AWS Academy Data Engineering - Sandbox

Environment Overview

This lab provides a sandbox environment for ad-hoc exploration of AWS services that appeared in other labs in the course.

This environment is NOT long-lived. When the session timer runs to 0:00, the session will end, and any data and resources that you created in the AWS account will be permanently deleted.

IMPORTANT: Monitor your budget in the lab interface above. Whenever you have an active lab session, the latest known remaining budget information will display at the top of this screen. This data comes from AWS Budgets which typically updates every 8 to 12 hours. Therefore *the remaining budget that you see may not reflect your most recent account activity*.

Environment Navigation

Use the **Readme** link above to return to these instructions at any time.

Use the **AWS Details** link above to access information about your environment.

After you start the lab, the AWS Details panel will display and you will need to choose **Readme** to return to these instructions.

Tip: you can resize this panel at anytime by dragging the bar to the left of these instructions to make it wider or narrower.

The terminal window to the left of these instructions can be used to run AWS CLI commands or code such as AWS SDK for Python code (details provided below).

Access the AWS Management Console

1. At the top of these instructions, choose
   Start Lab to start the lab session.
      a. The lab session is started and session information is displayed.
      b. A timer above shows the time remaining in the session.

Tip: You can refresh the session length at any time by choosing Start Lab again before the timer reaches 0:00.

2. Connect to the AWS Management Console by choosing the **AWS** link above the terminal window.
      a. You should be connected to the AWS Management Console.

Tip: If a new browser tab does not open, a banner or icon is usually at the top of your browser with the message that your browser is preventing the site from opening pop-up windows. Choose the banner or icon, and then choose **Allow pop-ups**.

Region restriction

All service access is limited to the **us-east-1** and **us-west-2** Regions. If you load a service console page in another AWS Region you will see access error messages.

Service usage and other restrictions

The following services can be used. Specific limitations apply as documented. Services restrictions are subject to change.

- Amazon Athena
  - This service can assume the LabRole IAM role.
- AWS Cloud9
  - Supported Instance types: nano, micro, and small.
- AWS CloudFormation
  - This service can assume the LabRole IAM role.
- AWS CloudShell
- Amazon Elastic Compute Cloud (EC2)
  - Supported AMIs: AMIs where the owner is Amazon.
    - To launch a **Windows** or **Amazon Linux** instance: Choose "Launch Instances", then choose from the ones available in the "Quick Start" tab.
  - Supported Instance types: nano, micro, and small. t2 or t3 series only.
  - Instance quantity - Maximum of 32 vCPU used by concurrently running instances in an AWS Region. You are also limited to launching no more than nine (9) instances (of any size) in a Region at once. *Recommendation*: size to your actual need to avoid using up your cost budget.
  - On-Demand instances only.
  - EBS volumes - sizes up to 35 GB and type must be General Purpose SSD (gp2, gp3) cold HDD (sc1), or standard.
  - Key pairs - If you are creating an EC2 instance in any AWS Region other than us-east-1, the vockey key pair will not be available. In such cases, you should create a new key pair and download it when creating the EC2 instance. Then use the new key pair to connect to that instance.
  - A role named **LabRole** and an instance profile named **LabInstanceProfile** have been pre-created for you. You can attach the role (via the instance profile) to an EC2 instance when you want to access an EC2 instance (terminal in the browser) using AWS Systems

Manager Session Manager. The role also grants permissions to any applications running on the instance to access many other AWS services from the instance.

- Amazon EMR
  - This service can assume the LabRole IAM role.
  - Cluster instances sizes must comply with the restrictions listed above for the EC2 service.
- AWS Glue
 - This service can assume the LabRole IAM role.

Note: The following limitations apply to the AWS Glue ETL job configuration

  - Allowed Worker type: G.1X, Standard
  - Maximum number of workers: 10
  - Maximum Concurrency: 1
- AWS Identity and Access Management (IAM)
  - Limited access. You cannot create users or groups. You cannot create roles, except that you can create service-linked roles.
  - Service role creation is generally permitted. If the service needs to create a role for you, you may need to retry role creation if it fails the first time.
  - A role named **LabRole** has been pre-created for you. This role is designed to be used when you want to attach a role to a resource in an AWS service. It grants many AWS services access to other AWS services and has permissions very similar to the permissions you have as a user in the console.
    - Example use: attach the LabRole via the instance profile named **LabInstanceProfile** to an EC2 instance for terminal in the browser access to an EC2 instance guest OS using AWS Systems Manager Session Manager.
    - Another example: Attach the LabRole to a Lambda function so that the Lambda function can access S3 or some other permitted service.
- AWS IoT Core

- o This service can assume the LabRole IAM role.
- AWS Key Management Service (KMS)
- AWS Lambda
  - o **Tip**: Attach the existing **LabRole** to any function that you create if that function will need permissions to interact with other AWS services.
- Amazon Redshift
  - o Supported instance type: ra3.large
  - o Supported cluster size: maximum two instances
- Amazon Simple Notification Service (SNS)
  - o This service can assume the LabRole IAM role.
- Amazon Simple Queue Service (SQS)
  - o This service can assume the LabRole IAM role.
- Amazon Simple Storage Service (S3)
- AWS Step Functions
  - o This service can assume the LabRole IAM role.
- AWS Systems Manager (SSM)

Accessing a Terminal to run AWS CLI commands or AWS SDK code

Tip: Use AWS CloudShell or launch an AWS Cloud9 instance. You can access CloudShell it in the AWS Management Console, at the top of the screen, by choosing the **AWS CloudShell** icon (highlighted in red in the screen capture below).
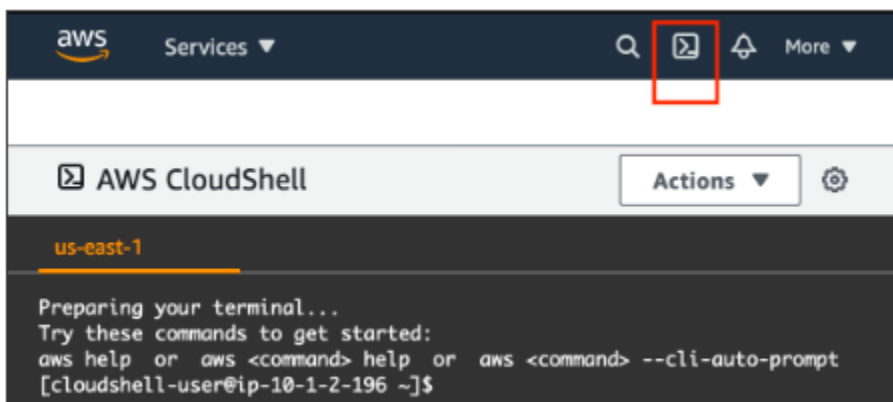
Running AWS CLI commands

Here is an example AWS CLI command to try running in the CloudShell or AWS Cloud9 terminal. If you have created any EC2 instances in the default account Region, running this command will provide information about them:

```
aws ec2 describe-instances
```

See the [AWS CLI Command Reference](#) documentation for details on how to use the AWS CLI.

Using the AWS SDK for Python

The AWS Cloud9 and AWS CloudShell terminals also have Python 3 installed with the boto3 library available. You can use it to run AWS Python SDK code. For example:

```
$ python3
>>> import boto3>>> ec2 = boto3.client('ec2', region_name='us-east-1')
>>> ec2.describe_regions()>>> exit()$
```

See the [documentation](#) for details on how to use the AWS SDK for Python.