

Tools to Interact with AWS Services

EC006 - AWS Cloud Services and Infrastructure

Activity

Tools to Interact with AWS Services

Activity Output

```
ubuntu@ip-172-31-16-215: ~  
  
Expanded Security Maintenance for Applications is not enabled.  
  
0 updates can be applied immediately.  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu@ip-172-31-16-215:~$
```

Activity: Instructions

- Review the topics discussed in the previous session.
- Answer the quiz form about the Introduction to AWS.

Tools to Interact with AWS Services

Tools to Interact with AWS Services

Tools to Interact with AWS Services

These are the tools that allow users to manage and interact with AWS services.

1. AWS Console
2. AWS CLI (Command Line Interface)
3. AWS CloudShell

AWS Console


AWS Console provides a **web-based graphical user interface** (GUI) for managing and monitoring AWS resources. Users can navigate through various menus and options to perform operations on AWS services, such as creating and configuring compute instances, managing storage, and setting up databases.


Tools to Interact with AWS Services


Console Home [Info](#)


Actions ▾


Recently visited [Info](#)


 Amazon Managed Blockchain


 CodeBuild


 DynamoDB


 ElastiCache


 CodePipeline


 EC2


 Resource Groups & Tag Editor

 Service Catalog


 AWS Cost Explorer


 Support


 S3

 Elastic Container Registry

Welcome to AWS [↗](#)

 **Getting started with AWS**
Learn the fundamentals and find valuable information to get the most out of AWS.

 **Training and certification**
Learn from AWS experts and advance your skills and knowledge.

 **What's new with AWS?**
Discover new AWS services, features, and Regions.

AWS Health [Info](#)

Open issues
0
Past 7 days

Scheduled changes
0
Upcoming and past 7 days

Other notifications
0
Past 7 days

[Go to AWS Health](#)

Cost and usage [Info](#)

Current month costs
\$100.07


Forecasted month end costs
\$100.28
Up 0% over last month


Last month costs
\$100.32


Costs shown are unblended. [Learn more](#) [↗](#)

[Go to AWS Cost Management](#)

Top costs for current month

 AWS Support (Business) \$100.00

 Amazon GuardDuty \$0.04

 Amazon Simple Storage Service \$0.04

© 2017-2021 Zuitt Learning Institute Incorporated and its affiliates. All Rights Reserved. Confidential

8

AWS CLI

AWS CLI provides a **command-line interface** for interacting with AWS services. Users can perform the same operations as in the AWS Console but by typing commands in the terminal or command prompt. For example, users can launch an EC2 instance, upload a file to S3, or create a Lambda function through the CLI.

Tools to Interact with AWS Services

```
Command Prompt - aws configure
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\colby>aws configure
AWS Access Key ID [None]:
AWS Secret Access Key [None]:
Default region name [None]: us-east-1
Default output format [None]: json_
```

AWS CloudShell

AWS CloudShell is a **browser-based shell environment** that provides users with a **command-line interface (CLI)** to manage and interact with their AWS resources. It is an integrated part of the AWS Console and provides users with a pre-configured environment to run the AWS CLI, SDKs, and other tools.

Tools to Interact with AWS Services

AWS CloudShell

us-east-1

Preparing your terminal...

Try these commands to get started:

aws help or aws <command> help or aws <command> --cli-auto-prompt

[cloudshell-user@ip-10-1-160-196 ~]\$ echo Thank goodness for this new AWS feature!

Thank goodness for this new AWS feature!

[cloudshell-user@ip-10-1-160-196 ~]\$ █

Tools to Interact with AWS Services

Feature	AWS CLI	AWS Console	AWS Cloudshell
Environment	Standalone tool that must be installed on a user's local computer or server	Web-based graphical user interface	Cloud-based environment that runs in the user's web browser
Access	Users must configure and manage their own access keys and credentials to access AWS services	Users can sign in with their existing AWS account and manage access to AWS services using IAM	Users can sign in with their existing AWS account and the environment is automatically pre-configured with secure access to AWS services using IAM
Persistence	Does not provide users with persistent storage for their scripts or files	Provides users with a console history and some limited session persistence	Provides users with persistent storage backed by Amazon Elastic File System (EFS)

Tools to Interact with AWS Services

Feature	AWS CLI	AWS Console	AWS Cloudshell
Resource Utilization	Utilizes local computing resources	Utilizes cloud computing resources	Utilizes cloud computing resources
Availability	Can be used offline, as long as the user has previously installed it on their local machine or server	Requires an internet connection and is accessible from any web browser	Requires an internet connection to run, and the availability of the service is dependent on AWS service availability

AWS Management Console

Tools to Interact with AWS Services

1. Go to the AWS website at <https://aws.amazon.com/>
2. Click on the "Sign in to the Console" button in the top right corner of the page.
3. Sign in with the credentials your instructor provided you.
4. Once you're signed in, you'll be taken to the AWS Management Console. From here, you can access all the different AWS services and tools.

AWS CLI

Tools to Interact with AWS Services

To set up the environment for AWS CLI, you need to follow these steps:

Windows:

1. Download and run the [AWS CLI MSI installer for Windows \(64-bit\)](https://awscli.amazonaws.com/AWSCLIV2.msi). Alternatively, you can run the `msiexec` command to run the MSI installer.

```
msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi
```

2. To confirm the installation, open the Start menu, search for `cmd` to open a command prompt window, and at the command prompt use the `aws --version` command.

```
C:\> aws --version
```

```
aws-cli/2.4.5 Python/3.8.8 Windows/10 exe/AMD64 prompt/off
```

To set up the environment for AWS CLI, you need to follow these steps:

MacOS:

1. In your browser, download the [macOS pkg file](#)
2. Run your downloaded file and follow the on-screen instructions. You can choose to install the AWS CLI in the following ways:
 - a. For all users on the computer (requires sudo)
 - You can install to any folder, or choose the recommended default folder of /usr/local/aws-cli.
 - The installer automatically creates a symlink at /usr/local/bin/aws that links to the main program in the installation folder you chose.

To set up the environment for AWS CLI, you need to follow these steps:

MacOS:

3. For only the current user (doesn't require sudo):
 - You can install to any folder to which you have write permission.
 - Due to standard user permissions, after the installer finishes, you must manually create a symlink file in your \$PATH that points to the aws and aws_completer programs by using the following commands at the command prompt. If your \$PATH includes a folder you can write to, you can run the following command without sudo if you specify that folder as the target's path. If you don't have a writable folder in your \$PATH, you must use sudo in the commands to get permissions to write to the specified target folder. The default location for a symlink is /usr/local/bin/.

```
$ sudo ln -s /<folder/installed>/aws-cli/aws /usr/local/bin/aws
```

```
$ sudo ln -s /<folder/installed>/aws-cli/aws_completer  
/usr/local/bin/aws_completer
```

3. To verify that the shell can find and run the aws command in your \$PATH, use the following commands.

```
$ which aws
```

```
/usr/local/bin/aws
```

```
$ aws --version
```

```
aws-cli/2.4.5 Python/3.8.8 Darwin/18.7.0 botocore/2.4.5
```

Configuring AWS CLI credentials

Tools to Interact with AWS Services

Configuring AWS CLI credentials

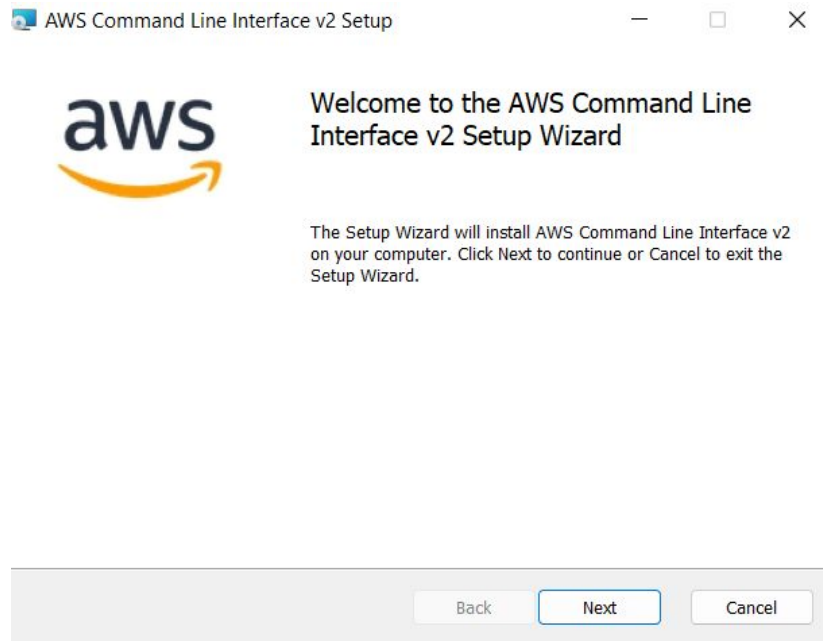
1. In the terminal, type the command `aws configure`.
2. Enter the prompted values (will be provided by your instructor):
 - a. AWS Access Key ID
 - b. AWS Secret Access Key
 - c. Default region name (optional)
 - d. Default output format (optional)

Windows Setup Guide for AWS CLI

Tools to Interact with AWS Services

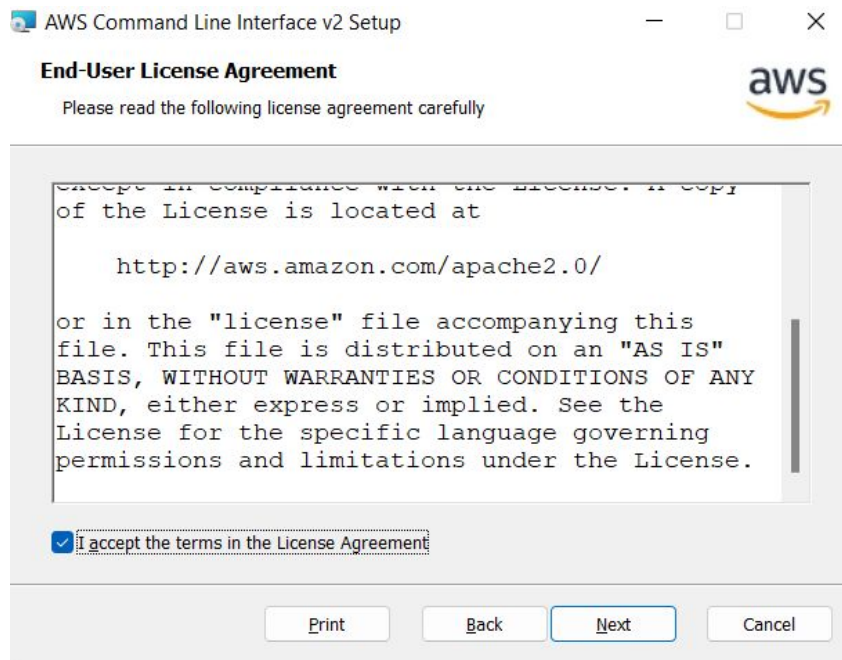
Windows Setup Guide for AWS CLI

1. Open the installer and click **Next**



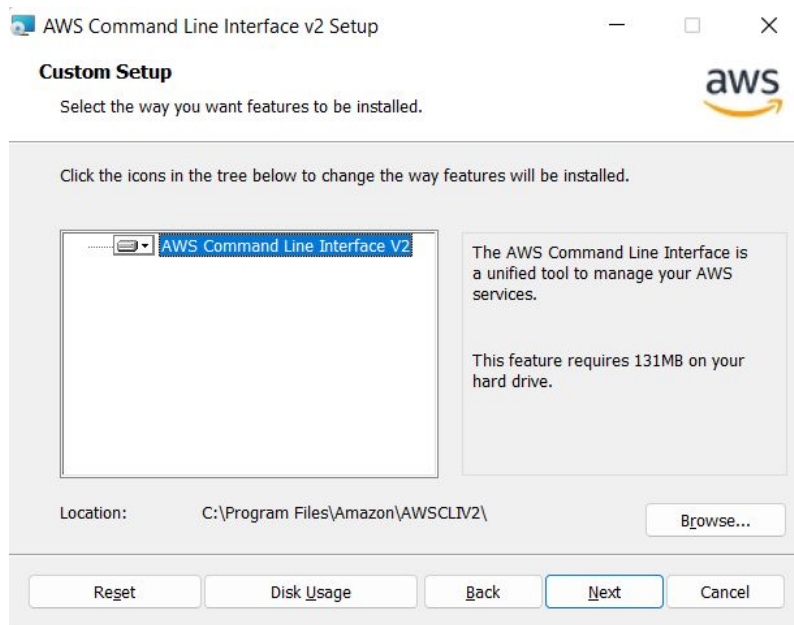
Windows Setup Guide for AWS CLI

2. **Agree** to the terms and condition and click **Next**



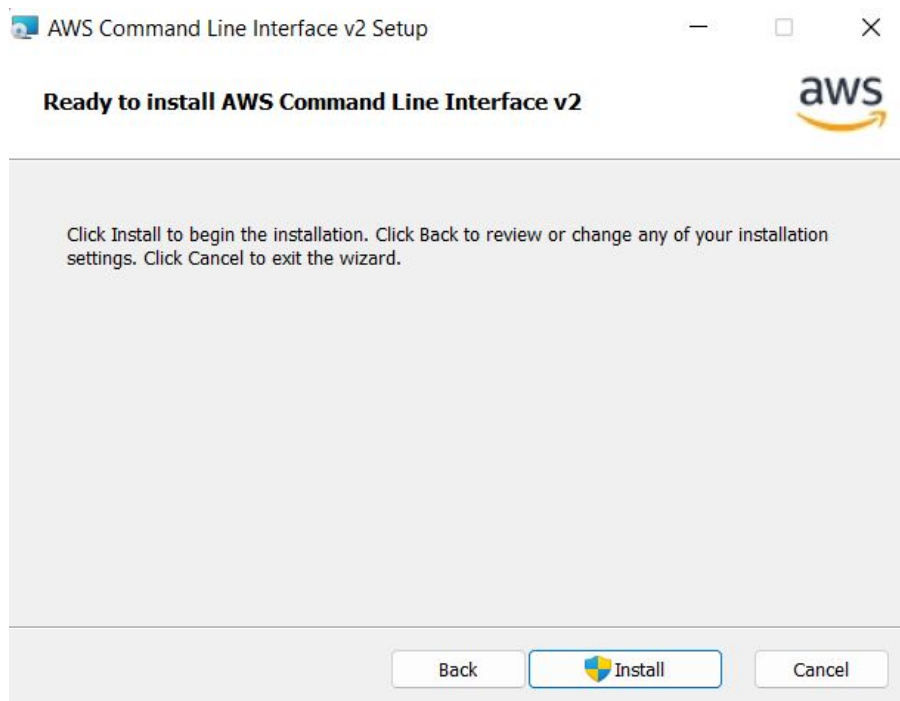
Windows Setup Guide for AWS CLI

3. Leave the default setting and click **Next**. Note of the **installation location**.



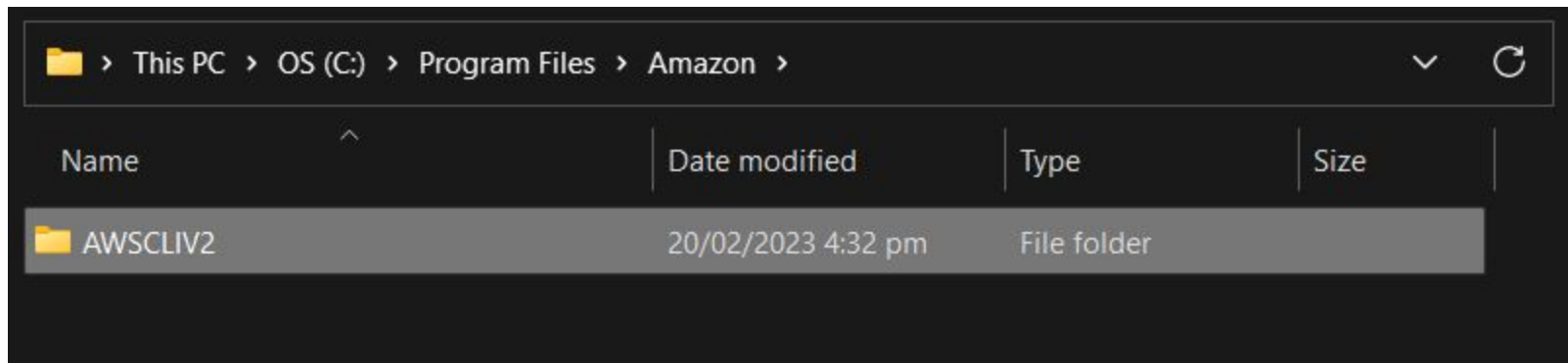
Windows Setup Guide for AWS CLI

4. Click **install** and you're set.



Windows Setup Guide for AWS CLI

5. You can check the file installed in your drive and in this tutorial, the folder **AWSCLIv2** is located at **Program Files > Amazon**.



Windows Setup Guide for AWS CLI

6. Now **open terminal** and **perform** the following command:

```
C:\> aws --version
```

```
C:\Users\USER>aws --version  
aws-cli/2.10.1 Python/3.9.11 Windows/10 exe/AMD64 prompt/off  
  
C:\Users\USER>_
```

Windows Setup Guide for AWS CLI

7. Now, with the **configure command**

```
C:\> aws configure
```

```
C:\Users\USER>aws configure
AWS Access Key ID [None]: 
AWS Secret Access Key [None]: 
Default region name [None]: 
Default output format [None]: 
```

Windows Setup Guide for AWS CLI

8. Finally, run the command `C:\> aws --version` to check everything is working

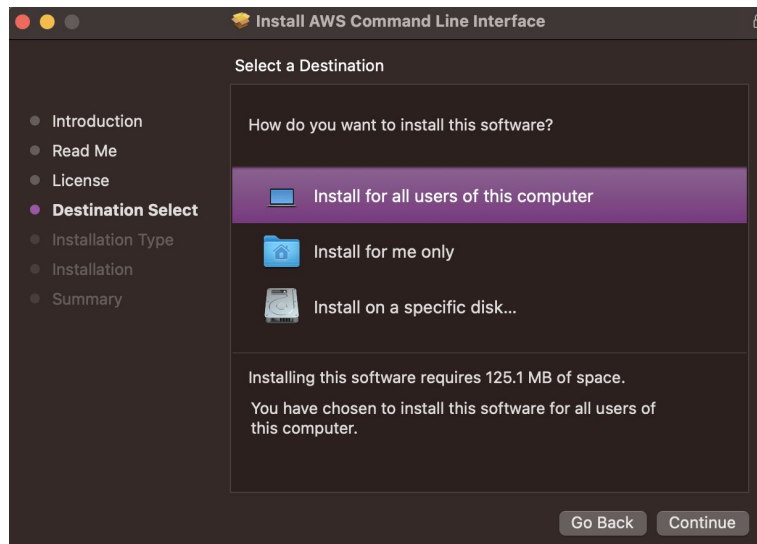
```
C:\Users\USER>aws --version
aws-cli/2.10.1 Python/3.9.11 Windows/10 exe/AMD64 prompt/off
```


MacOS Setup Guide for AWS CLI

Tools to Interact with AWS Services

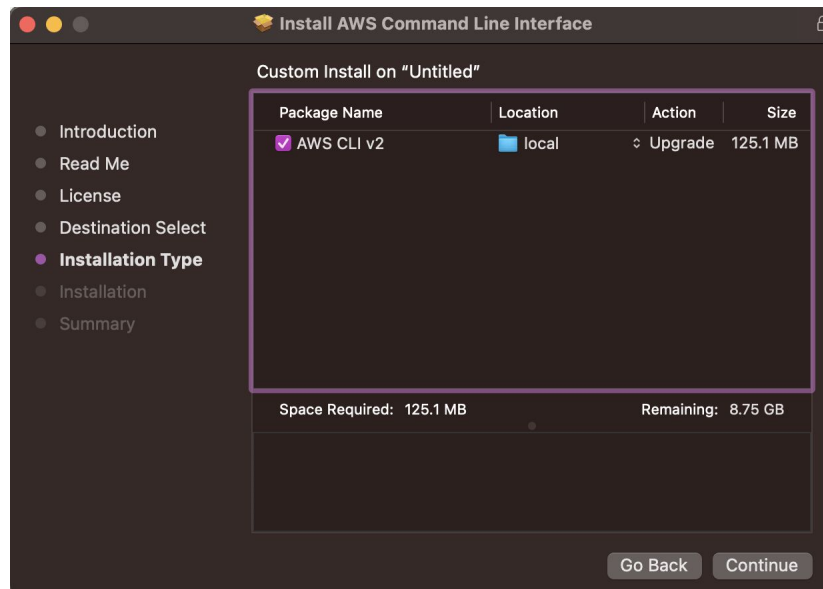
MacOS Setup Guide for AWS CLI

1. From Introduction to Readme, select **Continue** and **Agree** to the License Agreement.
2. For Destination Select, choose **Install for all users of this computer**.



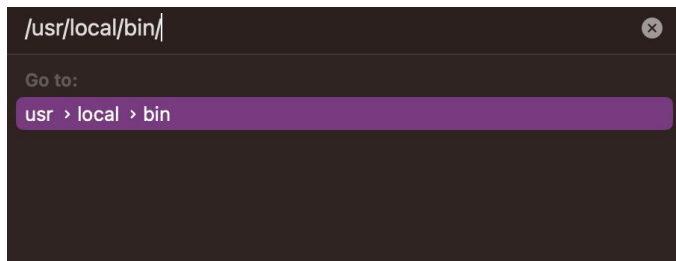
MacOS Setup Guide for AWS CLI

3. **Installation Type**, make sure that it is the same as the image below.

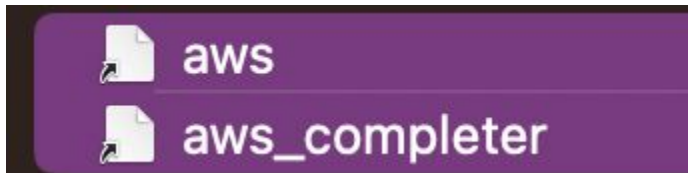


MacOS Setup Guide for AWS CLI

4. Click **continue**, enter your **password** if there's any and you'll see a
 - a. message that installation is successful.
5. **Optional**, you can check the files installed on your mac machine by
 - a. pressing Command ⌘ + Shift ⇧ + g and enter **/usr/local/bin**



6. **Optional**, you can go to that directory and check these 2 files installed



MacOS Setup Guide for AWS CLI

7. In your **Mac terminal** run the following commands:

```
$ which aws
```

```
$ aws --version
```

```
~ on 🌤️  
[> which aws  
/Users/zuitt/.pyenv/shims/aws
```

```
~ on 🌤️  
[> aws --version  
aws-cli/2.10.1 Python/3.9.11 Darwin/22.2.0 exe/x86_64 prompt/off
```

MacOS Setup Guide for AWS CLI

8. After checking the version and if it's really installed in your machine, next is our **configure command**:

```
$ aws configure
```

9. This will ask you to enter **some credentials** provided by your instructor

```
[> aws configure
AWS Access Key ID [*****]: 
AWS Secret Access Key [*****]: 
Default region name []: 
Default output format []:
```

```
~ on  (us-west-2) took 1m26s
> █
```

MacOS Setup Guide for AWS CLI

10. Finally, run the command `$ aws --version` to make sure everything is working.

```
~ on 🌥️ (us-west-2) took 1m26s  
[> aws --version  
aws-cli/2.10.1 Python/3.9.11 Darwin/22.2.0 exe/x86_64 prompt/off
```

AWS CloudShell

Tools to Interact with AWS Services

AWS CloudShell is a **browser-based, pre-authenticated shell** that you can launch directly from the AWS Management Console.

You can launch AWS CloudShell from the AWS Management Console, and the **AWS credentials** that you used to sign in to the console **are automatically available in a new shell session**.

This pre-authentication of AWS CloudShell users allows you to **skip configuring credentials** when interacting with AWS services using AWS CLI version 2. The AWS CLI is pre-installed on the shell's compute environment.

1. Sign in to your [AWS Management Console](#).
2. Click on the "**Services**" menu and select "**CloudShell**" under the "**Developer Tools**" section.
3. If you are using CloudShell for the first time, you will be prompted to set up an S3 bucket in your AWS account to store your CloudShell data. Follow the instructions provided by the setup wizard to complete this step.
4. Once CloudShell is launched, you will see a terminal window in your browser that you can use to run commands and access your AWS resources.

Quiz Form

Tools to Interact with AWS Services

Click on this link to navigate to the quiz form for the session on Introduction to AWS and EC2.

Make sure to **tick the checkbox and record your email with your response.**