

## Configure and Test the AWS CLI

### Overview:

In addition to accessing the AWS cloud through the AWS Console, and the AWS SDK, a command line Interface (CLI) is available. Similar to the AWS SDK, this interface, known as AWS CLI, allows one to generate and reuse scripts as opposed to using the AWS console GUI application. In most cases, the AWS CLI is efficient and easy to document and maintain. However; the AWS CLI does not support all services in the AWS cloud environment.

The AWS CLI runs on all popular operating systems including Windows, Linux and MacOS. The installation and downloads required are slightly different for each operating system. These details are available in the AWS CLI Users guide available in the reading materials for this week as well as on the AWS web site. However; we will be using the Cloud9 environment which already has the AWS-CLI installed. Therefore, there is no need to install the environment on your system to complete the assignments in the class.

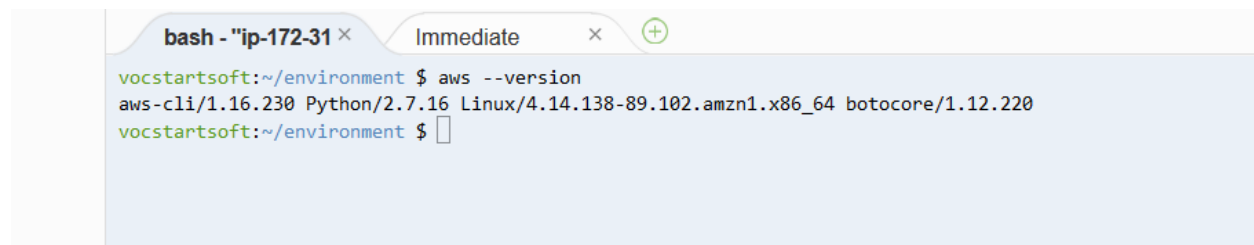
This document will provide screen captures and steps associated with using AWS-CLI through the AWS Educate classroom and your Cloud9 environment. Once installed in any environment, the commands and functionality will work the same in most cases.

### Confirming the AWS-CLI in your Cloud9 Environment:

To confirm the AWS-CLI is properly installed, login to your AWS Educate classroom, launch your Cloud9 IDE, use a bash terminal and type:

```
aws --version
```

Figure 1 shows some typical results after after invoking the version command. Be aware the version information returned may be different depending upon the default setting for your Cloud9 environment.

A screenshot of a Cloud9 IDE terminal window. The window has two tabs: 'bash - "ip-172-31"' and 'Immediate'. The terminal shows the prompt 'vocstartsoft:~/environment \$' followed by the command 'aws --version'. The output is 'aws-cli/1.16.230 Python/2.7.16 Linux/4.14.138-89.102.amzn1.x86\_64 botocore/1.12.220'. The prompt is now 'vocstartsoft:~/environment \$' with a cursor.

```
bash - "ip-172-31" x Immediate x +
vocstartsoft:~/environment $ aws --version
aws-cli/1.16.230 Python/2.7.16 Linux/4.14.138-89.102.amzn1.x86_64 botocore/1.12.220
vocstartsoft:~/environment $
```

Figure 1 AWS CLI Version Check using the Clou9 bash shell

### Configure the AWS CLI:

Since you are using the Cloud9 with AWS-CLI preinstalled the keys for access have already been configured and need no additional changes. In fact, you should **not** attempt to configure them as you could lose access to the environment from the AWS-CLI interface. If this does happen, you can just delete the corrupted Cloud9 environment and spin-up another one. This is another huge advantage of programming in the cloud.

## Using the AWS CLI:

Similar to other command line tools, the AWS CLI is run using the shell or command prompt of your operating system and includes command line options preceded by two hyphens. For this class, we will be using the bash shell of the Cloud9 environment.

The format of an AWS CLI call is:

```
aws [options] <command> <subcommand> [parameters]
```

The items in brackets “[ ]” are optional.

The following command line options are frequently used within AWS:

--profile – name of a profile to use, or "default" to use the default profile.

--region – AWS region to call.

--output – output format.

--endpoint-url – The endpoint to make the call against.

Other command options are available and can be viewed by using the help command.

The AWS CLI tool provides access to hundreds of possible commands, subcommands and parameters. With this diversity and complexity, the help command can be quite useful. The general help command can be invoked by typing:

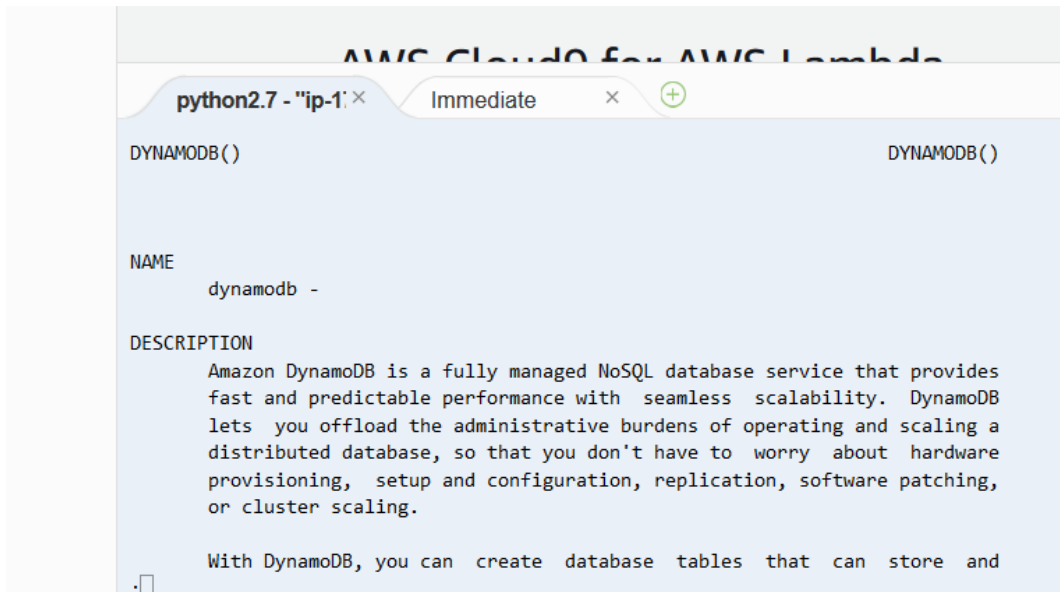
```
aws help
```

To obtain help on a specific command or service, type the name of the service prior to the help subcommand:

```
aws snowball help
```

You scroll through the help commands by hitting the “enter” key. To exit or escape from the help display use the character ‘q’.

Figure 2 shows the first part of the results of entering “aws dynamodb help” at the command prompt.



```
python2.7 - "ip-1" × Immediate × +
DYNAMODB() DYNAMODB()

NAME
    dynamodb -

DESCRIPTION
    Amazon DynamoDB is a fully managed NoSQL database service that provides
    fast and predictable performance with seamless scalability. DynamoDB
    lets you offload the administrative burdens of operating and scaling a
    distributed database, so that you don't have to worry about hardware
    provisioning, setup and configuration, replication, software patching,
    or cluster scaling.

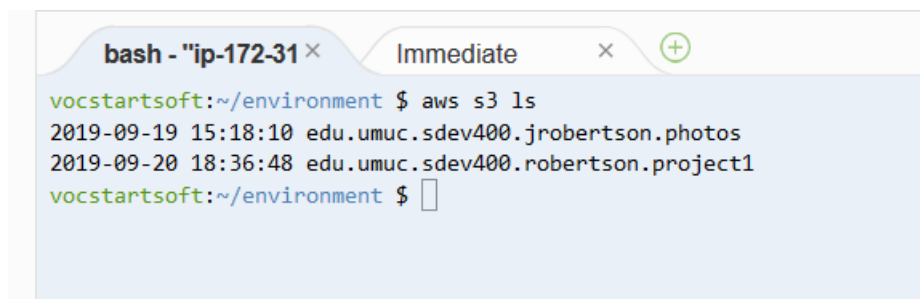
    With DynamoDB, you can create database tables that can store and
```

Figure 2 Calling AWS dynamoDB help

Although the available information, can vary between help commands, most help requests will return the name of the command, a description, and available commands or sub-commands.

Each of AWS services can be accessed through the AWS CLI using specific commands and subcommands. For example, to list the available S3 buckets in your AWS Educate classroom account, use the following command (See the output in figure 3):

```
aws s3 ls
```



```
bash - "ip-172-31" × Immediate × +
vocstartsoft:~/environment $ aws s3 ls
2019-09-19 15:18:10 edu.umuc.sdev400.jrobertson.photos
2019-09-20 18:36:48 edu.umuc.sdev400.robertson.project1
vocstartsoft:~/environment $
```

Figure 3 Listing the S3 buckets

In the above call, “s3” is the command and ls is the sub-command.

There is significant flexibility available for management of the S3 buckets through the AWS CLI. For example, not only can files be moved from your Cloud9 environment to the S3 buckets using a few simple commands, but they can be moved between existing S3 buckets. If you need to copy, move or delete an image or document, this can be easily accomplished with commands nearly identical to popular LINUX commands such as cp, mv and rm.

Table 1 provides a list of AWS CLI commands used for S3 management along with some specific examples using the folder structure presented earlier in this document. Note the s3:// portion of the bucket name is not required to resolve ambiguity and may be omitted.

Table 1 AWS CLI Commands and Examples for S3

AWS CLI Call	Description
<code>aws s3 help</code>	List help options and overview of the S3 service.
<code>aws s3 mb help</code>	Describes help associated with creating a bucket in S3 through the AWS CLI tool.
<code>aws s3 mb s3://edu.umuc.sdev400.robertson.project1</code>	Creates a bucket named edu.umuc.sdev400.robertson.project1
<code>aws s3 rb s3://edu.umuc.sdev400.robertson.project1</code>	Removes a bucket named edu.umuc.sdev400.robertson.project1 (use --recursive to remove files as well)
<code>aws s3 ls</code>	Lists all buckets.
<code>aws s3 ls s3://edu.umuc.sdev400.jrobertson.photos</code>	Lists folders of the photos bucket.
<code>aws s3 ls s3://edu.umuc.sdev400.jrobertson.photos/2017/</code>	list files in subdirectory of California2015. Note the use of the / at the end to denote the directory. If omitted only the Pre(fix) of 2017 is listed.
<code>aws s3 cp Picture1.png s3://edu.umuc.sdev400.jrobertson.photos/2019/</code>	Copy the Picture1.png file from your Cloud9 current folder to the 2019 S3 bucket folder. Note: you must upload the Picture1.png file from your desktop to the Cloud9 current folder for this to work.
<code>aws s3 mv s3://edu.umuc.sdev400.jrobertson.photos/2018/Picture1.png s3://edu.umuc.sdev400.jrobertson.photos/2017/</code>	Moves Picture1.png from the 2018 folder to the 2017 folder.
<code>aws s3 rm s3://edu.umuc.sdev400.jrobertson.photos/2017/Picture1.png</code>	Remove the Picture1.png from the 2017 folder.
<code>aws s3 ls s3://edu.umuc.sdev400.jrobertson.photos/ --recursive --summarize</code>	Recursively lists folders and files in photos and summarize the results

When testing these commands you will get a positive response from the AWS CLI prompt as shown in figure 4.

```

vocstartsoft:~/environment $ aws s3 ls s3://edu.umuc.sdev400.jrobertson.photos/ --recursive --summarize
2019-09-19 16:16:45      0 2017/
2019-09-19 16:28:12 1385140 2017/CAM00132.jpg
2019-09-19 16:29:58 1297527 2017/CAM00151.jpg
2019-09-19 16:29:58 2263961 2017/CAM00153.jpg
2019-09-19 16:18:07      0 2018/
2019-09-19 16:18:13      0 2019/

Total Objects: 6
Total Size: 4946628
vocstartsoft:~/environment $

```

Figure 4 Response from successful AWS CLI command

If you want visual confirmation of the action, you should go to the AWS Management console and navigate to the S3 service interface. Then navigate the buckets to review the contents. Be aware that often a refresh is needed on your browser to show the most current results.

To help automate the running AWS CLI commands, batch or script files can be created. Each script file could list several commands and be executed at the shell prompt.

For example, the following script file, saved as AWSCLIS3.sh within the Cloud9 environment can be executed resulting in the execution of several AWS CLI commands sequentially.

```

#####
# Script to Demo S3 AWS CLI
# SDEV400
#####

# Make a project 2 bucket
aws s3 mb s3://edu.umuc.sdev400.robertson.project22020

# List the current buckets
aws s3 ls

# Copy some images to folders
# Note I already uploaded the Images to Cloud9
aws s3 cp CAM00190.jpg s3://edu.umuc.sdev400.robertson.project22020/2019/
aws s3 cp CAM00198.jpg s3://edu.umuc.sdev400.robertson.project22020/2019/

# copy a file from one folder to another
aws s3 cp s3://edu.umuc.sdev400.robertson.project22020/2019/CAM00190.jpg
s3://edu.umuc.sdev400.robertson.project1/2019/

# Check the folders
aws s3 ls s3://edu.umuc.sdev400.robertson.project22020/ --recursive --
summarize
aws s3 ls s3://edu.umuc.sdev400.robertson.project1/ --recursive --summarize

```

To run the shell script within the Cloud9 Linux environment several steps are required:

1. Be sure to create the file within the Cloud9 environment. Just use File->New File to create the file. Name the file using a .sh extension. (See figure 5)

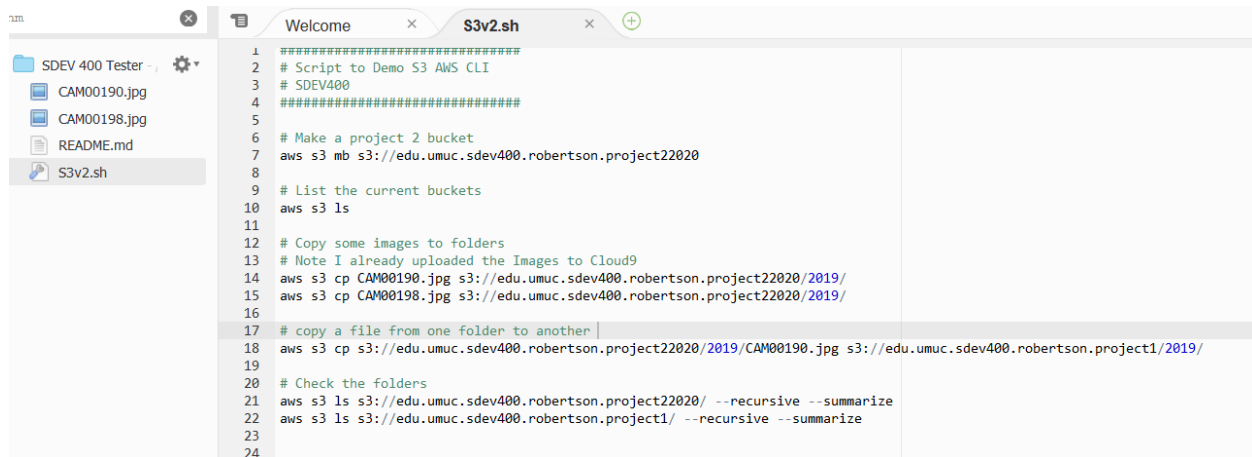


Figure 5 Saving the Shell Script within the Cloud9 environment

2. You need to make the shell script executable. By default the permissions are just read and write. To make it executable, just run the `chmod +x filename` command at the shell script. See figure 6. You can use the `ls -al` command to verify the script is now executable as you will see the `-rwxr-xr-x` label.

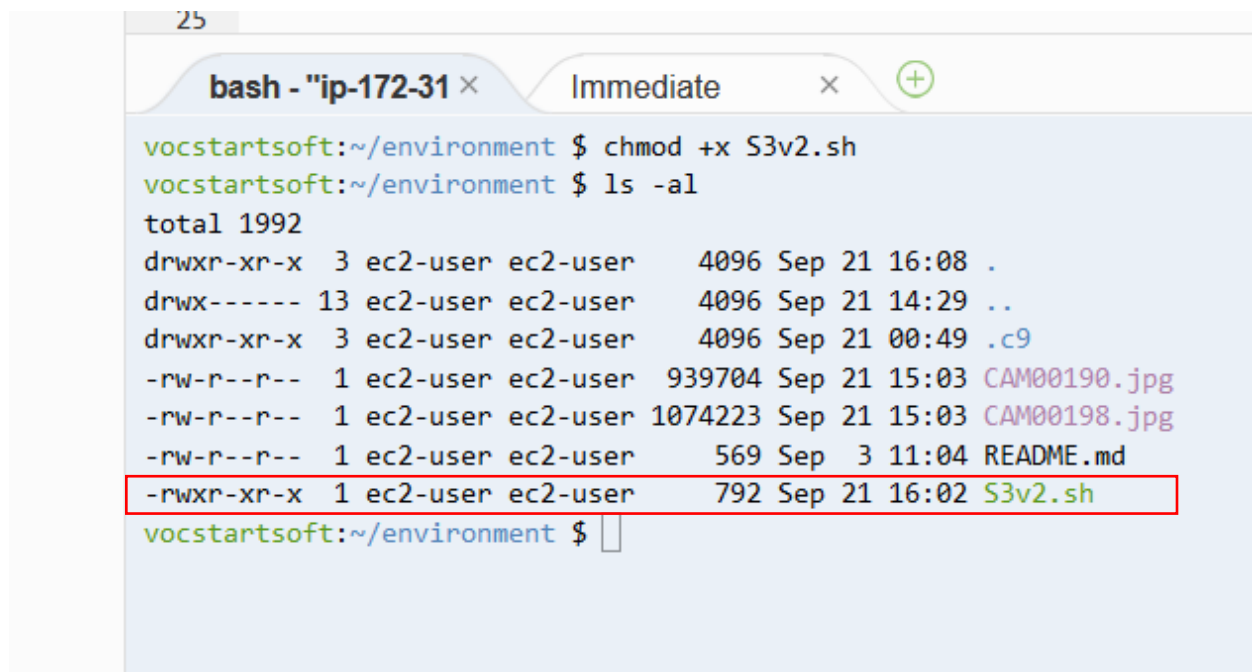


Figure 6 Changing the mode to Executable in Linux

3. To run the script use `./filename`. For example,
 

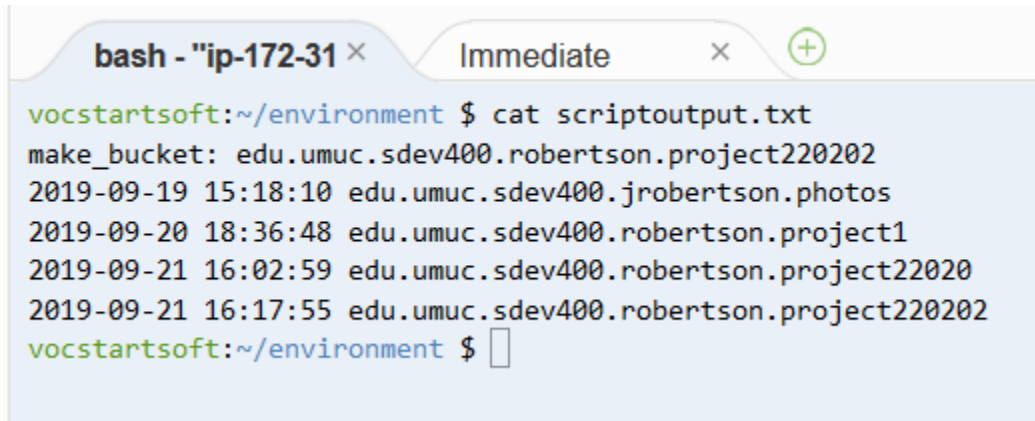
```
./S3v2.sh
```
4. You can redirect the output of the script to a file with the `>` redirection operator. For example
 

```
./S3v2.sh > scriptoutput.txt
```

You can view the results of the redirected output by downloading file from your Cloud9 environment of using the `cat` linux command. To download the file to your desktop select the file and use a right-mouse click to select download. To view with the cat command just type `cat filename` at the bash shell terminal in Cloud 9:

```
cat scriptoutput.txt
```

Figure 7 shows the results of using the cat command.



```
bash - "ip-172-31" x Immediate x +
vocstartsoft:~/environment $ cat scriptoutput.txt
make_bucket: edu.umuc.sdev400.robertson.project220202
2019-09-19 15:18:10 edu.umuc.sdev400.jrobertson.photos
2019-09-20 18:36:48 edu.umuc.sdev400.robertson.project1
2019-09-21 16:02:59 edu.umuc.sdev400.robertson.project22020
2019-09-21 16:17:55 edu.umuc.sdev400.robertson.project220202
vocstartsoft:~/environment $
```

Figure 7 Using cat to display a file in Linux

Finally, when creating buckets in S3 remember they must be globally unique and DNS-compliant. This means if you create a bucket using a command or within a script from AWS CLI or the AWS SDK, no one else in the world will be able to create a bucket with that same name. This includes you. So, now that `edu.umuc.edu.sdev400.robertson.project1` has been created, it can't be created again until it is deleted. I mention this because often in testing scripts, a script might be run multiple times before all of the components work. When doing so, your bucket name needs be changed or you should delete it to avoid this issue.