Lab 2 Hadoop - AWS EMR

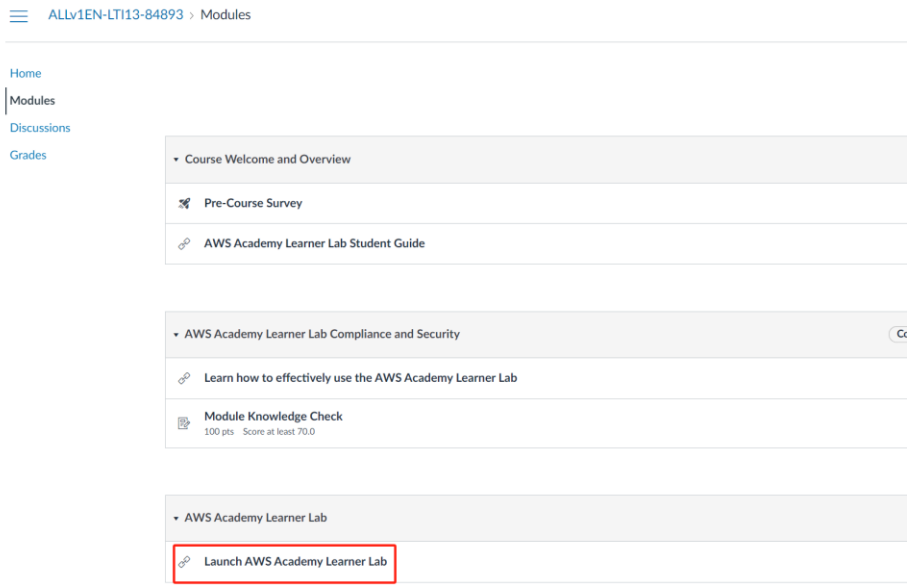
In this lab, you will create a simple Amazon EMR cluster in the AWS Management Console. After you create the cluster, you will submit a Hive script to process sample data stored in Amazon Simple Storage Service (Amazon S3). Please note that the cluster should be terminated after the lab!

Task 1 Launch an Amazon EMR cluster

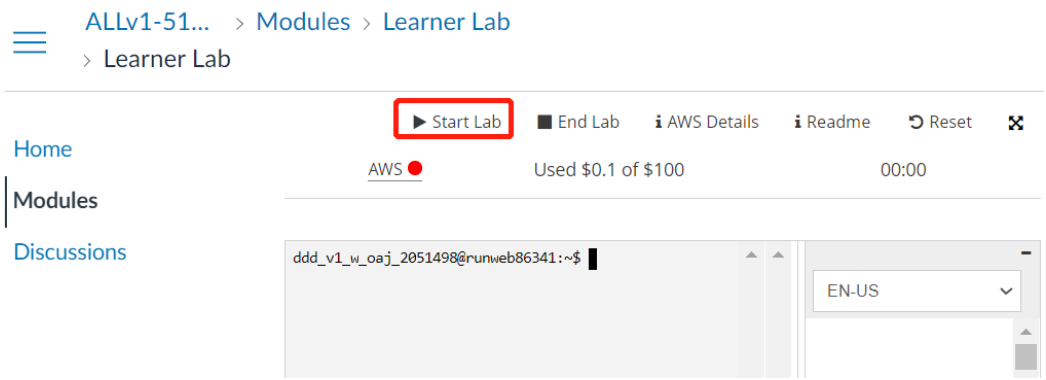
Task 1.1 Login AWS Academy account

- If you haven't completed AWS Academy account registration.
- You should receive an email from AWS, and please follow the instructions. You may also refer to the Lab 1.
- Open Amazon Academy Canvas: https://awsacademy.instructure.com/
- Please Login as a Student.

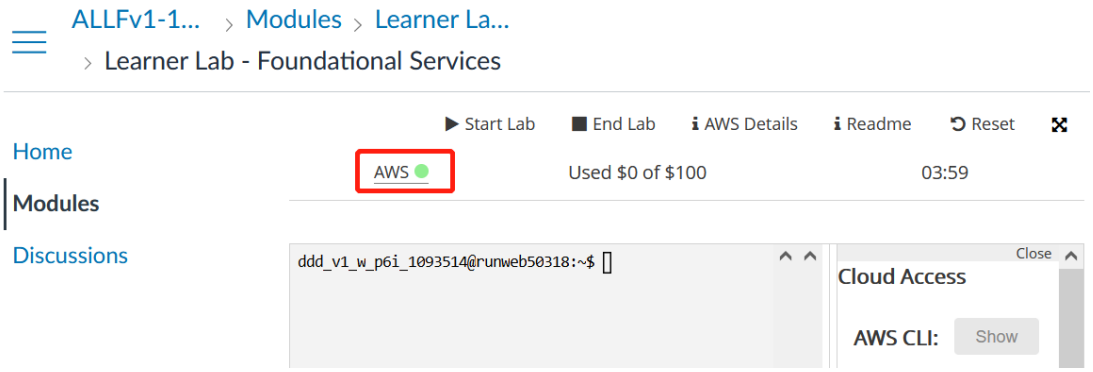
- Please go to the course: AWS Academy Learner Lab
- Under the Modules, please select Leaner Lab



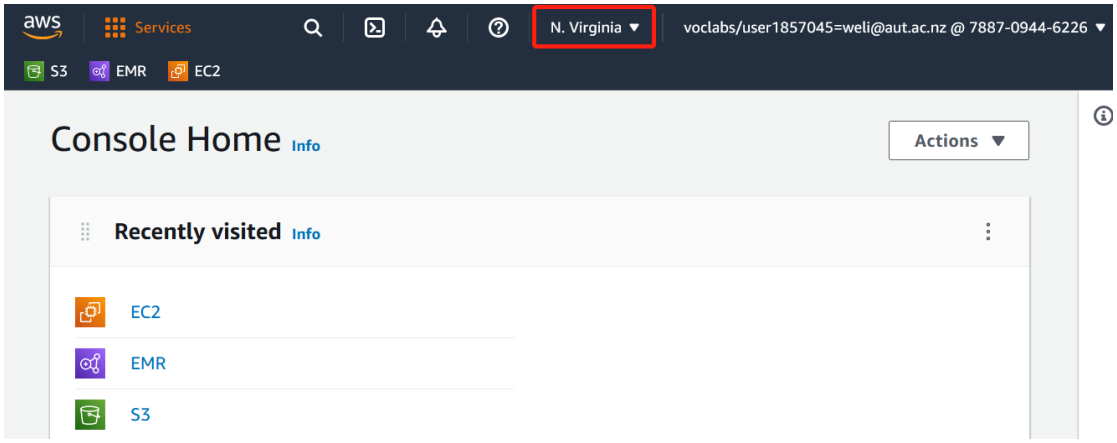
- Click Start Lab



- When you find the red dot becomes green, it means the AWS Services are ready to use.



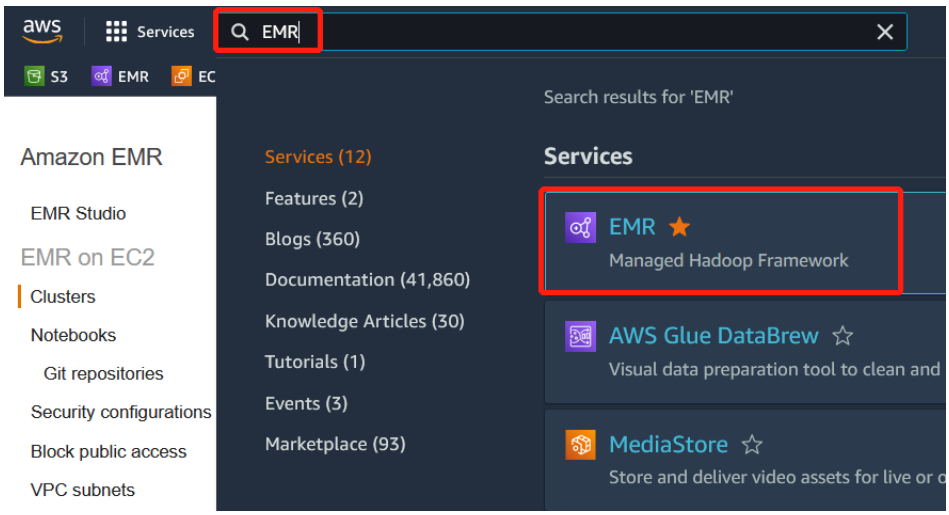
- Click AWS next to the green dot. You will see the screen below. You are not supposed to change the region, because the services for educate account are only supported in the *N. Virginia region*. You may find the connection a bit slow but no choice.



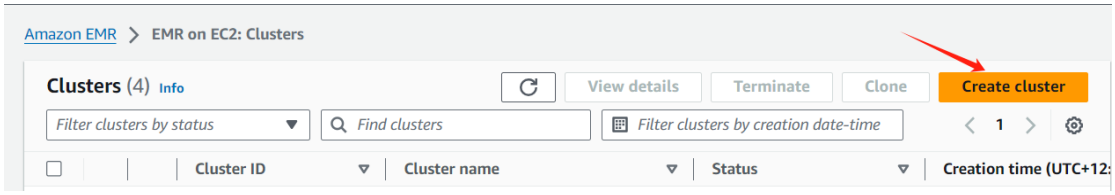
Task 1.2 Launch Your First Amazon EMR Cluster

In this task, you launch your first Amazon EMR cluster by using Quick Options in the Amazon EMR console and leaving most options to their default values. To learn more about these options, see Summary of Quick Options after the procedure. You can also select Go to advanced options to explore the additional configuration options available for a cluster.

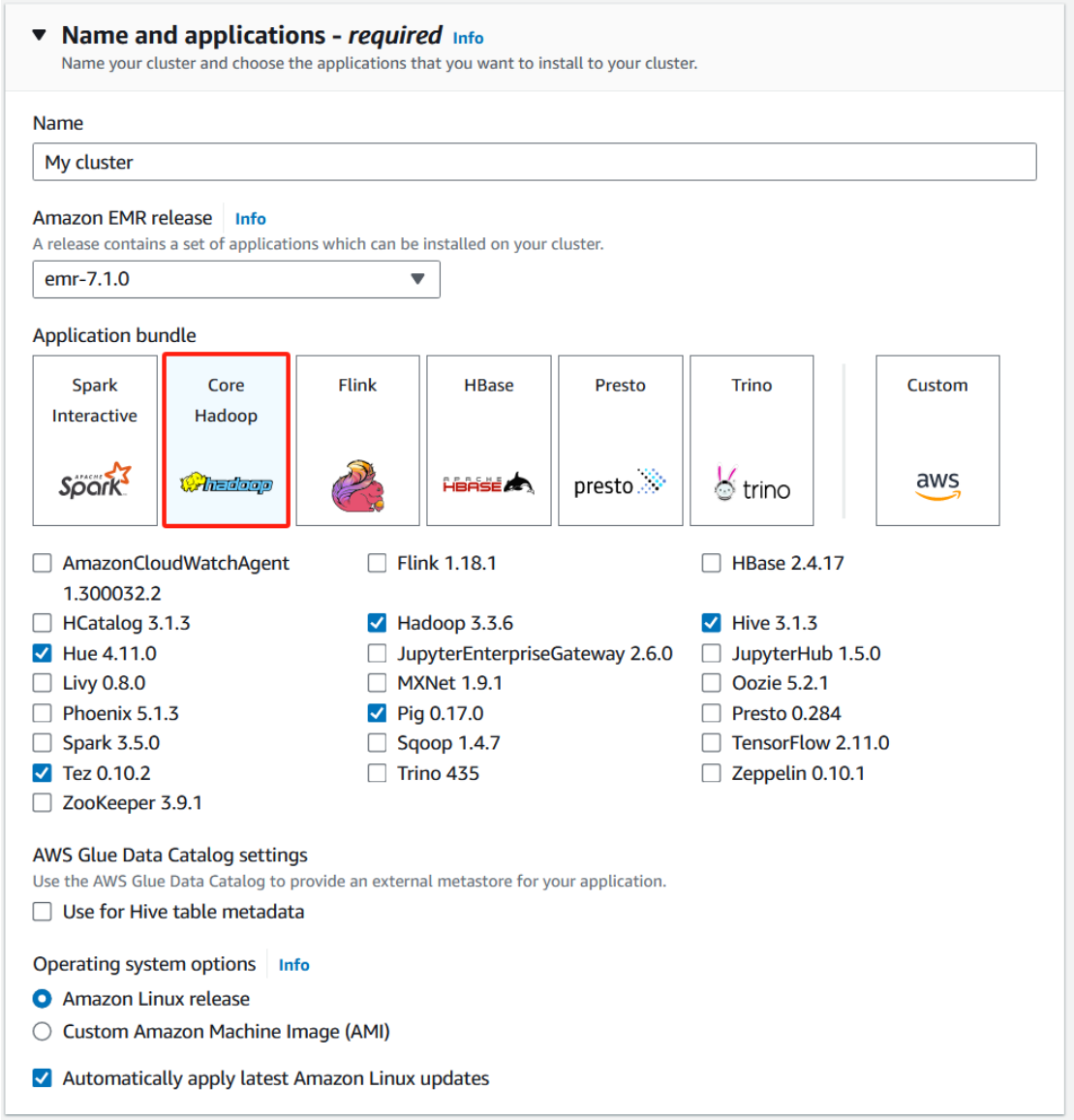
Find EMR from the service list and click it.



Choose Create cluster.



Follow the below steps to give the options and create a cluster.



Cluster configuration - *required* Info

Choose a configuration method for the primary, core, and task node groups for your cluster.

- (●) **Uniform instance groups**
  Choose the same EC2 instance type and purchasing option (On-Demand or Spot) for all nodes in your node group. Learn more
- ( ) **Flexible instance fleets**
  Choose from the widest variety of provisioning options for the EC2 instances in your cluster. Diversify instance types and purchasing options, and use an allocation strategy. Learn more

Uniform instance groups

Primary

Choose EC2 instance type

m4.large
2 vCore 8 GiB memory EBS only storage
On-Demand price: - Lowest Spot price: -

Actions

[ ] Use high availability
Launch highly available, more resilient cluster with three primary nodes on On-Demand Instances. This configuration applies for the lifetime of your cluster. Learn more

▶ **Node configuration - *optional***

Core

Choose EC2 instance type

m4.large
2 vCore 8 GiB memory EBS only storage
On-Demand price: - Lowest Spot price: -

Actions

▶ **Node configuration - *optional***

Task 1 of 1

Remove instance group

Name

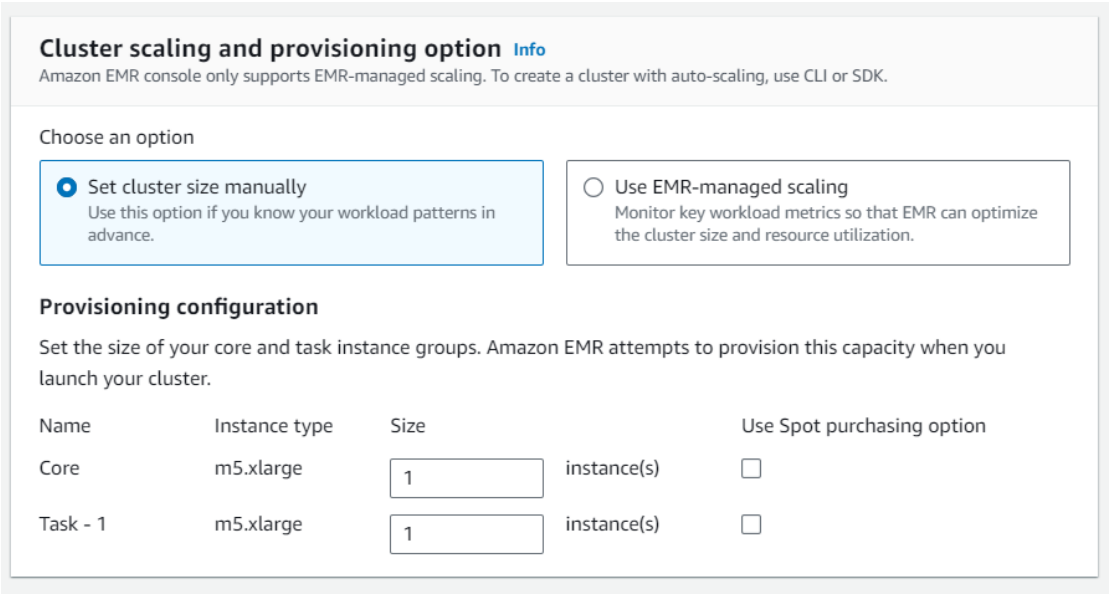
Task - 1

Choose EC2 instance type

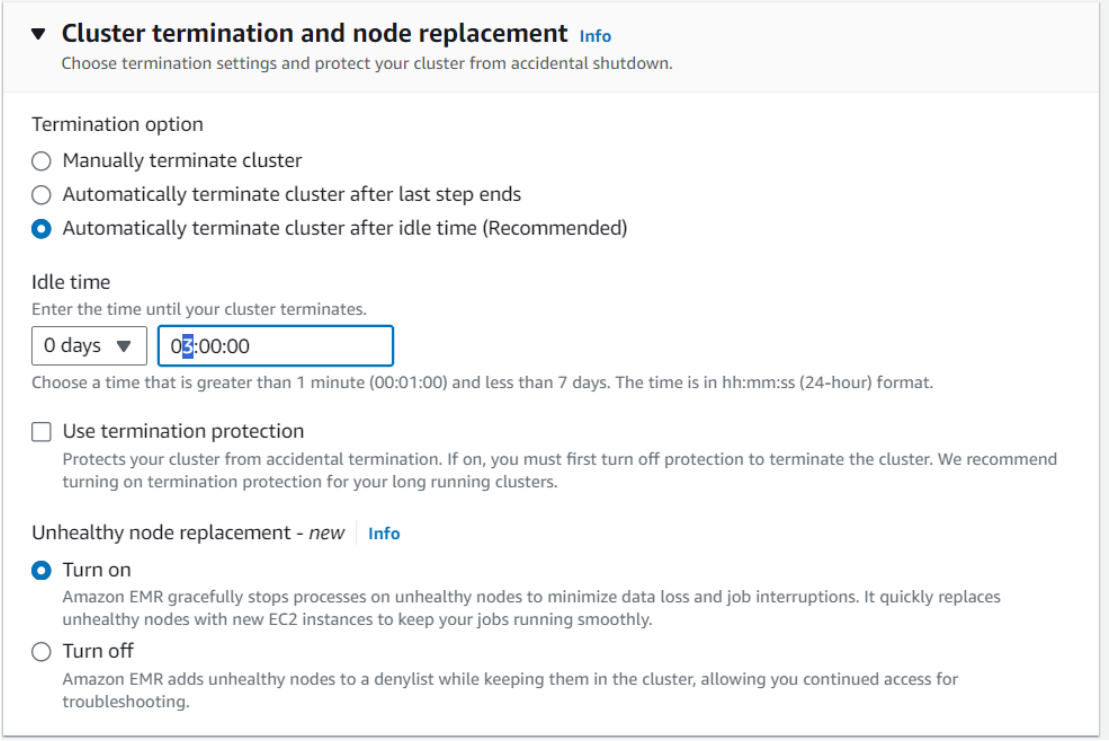
m4.large
2 vCore 8 GiB memory EBS only storage
On-Demand price: - Lowest Spot price: -

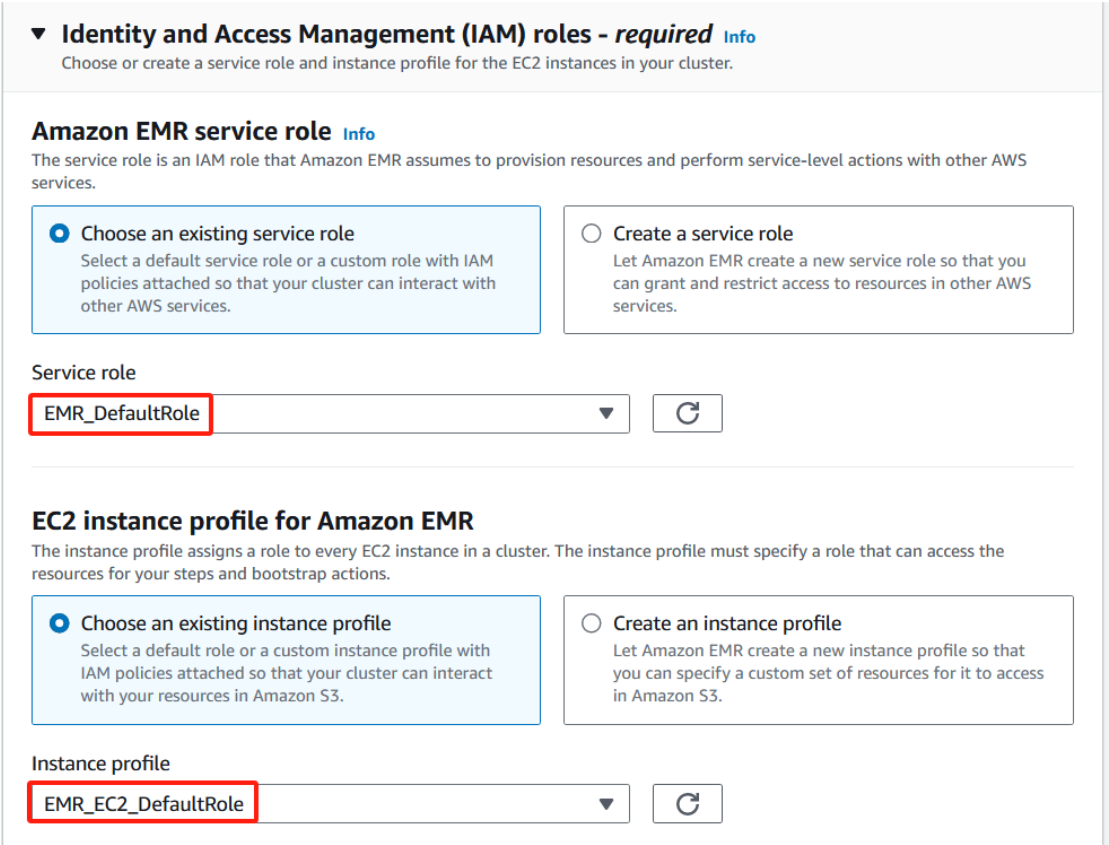
Actions

▶ **Node configuration - *optional***

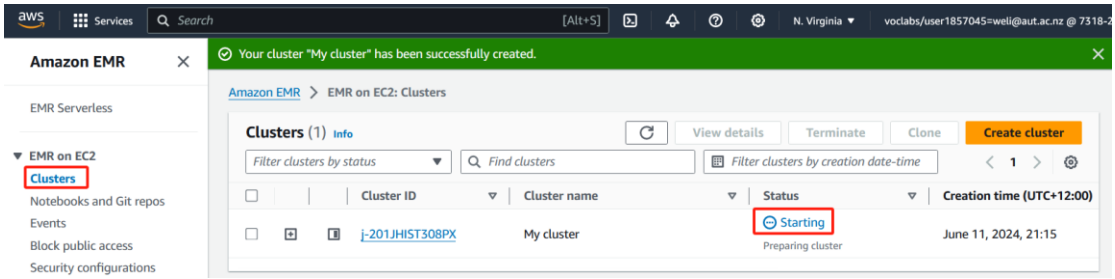


Let's change the cluster termination time from 1 hour to 3 hours.

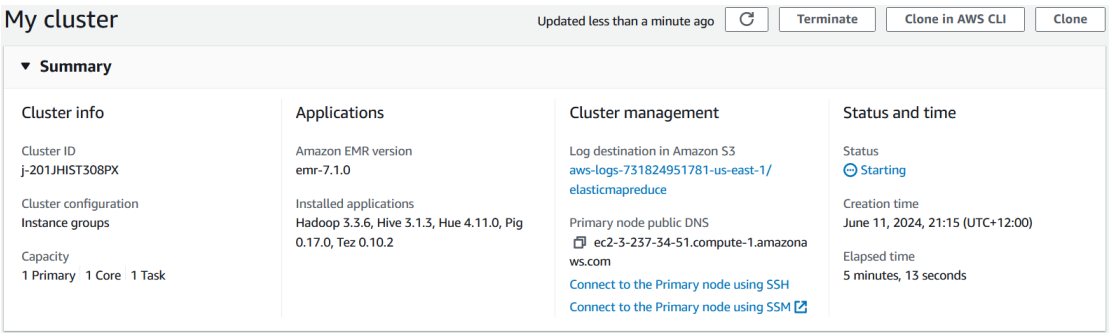




After that, choose “Create cluster”, and you will find your cluster will be running after a few minutes. The cluster status page with the cluster Summary appears. You can use this page to monitor the progress of cluster creation and view details about cluster status. As cluster creation tasks finish, items on the status page update. You may need to choose the refresh icon on the right or refresh your browser to receive updates.



Click the cluster ID of your new cluster, and you will see its summary and settings.

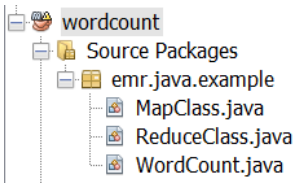


Task 2 Elastic MapReduce Hadoop Job Using Custom Jar

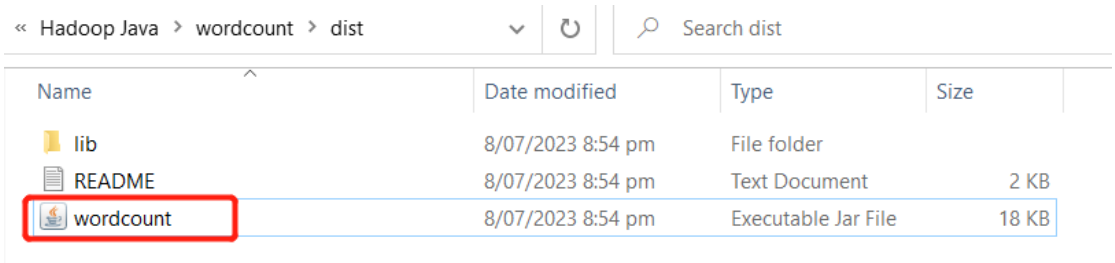
Download *input.txt* file, and wordcout.jar from Canvas, or prepare your own text file to be processed.

Task 2.1 Prepare Java map-reduce program (Optional)

Download *wordcount* project from Canvas, unzip it and then open this project using NetBeans IDE.



Walkthrough all the three classes and compile the project. *wordcount.jar* is a compiled version of this project.

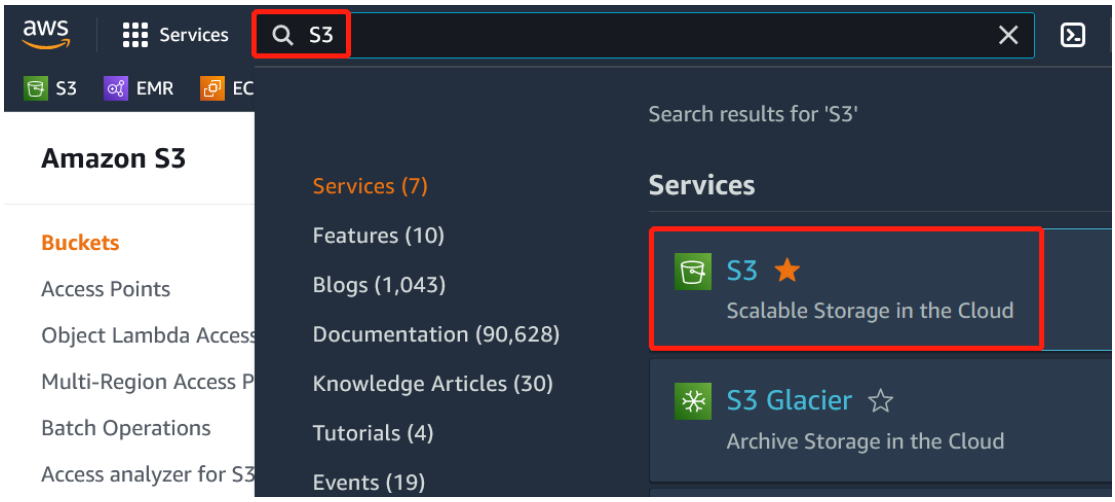


If you want to simulate MapReduce without the assistance of AWS EMR, please try *WordCount.java* under the *emr.simulation.wordcount* package.

Task 2.2 Create Amazon S3 Bucket

In this task, you need to specify an Amazon S3 bucket and create two folders to store the input data (files to be processed by your Java program) and your Java program – the jar file.

To access Amazon S3, please search S3 and click S3 from the result list.



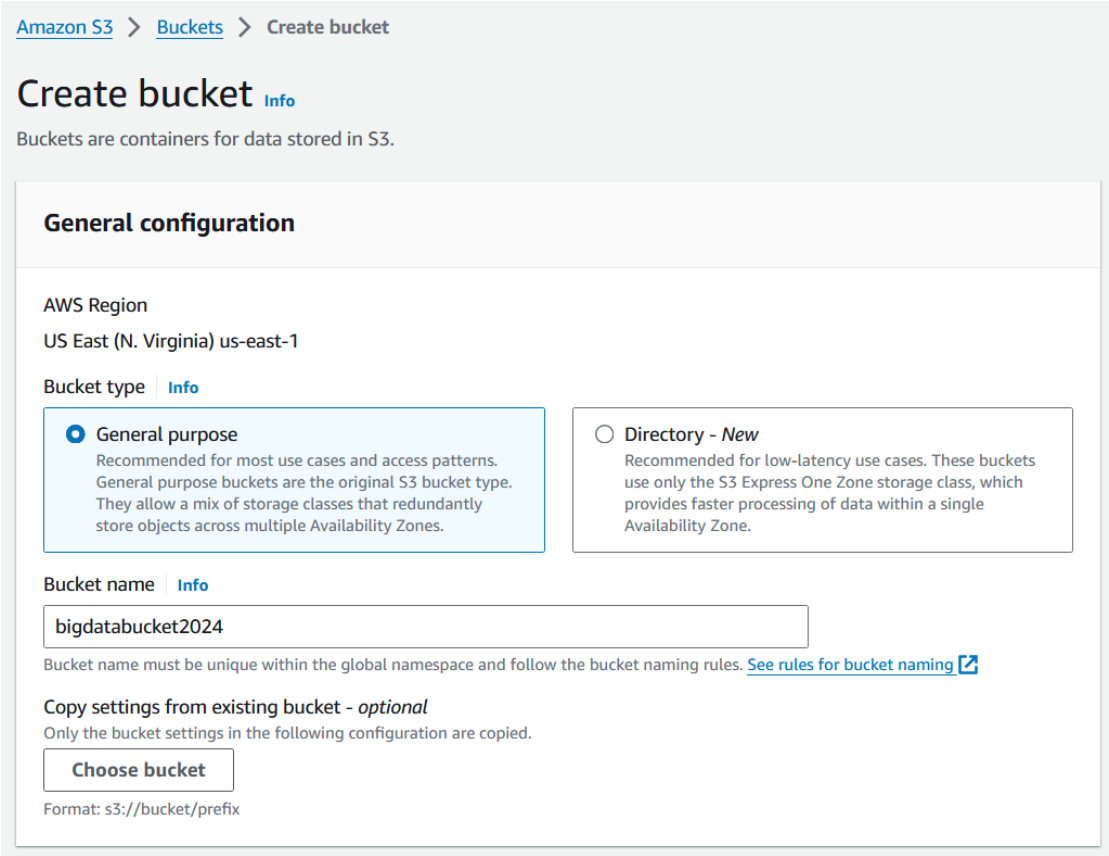
Because of Hadoop requirements, bucket and folder names that you use with Amazon EMR have the following limitations:

- They must contain only letters, numbers, periods (.), and hyphens (-).
- They cannot end in numbers.
- Bucket names must be unique *across all AWS accounts*.

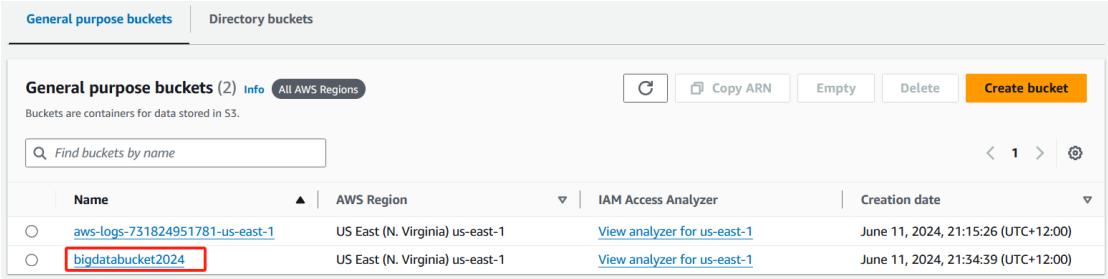
For more information about creating a bucket, see Create a Bucket in the *Amazon Simple Storage Service Getting Started Guide*. After you create the bucket, choose it from the list and then choose Create folder, replace the New folder with a name that meets the requirements, and then choose Save.

The bucket and folder name used later in the tutorial is *s3://bigdatabucket2024/input* and *s3://bigdatabucket2024/jar*

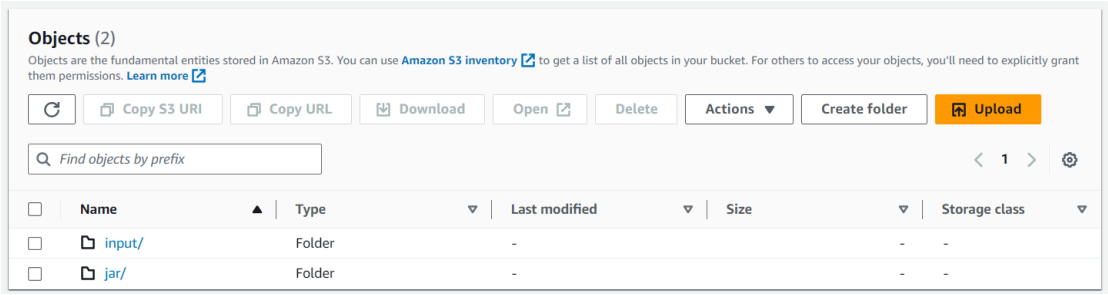
Yours will be different, which means you cannot use bigdatabucket2024 if it has already been used by others.



Use the default settings for the rest of the options.

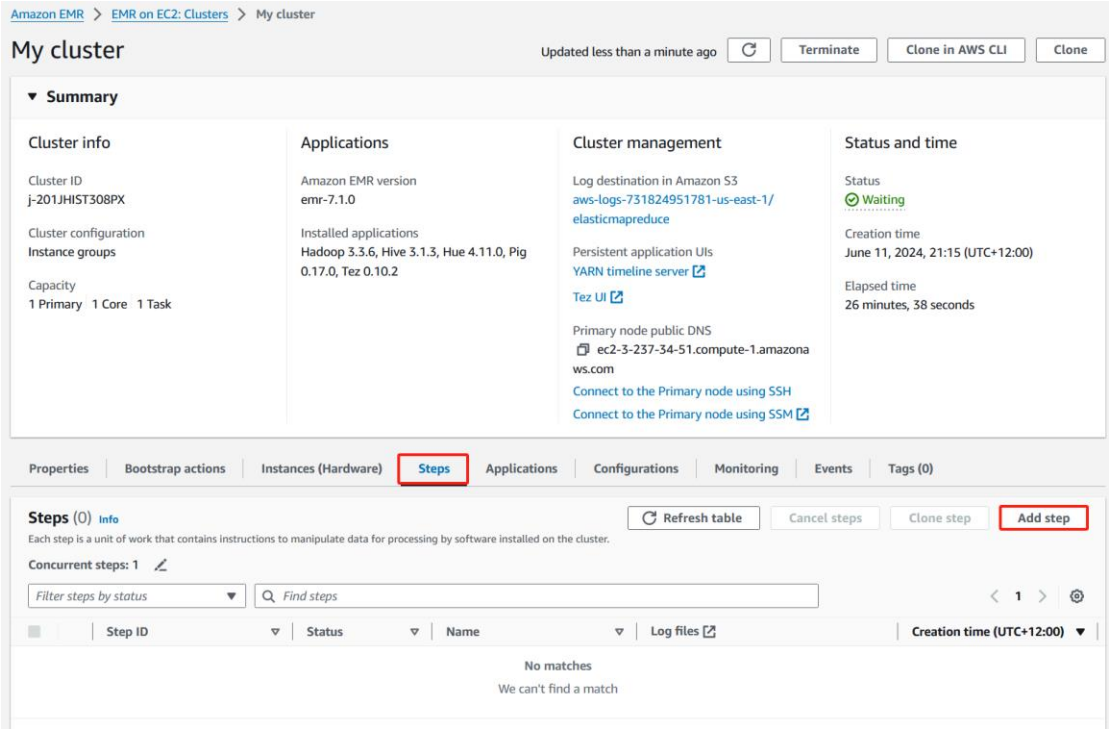


Then, create two folders under the bucket: input and jar. Next, upload the *input.txt* to the input folder and *wordcount.jar* to the jar folder.

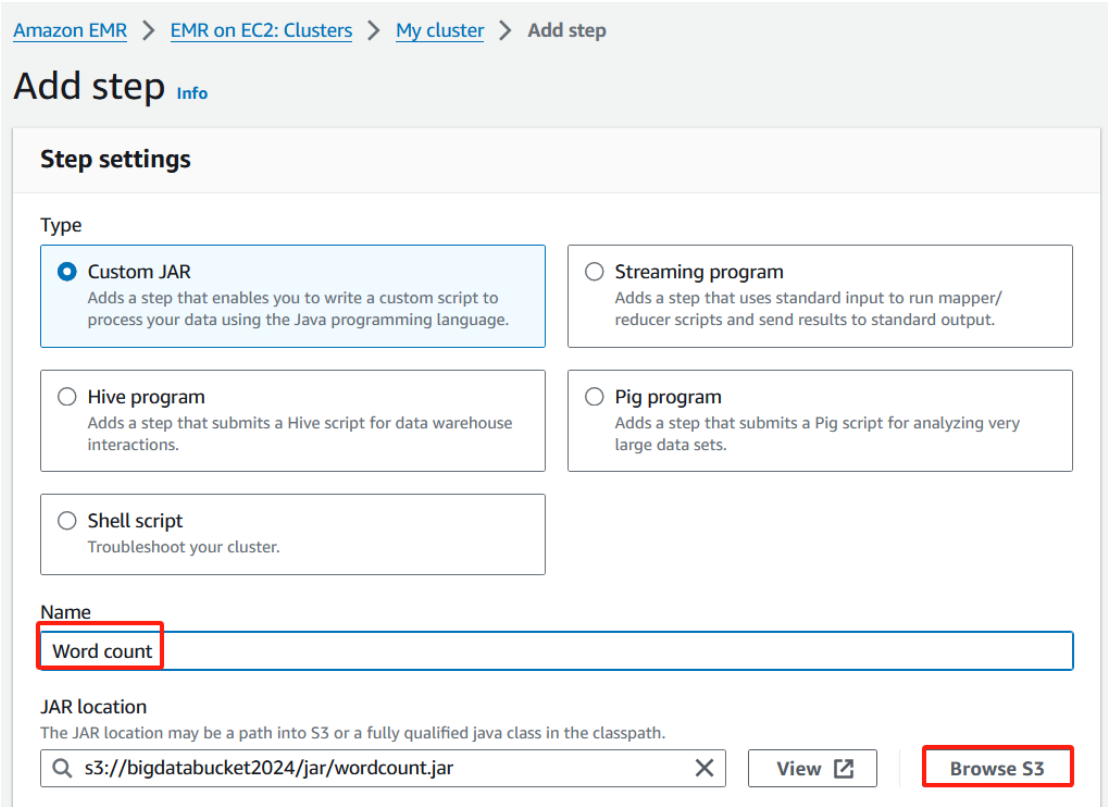


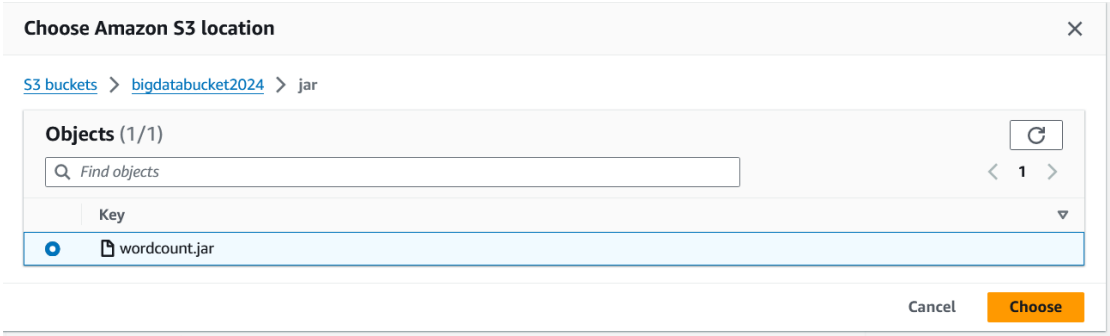
Task 2.3 Execute the Java program using Map-reduce

Go back to the EMR service, select your cluster, and click “add step” under the Steps tab.



Choose Custom JAR as the step type, and give a name for this step, e.g., wordcount. Specify the jar location by selecting it from your S3 directory.



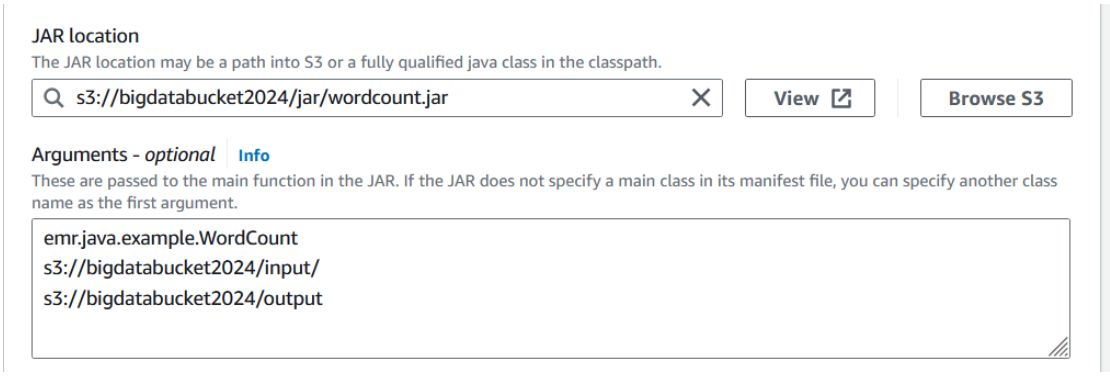


The arguments are very important. In the first line, place the main function of the program, which is *emr.java.example.WordCount*. Place the input and output folder to the second line and third line, respectively. *Please make sure that the "output" folder does NOT exist*!

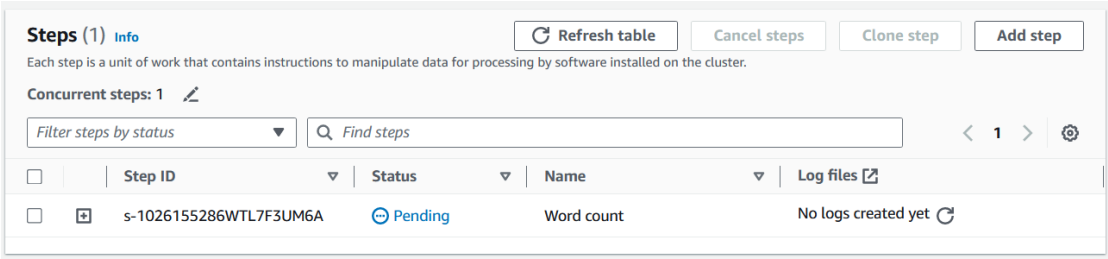
In this example, we use the following argument:

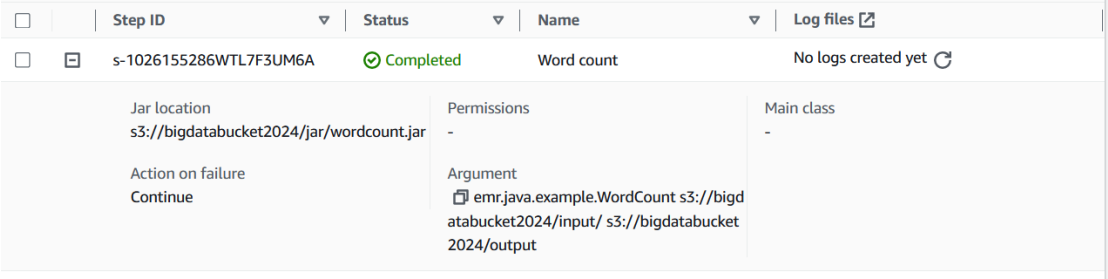
emr.java.example.WordCount
s3://bigdatabucket2024/input/
s3://bigdatabucket2024/output

** replace the input and output augment with the folders that you have created.*

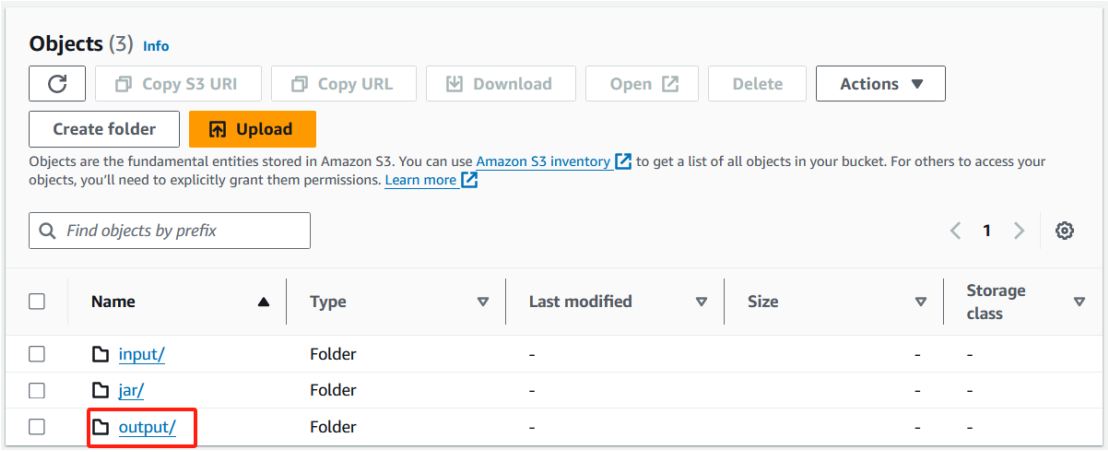


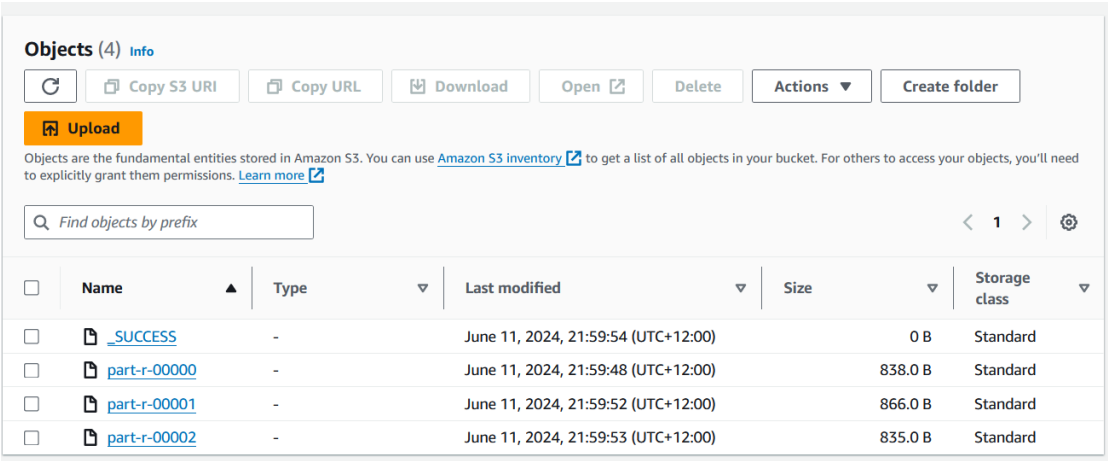
Click Add step, and you will see the status of the step *pending*. Give it around 30 seconds and refresh it. The status will be changed to *completed*.



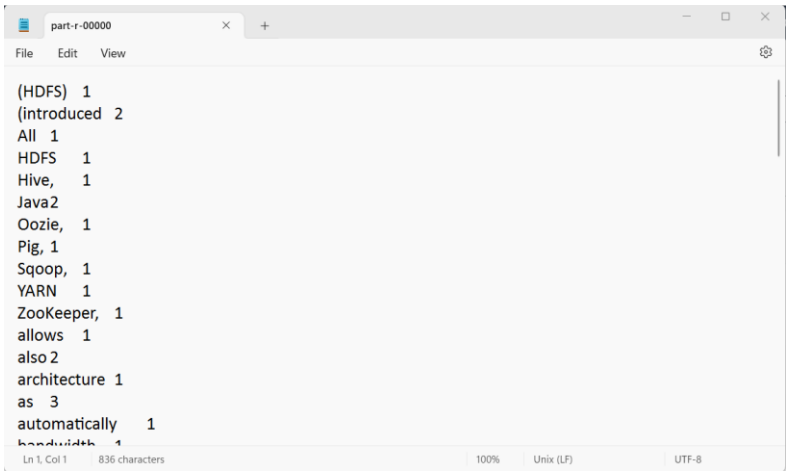


When you return to your S3, you should be able to find the output folder. Open the output folder and explore the file's content (result).



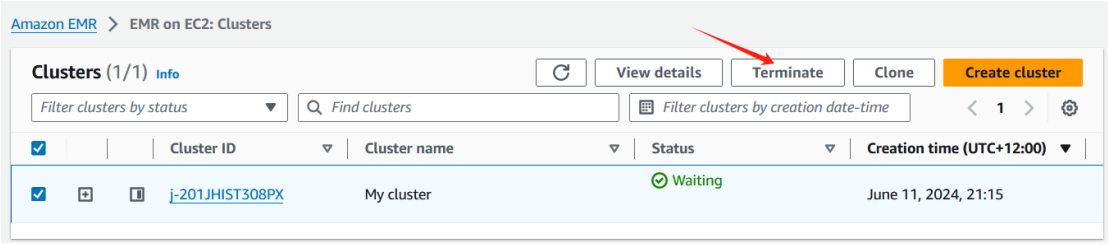


Check out the text of the file – part-r00000. If you create multiple nodes, you will have multiple outputs from different nodes. Here is the output from the text file:



Task 2.4 Terminate your cluster

If you don't want any extra costs incurred, never forget to terminate your cluster after using it!



In the end, please go back to the Modules and click End Lab.