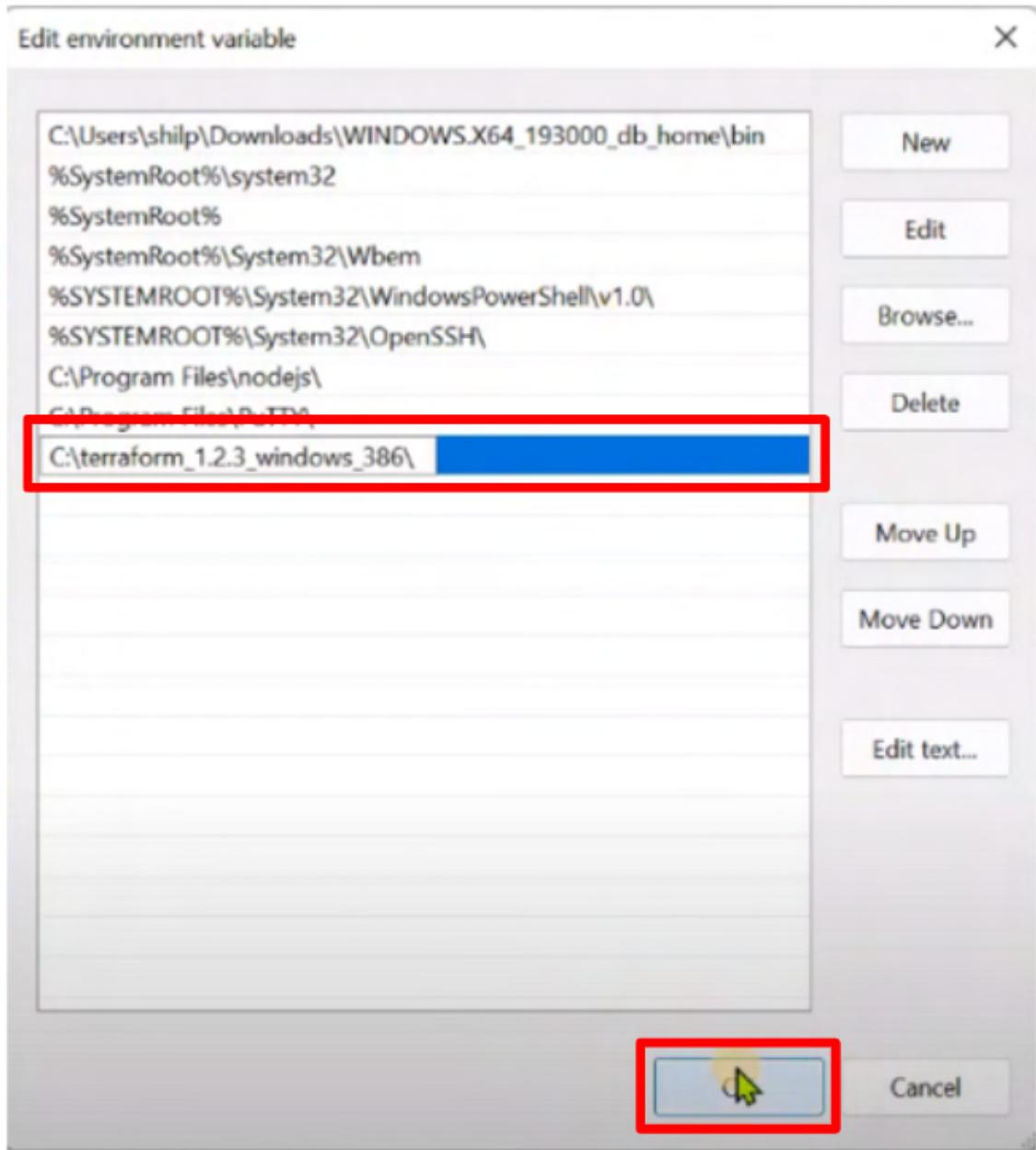
7. Under System variables, select the **Path** environment variable and **Edit**



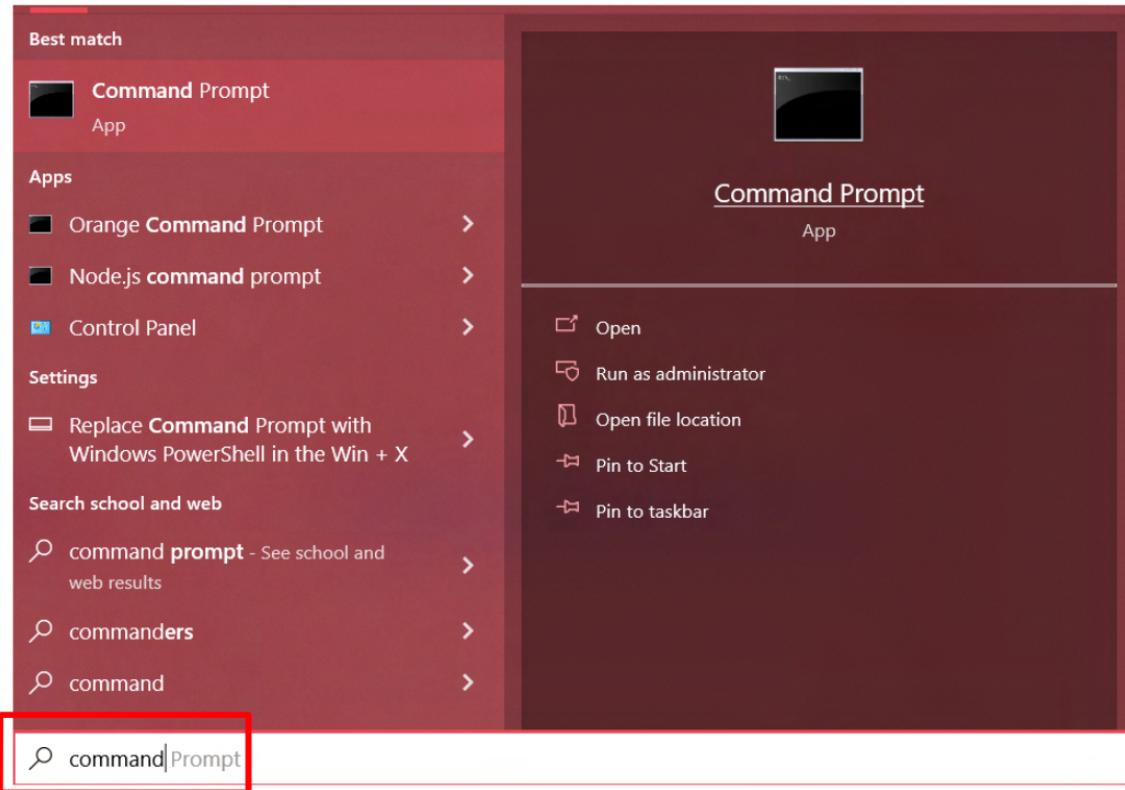
6. Select **New**



7. Then, **enter in your file path** (In this case, the file path is "C:\terraform_1.2.3_windows_386")
Then select **OK**



7. To test whether terraform is successfully installed, first search **Command Prompt** in the Windows search bar. (Use shortcut → **Windows Key + S** to access the Windows search bar)



8. Within the Command Prompt, type in 'terraform -v' and **Enter**

```
Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Luke>terraform -v
```

8. If you see the following message, then the installation is successful!

```
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nigel Lim>terraform -v
Terraform v1.4.5
on windows_386
```