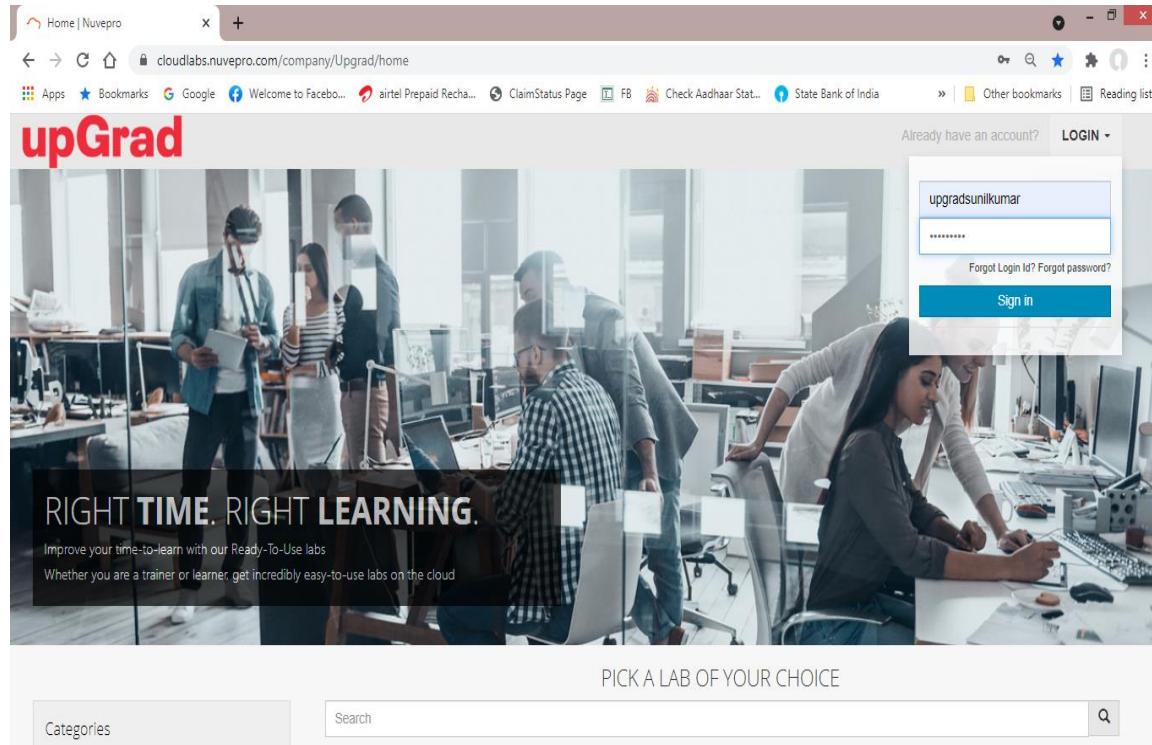


Hive Case Study:

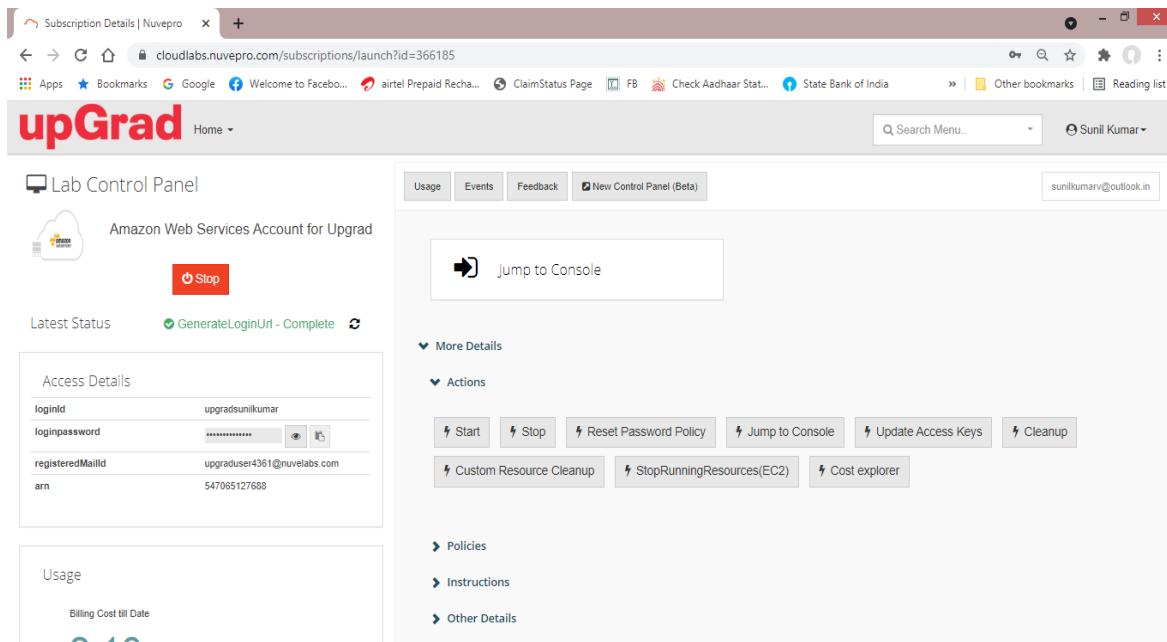
Logged into Amazon Web Services using the following link

<https://cloudlabs.nuvepro.com/company/Upgrad/home>



The screenshot shows a web browser window with the title "Home | Nuvepro". The address bar contains the URL "cloudlabs.nuvepro.com/company/Upgrad/home". The page itself is for "upGrad" and features a large background image of people working in an office. On the right side, there is a login form with fields for "upgradsunilkumar" and "*****", and a "Sign in" button. Below the login form, there are links for "Forgot Login Id? Forgot password?". At the bottom of the page, there is a section titled "PICK A LAB OF YOUR CHOICE" with a search bar and a "Categories" dropdown.

After Login: Clicked on Jump to console



The screenshot shows the "Lab Control Panel" for an "Amazon Web Services Account for Upgrad". The top navigation bar includes "Subscription Details | Nuvepro" and the URL "cloudlabs.nuvepro.com/subscriptions/launch?id=366185". The main interface has several sections: "Lab Control Panel", "Amazon Web Services Account for Upgrad", "Stop" button, "Latest Status" (with "GenerateLoginUrl - Complete" status), "Access Details" (showing loginId: upgradsunilkumar and loginpassword: *****), "Actions" (with buttons for Start, Stop, Reset Password Policy, Jump to Console, Update Access Keys, Custom Resource Cleanup, StopRunningResources(EC2), and Cost explorer), "More Details", "Policies", "Instructions", and "Other Details". The URL in the address bar is https://cloudlabs.nuvepro.com/subscriptions/launch?id=366185.

We have entered the Aws Console, First we will go to the EC2 – Elastic Compute cloud service to generate new Key Pair

The screenshot shows the AWS Management Console homepage. The top navigation bar includes links for Subscription Details, AWS Management Console, and various other services like Google, Facebook, and State Bank of India. A search bar at the top right allows users to search for services and products. The main content area features a large title 'AWS Management Console' and a sidebar titled 'AWS services' which lists recently visited services (EC2, S3, EMR, IAM) and all services. Below this are sections for 'Build a solution', 'Launch a virtual machine', 'Build a web app', 'Build using virtual servers', and 'Stay connected to your AWS resources on-the-go'. The bottom of the page includes a footer with links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences, along with icons for various Microsoft and AWS services.

Click on the Key Pairs to generate new key Pair

Subscription Details | Nuvepro x Dashboard | EC2 Management C +

console.aws.amazon.com/ec2/v2/home?region=us-east-1#Home:

Apps Bookmarks Google Welcome to Facebo... airtel Prepaid Recha... ClaimStatus Page FB Check Aadhaar Stat... State Bank of India Other bookmarks Reading list

Services ▾

New EC2 Experience Tell us what you think

EC2 Dashboard

Events

Tags

Limits

Instances

Instances New

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances New

Dedicated Hosts

Scheduled Instances

Capacity Reservations

Images

Search for services, features, marketplace products, and docs [Alt+S]

Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:

Instances (running)	0	Dedicated Hosts	0
Elastic IPs	0	Instances	1
Key pairs	3	Load balancers	0 API Error
Placement groups	0	Security groups	4
Snapshots	0	Volumes	1

Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server. Learn more

Account attributes

Supported platforms VPC Default VPC vpc-fff88f82 Settings EBS encryption Zones EC2 Serial Console Default credit specification Console experiments

Explore AWS Run Apache Spark on EMR

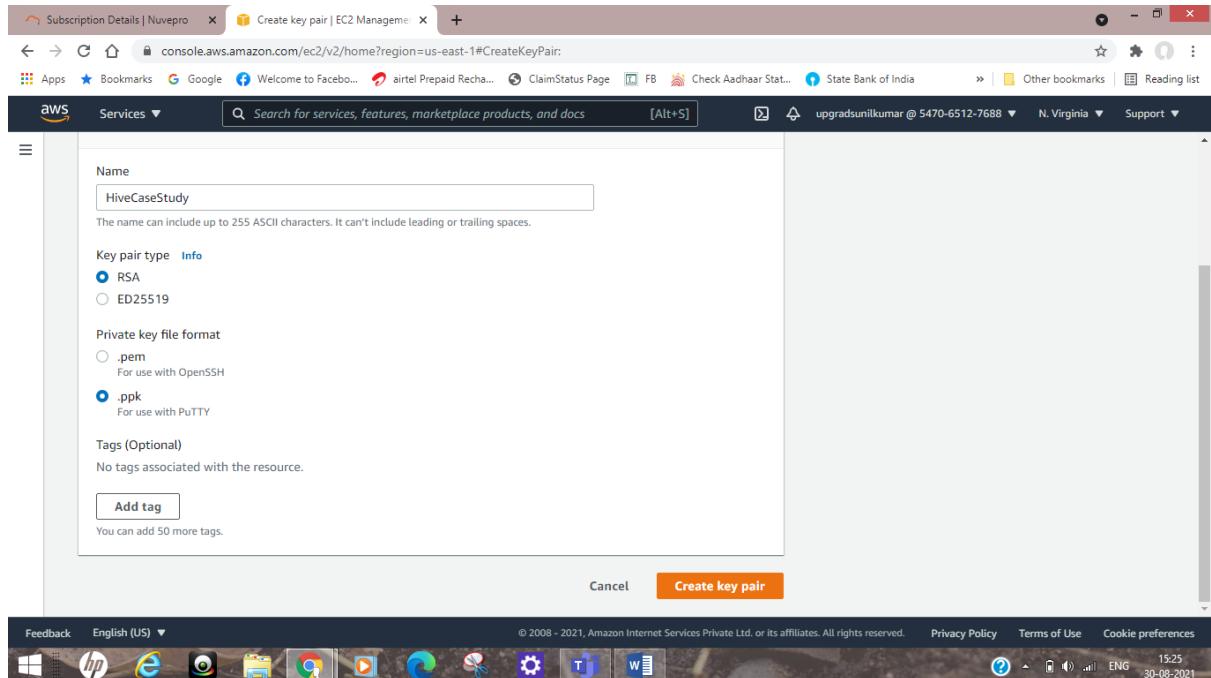
https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#Home:

© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

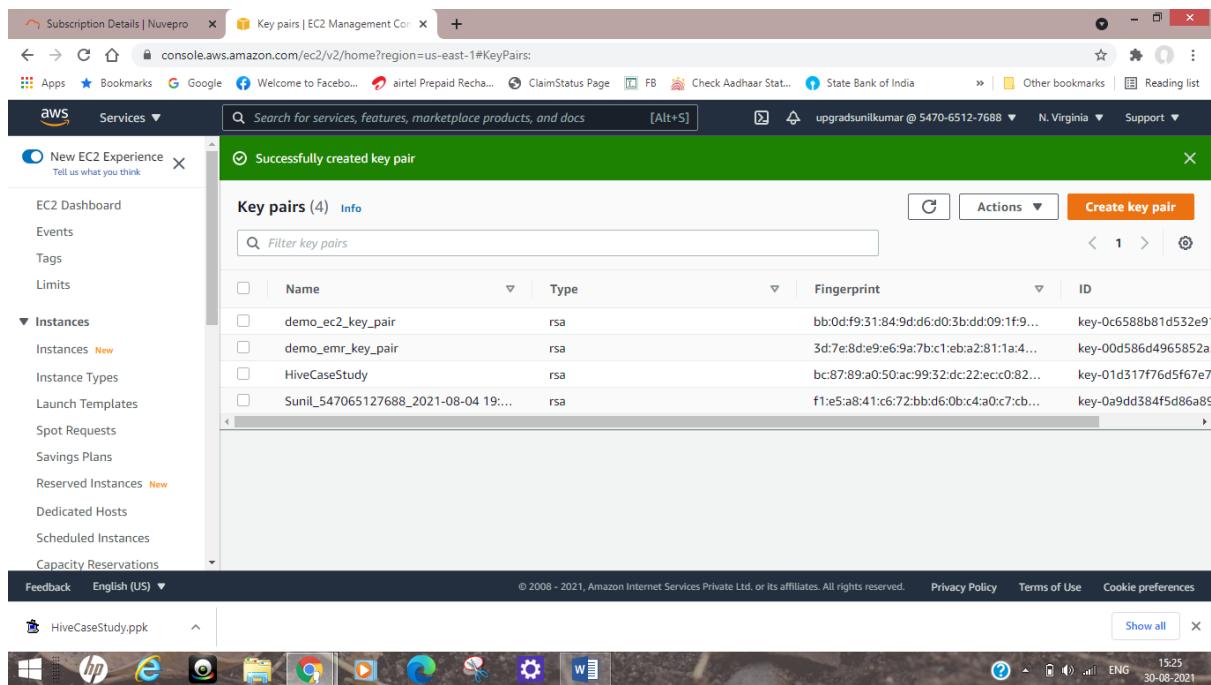
Privacy Policy Terms of Use Cookie preferences

15:24 30-08-2021

Creating Key Pair with ppk file format to use directly with putty



HiveCaseStudy- Key Pair successfully created and downloaded. It can be used to connect with putty.



S3 BUCKET

Click on services and enter S3 – Simple Storage Service

The screenshot shows the AWS S3 Management Console. On the left, there's a sidebar with 'Amazon S3' selected under 'Services'. The main area has a 'Search for services, features, marketplace products, and docs' bar. Below it is an 'Account snapshot' section with metrics like Total storage (9.4 MB), Object count (488), and Avg. object size (19.7 KB). A note says you can enable advanced metrics in the 'default-account-dashboard' configuration. Below this is a 'Buckets (2) info' section with a table:

Name	AWS Region	Access	Creation date
aws-logs-547065127688-us-east-1	US East (N. Virginia) us-east-1	Objects can be public	August 19, 2021, 14:36:08 (UTC+05:30)
upgradsunilbucket	US East (N. Virginia) us-east-1	Objects can be public	August 8, 2021, 22:10:42 (UTC+05:30)

At the bottom, there are links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences.

To Store the data in S3 – Click on “Create Bucket”, given Global unique name to the bucket

The screenshot shows the 'Create bucket' page in the AWS S3 Management Console. The URL is s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1. The page has a 'General configuration' section with fields for 'Bucket name' (set to 'SunilHiveCaseStudy') and 'AWS Region' (set to 'US East (N. Virginia) us-east-1'). There's also a 'Choose bucket' button for copying settings from an existing bucket. At the bottom, there are links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences.

Keeping the Default settings and allowed the public access to the bucket.

The screenshot shows the AWS S3 Bucket creation wizard. In the 'Default encryption' section, the 'Server-side encryption' dropdown is set to 'Disable'. Below it, there's a note: 'After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.' At the bottom right are 'Cancel' and 'Create bucket' buttons. The browser status bar at the bottom indicates the URL is s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1.

Successfully created the sunilhivecasestudy bucket. Now it's ready for data upload

The screenshot shows the AWS S3 Management Console. On the left, the 'Amazon S3' sidebar lists 'Buckets', 'Access Points', 'Object Lambda Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens', 'Dashboards', 'AWS Organizations settings', and 'Feature spotlight'. The main area displays bucket statistics: Total storage 9.4 MB, Object count 488, Avg. object size 19.7 KB. A note says 'You can enable advanced metrics in the "default-account-dashboard" configuration.' Below this is a 'Buckets (3) Info' table:

Name	AWS Region	Access	Creation date
aws-logs-547065127688-us-east-1	US East (N. Virginia) us-east-1	Objects can be public	August 19, 2021, 14:36:08 (UTC+05:30)
sunilhivecasestudy	US East (N. Virginia) us-east-1	Objects can be public	August 30, 2021, 15:29:29 (UTC+05:30)
upgradsunilbucket	US East (N. Virginia) us-east-1	Objects can be public	August 8, 2021, 22:10:42 (UTC+05:30)

The browser status bar at the bottom indicates the URL is s3.console.aws.amazon.com/s3/home?region=us-east-1.

Uploading the 2019 October and 2019 November csv files to S3 bucket

The screenshot shows the AWS S3 Management Console with the 'Upload' page. A large text input field at the top says 'Drag and drop files and folders you want to upload here, or choose Add files, or Add folders.' Below it, a table lists two files: '2019-Nov.csv' and '2019-Oct.csv'. Both files are of type 'application/vnd.ms-excel' and have sizes of 520.6 MB and 460.2 MB respectively. There are 'Remove', 'Add files', and 'Add folder' buttons above the table. The status bar at the bottom indicates the files are being uploaded.

Successfully Uploaded the Files and now ready to move to hdfs.

The screenshot shows the AWS S3 Management Console with the 'sunilhivecasestudy' bucket selected. The 'Objects' tab is active, showing a list of two objects: '2019-Nov.csv' and '2019-Oct.csv'. Both files are of type 'csv' and were last modified on August 30, 2021. The storage class is set to 'Standard'. There are buttons for 'Actions', 'Create folder', and 'Upload' at the top of the object list. The status bar at the bottom indicates the files are stored in the 'Standard' storage class.

EMR CLUSTER CREATION:

Click on AWS Services and select EMR – Elastic Map Reduce

The screenshot shows the AWS EMR - AWS Console interface. On the left, there's a sidebar with navigation links for Amazon EMR, EMR Studio, EMR on EC2 (selected), Clusters, Notebooks, Git repositories, Security configurations, Block public access, VPC subnets, Events, EMR on EKS, and Virtual clusters. The main content area has tabs for 'Create cluster', 'View details', 'Clone', and 'Terminate'. A table lists one cluster: 'demo_emr_cluster' (ID: JS4KJP9B1Z91J, Status: Terminated, User request, Creation time: 2021-08-19 14:36 (UTC+5:30), Elapsed time: 5 hours, 27 minutes, Normalized instance hour: 48). Below the table is a search bar with 'Filter: All clusters' and a 'Filter clusters...' dropdown. At the bottom, there are links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences.

Click on “Create cluster” button to create the EMR cluster and select advances options.

Selected emr 5.29.0 release for this case study

The screenshot shows the 'Create Cluster - Advanced Options' page. On the left, a sidebar lists steps: Step 1: Software and Steps (selected), Step 2: Hardware, Step 3: General Cluster Settings, and Step 4: Security. The main area is titled 'Software Configuration' and shows a dropdown for 'Release' set to 'emr-5.29.0'. Underneath, there are two columns of checkboxes for various software components: Hadoop 2.8.5, Zeppelin 0.8.2, Livy 0.6.0; JupyterHub 1.0.0, Tez 0.9.2, Flink 1.9.1; Ganglia 3.7.2, HBase 1.4.10, ZooKeeper 3.4.14; Hive 2.3.6, Presto 0.227, Mahout 0.13.0; MXNet 1.5.1, Sqoop 1.4.7, Oozie 5.1.0; Hue 4.4.0, Phoenix 4.14.3, TensorFlow 1.14.0; Spark 2.4.4, HCatalog 2.3.6, TensorFlow 1.14.0. Below these are sections for 'Multiple master nodes (optional)' and 'AWS Glue Data Catalog settings (optional)'. At the bottom, there's a link 'Edit software settings' and standard browser footer links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences.

We are using 2 node cluster with m4.large for Master node and M4.large for Core node.

Node type	Instance type	Instance count	Purchasing option
Master	m4.large	1 Instances	<input checked="" type="radio"/> On-demand <input type="radio"/> Spot Use on-demand as max price
Core	m4.large	1 Instances	<input checked="" type="radio"/> On-demand <input type="radio"/> Spot Use on-demand as max price
Task	m5.xlarge	0 Instances	<input checked="" type="radio"/> On-demand <input type="radio"/> Spot Use on-demand as max price

We will be using the HiveCaseStudy Key Pair which we created for connecting the cluster.

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps

Step 2: Hardware

Step 3: General Cluster Settings

Step 4: Security

Security Options

EC2 key pair [HiveCaseStudy](#)

Cluster visible to all IAM users in account [?](#)

Permissions [?](#)

Default Custom

Use default IAM roles. If roles are not present, they will be automatically created for you with managed policies for automatic policy updates.

EMR role [EMR_DefaultRole](#) Use EMR_DefaultRole_V2 [?](#)

EC2 instance profile [EMR_EC2_DefaultRole](#) [?](#)

Auto Scaling role [EMR_AutoScaling_DefaultRole](#) [?](#)

▼ Security Configuration

Security configuration [None](#)

▼ EC2 security groups

After selecting all the specifications. Click on Create Cluster

EC2 instance profile [EMR_EC2_DefaultRole](#) ⓘ
Auto Scaling role [EMR_AutoScaling_DefaultRole](#) ⓘ
▼ Security Configuration
Security configuration [None](#) ⓘ
▼ EC2 security groups
An EC2 security group acts as a virtual firewall for your cluster nodes to control inbound and outbound traffic. There are two types of security groups you can configure, [EMR managed security groups](#) ⓘ and [additional security groups](#) ⓘ. EMR will automatically update ⓘ the rules in the EMR managed security groups in order to launch a cluster. [Learn more](#) ⓘ

Type	EMR managed security groups	Additional security groups
Master	Default: sg-06e48dae7153c76ff (ElasticMapReduce-)	No security groups selected
Core & Task	Default: sg-0abda5d07139ff21a (ElasticMapReduce-)	No security groups selected

[Create a security group](#) ⓘ
Cancel Previous **Create cluster**

EMR Cluster Creation has started.

Subscription Details | Nuvepro x S3 Management Console x EMR - AWS Console x +
azn.com/elasticmapreduce/home?region=us-east-1#cluster-detailsj-65RPRB5QF19Z
Welcome to Facebook... airtel Prepaid Recharge... ClaimStatus Page FB Check Aadhaar Status... State Bank of India Other bookmarks Reading list
Feedback English (US) Show all 15:55 upgradsunilkumar @ 5470-6512-7688 N. Virginia Support
Amazon EMR Services Search for services, features, marketplace products, and docs [Alt+S] upgradsunilkumar @ 5470-6512-7688 N. Virginia Support
Clusters Clusters
Notebooks Notebooks
Git repositories Git repositories
Security configurations Security configurations
Block public access Block public access
VPC subnets VPC subnets
Events Events
EMR on EKS EMR on EKS
Virtual clusters Virtual clusters
Help Help
Clone Terminate AWS CLI export
Cluster: HiveCaseStudy Starting Configuring cluster software
Summary Application user interfaces Monitoring Hardware Configurations Events Steps Bootstrap actions
Summary
ID: j-65RPRB5QF19Z
Creation date: 2021-08-30 15:56 (UTC+5:30)
Elapsed time: 5 minutes
After last step completes: Cluster waits
Termination protection: On Change
Tags: -- View All / Edit
Master public DNS: ec2-3-236-37-163.compute-1.amazonaws.com Connect to the Master Node Using SSH
Configuration details
Release label: emr-5.29.0
Hadoop distribution: Amazon 2.8.5
Applications: Hive 2.3.6, Pig 0.17.0, Hue 4.4.0
Log URI: s3://aws-logs-547065127688-us-east-
Feedback English (US) Show all 16:02 upgradsunilkumar @ 5470-6512-7688 N. Virginia Support
15:55 30-08-2021

Cluster is ready and status has changed from starting to “Waiting”

The screenshot shows the AWS EMR console interface. On the left, there's a sidebar with navigation links for Amazon EMR, EMR Studio, EMR on EC2 (with 'Clusters' selected), Clusters, Notebooks, Git repositories, Security configurations, Block public access, VPC subnets, Events, EMR on EKS, Virtual clusters, and Help. The main content area displays a cluster named 'HiveCaseStudy' in a 'Waiting' state. The 'Summary' tab is selected, showing details like ID: j-65RPRB5QF19Z, Creation date: 2021-08-30 15:56 (UTC+5:30), Elapsed time: 13 minutes, and After last step completes: Cluster waits. It also shows Termination protection: On Change, Tags: -- View All / Edit, Master public DNS: ec2-3-236-37-163.compute-1.amazonaws.com, and a link to Connect to the Master Node Using SSH. Below the summary, there are tabs for Application user interfaces, Monitoring, Hardware, Configurations, Events, Steps, and Bootstrap actions. At the bottom of the page, there are links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences.

Master and Core nodes are also in running status.

The screenshot shows the AWS EMR console interface, similar to the previous one but focusing on the 'Network and hardware' section. The sidebar and cluster details are identical. In the main content area, under 'Network and hardware', it shows the Availability zone: us-east-1c, Subnet ID: subnet-20540e46, Master: Running 1 m4.large, Core: Running 1 m4.large, Task: --, and Cluster scaling: Not enabled. Below this, the 'Security and access' section lists Key name: HiveCaseStudy, EC2 instance profile: EMR_EC2_DefaultRole, EMR role: EMR_DefaultRole, Auto Scaling role: EMR_AutoScaling_DefaultRole, and security groups for Master and Core nodes. The taskbar at the bottom shows various application icons and the system tray indicates the date and time as 30-08-2021.

We can check the status of the nodes in the hardware section.

Master and Core node both are running.

Amazon EMR

EMR Studio

EMR on EC2

Clusters

Notebooks

Git repositories

Security configurations

Block public access

VPC subnets

Events

EMR on EKS

Virtual clusters

Help

What's new

Subscription Details | Nuvepro | sunilhivecasestudy - S3 | EMR - AWS Console | View web interfaces home | EC2 Management Console | Instances | EC2 Manager | +

console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#cluster-detailsj-65RPRB5QF92

Search for services, features, marketplace products, and docs [Alt+S]

Clone Terminate AWS CLI export

Cluster: HiveCaseStudy Waiting Cluster ready after last step completed.

Summary Application user interfaces Monitoring Hardware Configurations Events Steps Bootstrap actions

Add task instance group

Instance groups

Filter: Filter instance groups ... 2 instance groups (all loaded) C

ID	Status	Node type & name	Instance type	Instance count
ig-W60YAZ118OB6	Running	CORE Core - 2	m4.large 2 vCore, 8 GiB memory, EBS only storage EBS Storage: 32 GiB	1 Instances Resize
ig-2UHSEYGCXAQE1	Running	MASTER Master - 1	m4.large 2 vCore, 8 GiB memory, EBS only storage EBS Storage: 32 GiB	1 Instances

Cluster Scaling Policy

No scaling enabled

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences 21:48 30-08-2021

We have to edit the Security Group of Master Node

New EC2 Experience Tell us what you think

EC2 Dashboard

Events

Tags

Limits

Instances

Instances New

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances New

Dedicated Hosts

Scheduled Instances

Capacity Reservations

Subscription Details | Nuvepro | S3 Management Console | EMR - AWS Console | EC2 Management Console | Instances | EC2 Manager | +

console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroups:search=sg-06e48dae7153c76ff

Search for services, features, marketplace products, and docs [Alt+S]

Actions Create security group

Security Groups (2) Info

Filter security groups

search: sg-06e48dae7153c76ff Clear filters

Name	Security group ID	Security group name	VPC ID	Description
-	sg-06e48dae7153c76ff	ElasticMapReduce-master	vpc-fff88f82	Master group for Elastic...
-	sg-dabda5d07139ff21a	ElasticMapReduce-slave	vpc-fff88f82	Slave group for Elastic...

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences Show all 16:15 30-08-2021

Then save the SSH rule to anywhere in the inbound rules

The screenshot shows the AWS Management Console with multiple tabs open. The active tab is 'ModifyInboundSecurityGroupRules' for a specific security group. The interface lists five rules with their respective port ranges and CIDR ranges. Rule 1: Custom TCP (port 8443) allowing 207.171.167.26/32. Rule 2: Custom TCP (port 8443) allowing 207.171.172.6/32. Rule 3: All TCP (port 0 - 65535) allowing 0.0.0.0/0. Rule 4: Custom TCP (port 22) allowing 0.0.0.0/0. Rule 5: SSH (port 22) allowing 0.0.0.0/0. Each rule has a 'Delete' button next to it.

CONNECTING TO MASTER NODE:

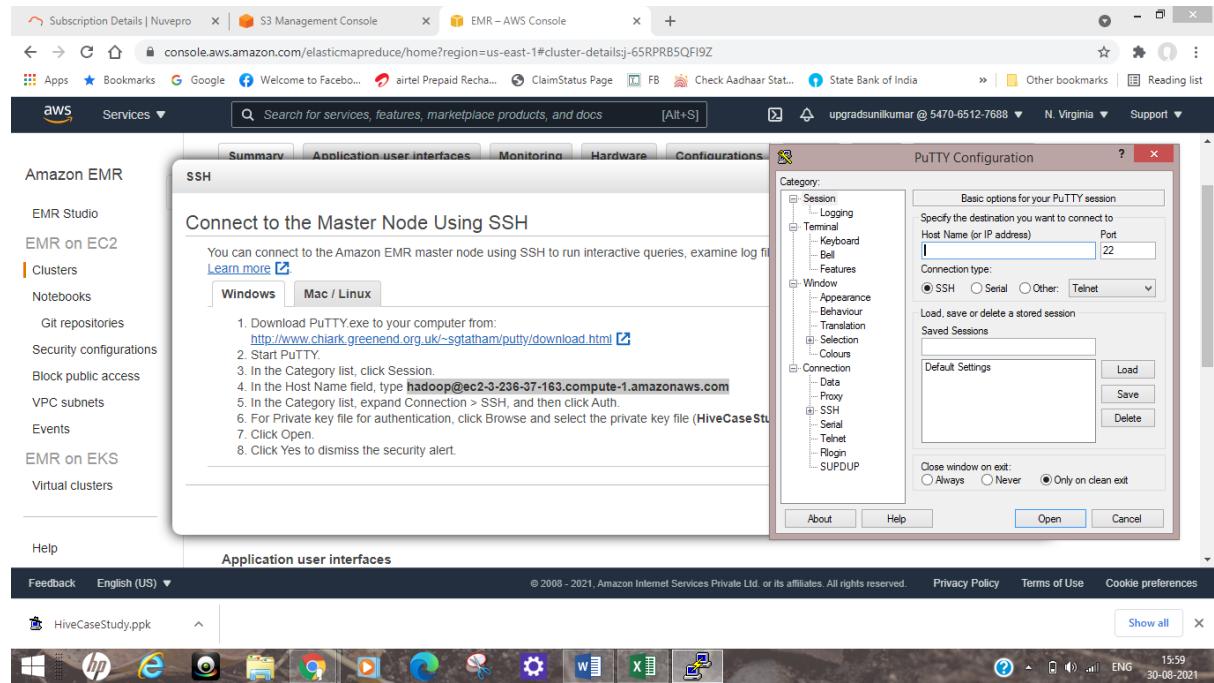
Host: hadoop@ec2-3-236-37-163.compute-1.amazonaws.com

The screenshot shows a 'SSH' connection dialog box. It contains instructions for connecting to the master node using PuTTY. It includes tabs for 'Windows' and 'Mac / Linux'. A list of 8 steps is provided for Windows users:

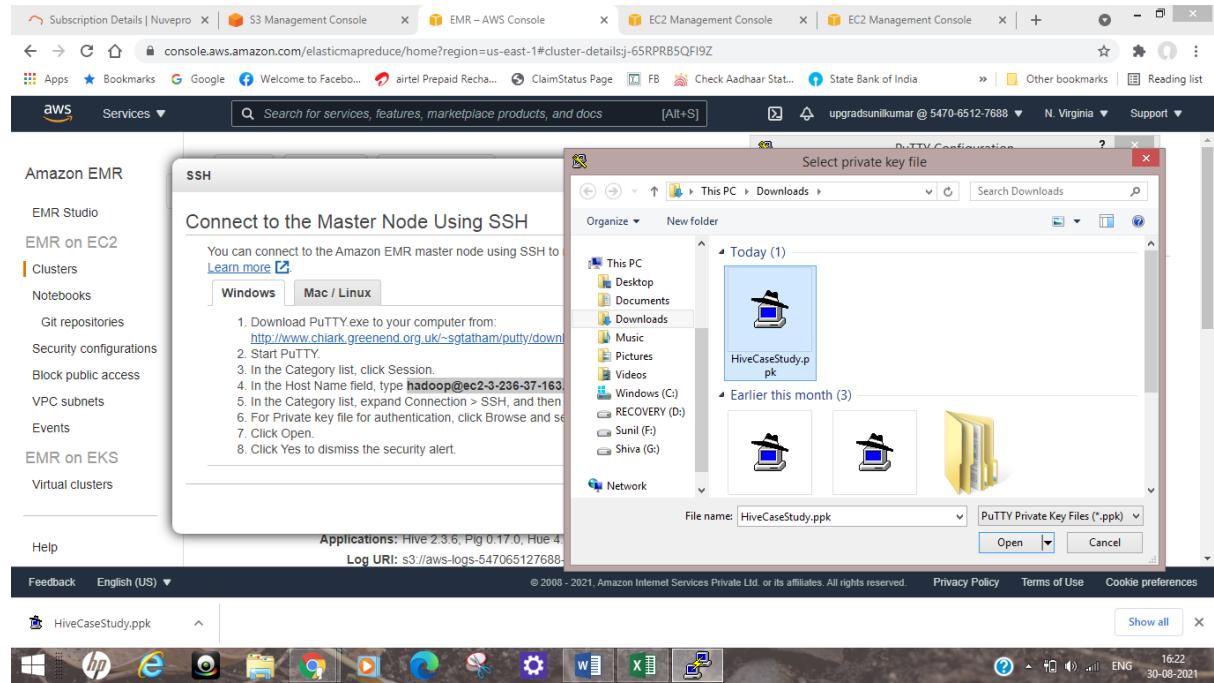
1. Download PuTTY.exe to your computer from: <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>
2. Start PuTTY.
3. In the Category list, click Session.
4. In the Host Name field, type **hadoop@ec2-3-236-37-163.compute-1.amazonaws.com**.
5. In the Category list, expand Connection > SSH, and then click Auth.
6. For Private key file for authentication, click Browse and select the private key file (**HiveCaseStudy.ppk**) used to launch the cluster.
7. Click Open.
8. Click Yes to dismiss the security alert.

A 'Close' button is located at the bottom right of the dialog box.

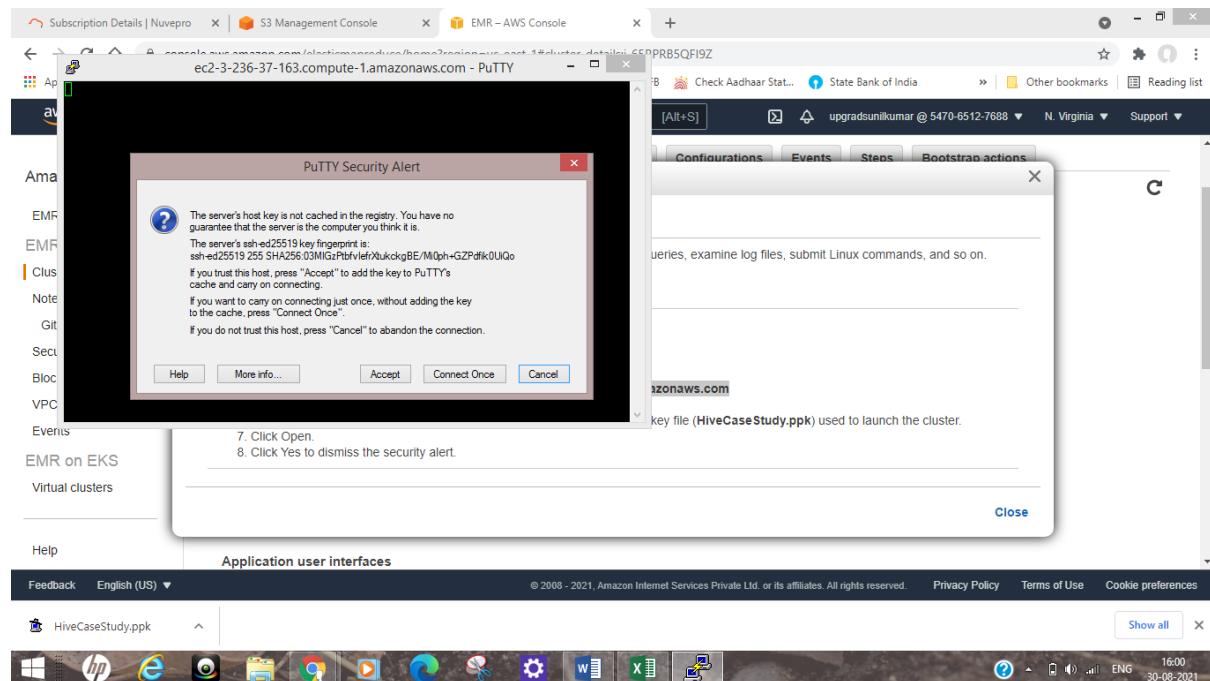
Open the putty and enter the Host Name and click on SSH→ Auth



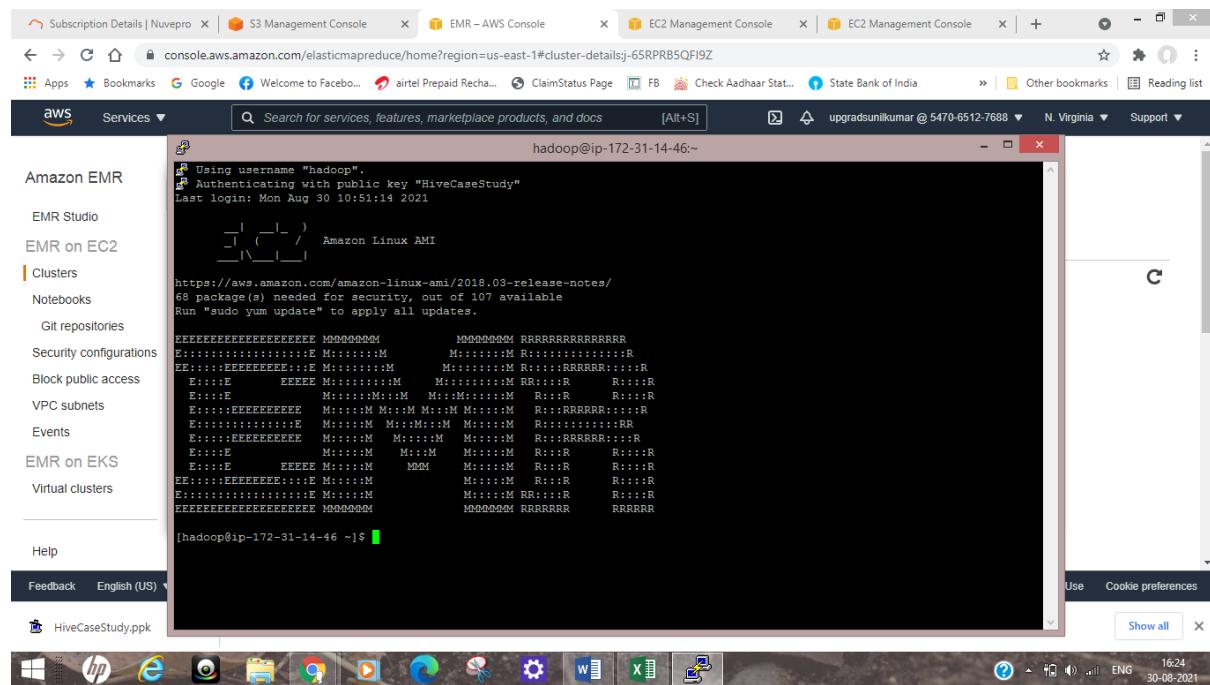
Browse and select the private key HiveCaseStudy.ppk file



Click on Open and accept the connection

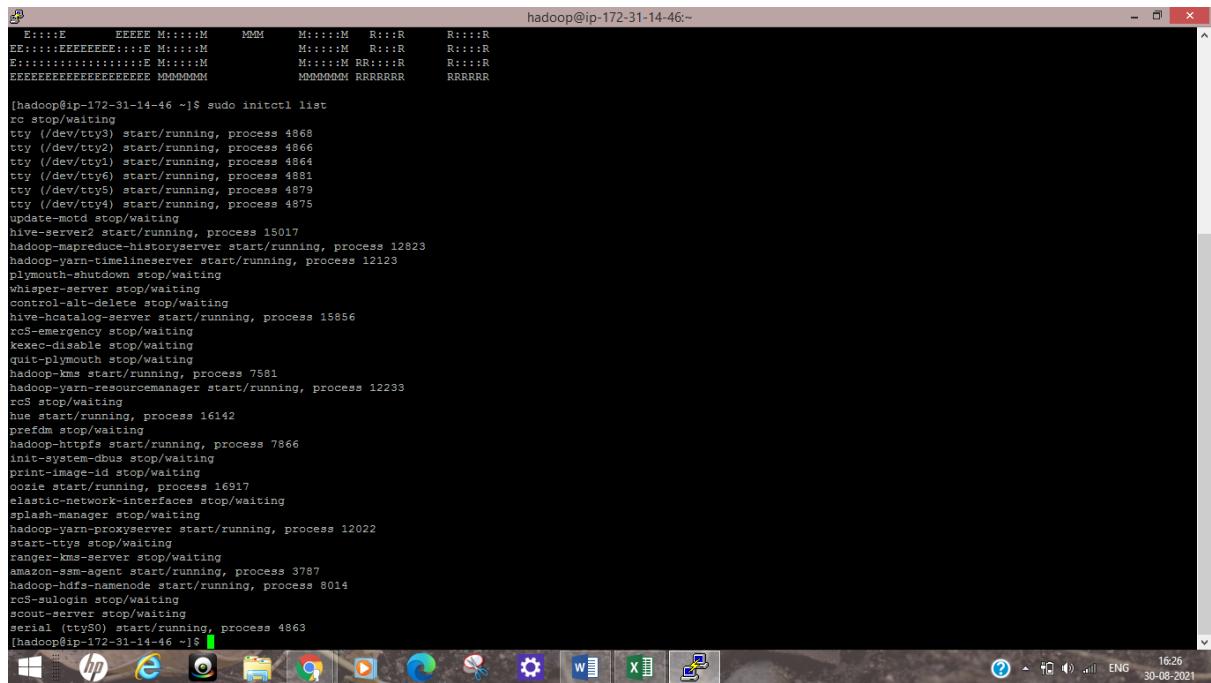


EMR – Elastic Map Reduce CLI is launched successfully.



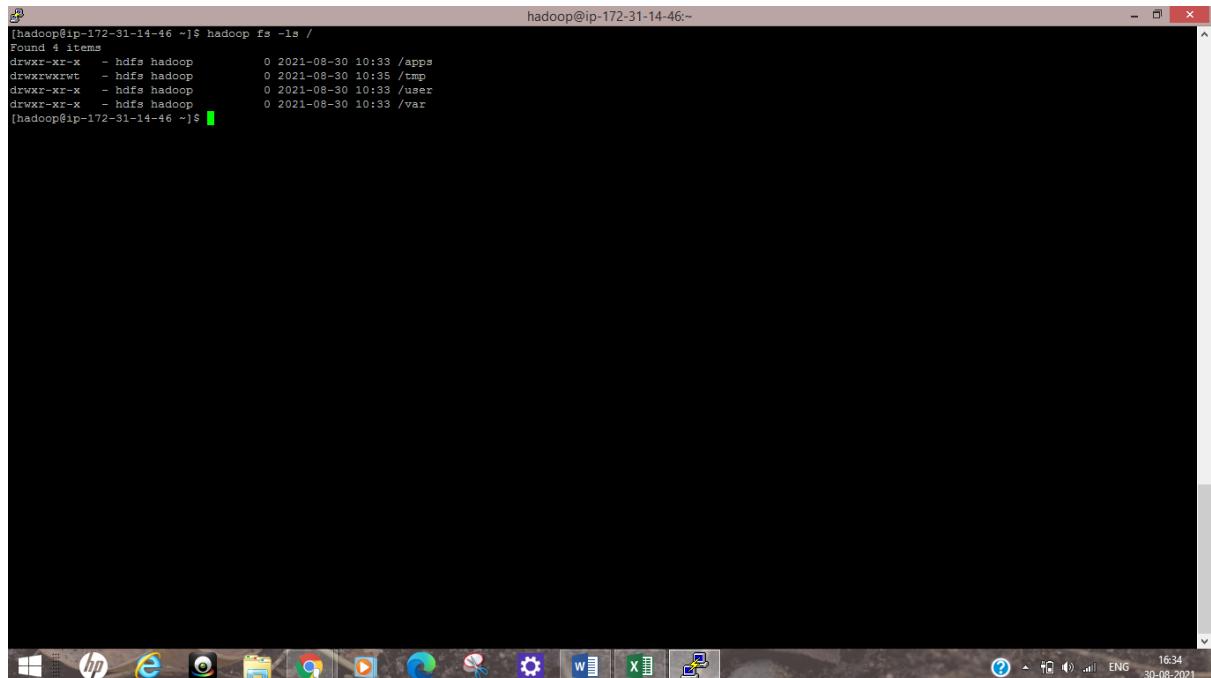
Enter “sudo initctl list” to check the services running in the EMR.

Hive services are running.



```
[hadoop@ip-172-31-14-46 ~]$ sudo initctl list
rc_stop/waiting
tty (/dev/tty3) start/running, process 4868
tty (/dev/tty2) start/running, process 4866
tty (/dev/tty1) start/running, process 4864
tty (/dev/tty6) start/running, process 4881
tty (/dev/tty5) start/running, process 4879
tty (/dev/tty4) start/running, process 4875
update-moto stop/waiting
hive-server2 start/running, process 15017
hadoop-mapreduce-historyserver start/running, process 12823
hadoop-yarn-timelinesserver start/running, process 12123
plymouth-shutdown stop/waiting
whisper-server stop/waiting
control-alt-delete stop/waiting
hive-hcatalog-server start/running, process 15856
rcS-emergency stop/waiting
kexec-disable stop/waiting
quit-plymouth stop/waiting
hadoop-kms start/running, process 7581
hadoop-yarn-resourcemanager start/running, process 12233
rcS stop/waiting
hue start/running, process 16142
prefdm stop/waiting
hadoop-https start/running, process 7866
init-system-dbus stop/waiting
print-image-id stop/waiting
oozie start/running, process 16917
elastic-network-interfaces stop/waiting
splash-manager stop/waiting
hadoop-yarn-proxyserver start/running, process 12022
start-ttys stop/waiting
ranger-kms-server stop/waiting
amazon-ssm-agent start/running, process 3787
hadoop-hdfs-namenode start/running, process 8014
rcS-sulogin stop/waiting
scout-server stop/waiting
serial (ttyS0) start/running, process 4863
[hadoop@ip-172-31-14-46 ~]$
```

Verifying the Hadoop file system with command “**hadoop fs -ls /**”



```
[hadoop@ip-172-31-14-46 ~]$ hadoop fs -ls /
Found 4 items
drwxr-xr-x  - hdfs hadoop      0 2021-08-30 10:33 /apps
drwxrwxrwt - hdfs hadoop      0 2021-08-30 10:35 /tmp
drwxr-xr-x  - hdfs hadoop      0 2021-08-30 10:33 /user
drwxr-xr-x  - hdfs hadoop      0 2021-08-30 10:33 /var
[hadoop@ip-172-31-14-46 ~]$
```

CREATING A DIRECTORY FOR CASE STUDY:

Creating a new directory under user>hive for Hive case study and to store the data files

hadoop fs -mkdir /user/hive/hivecasestudy

We have also verified the directory using the command

hadoop fs -ls /user/hive/

New directory is successfully created

LOADING DATA into HDFS:

S3 Path for data files:

s3://sunilhivecasestudy/2019-Nov.csv

s3://sunilhivecasestudy/2019-Oct.csv

Copying the data from S3 to HDFS Using distributed copy command:

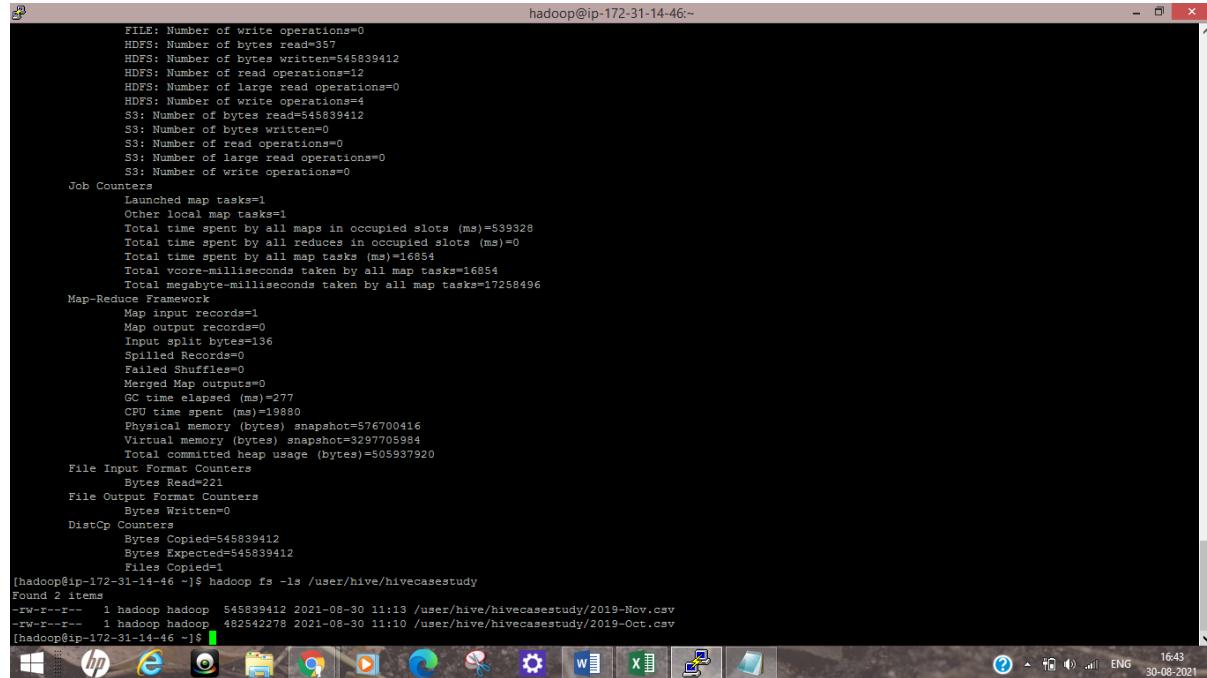
hadoop distcp s3://sunilhivecasestudy/2019-Oct.csv /user/hive/hivecasestudy/2019-Oct.csv

hadoop distcp s3://sunilhivecasestudy/2019-Nov.csv /user/hive/hivecasestudy/2019-Nov.csv

```
hadoop@ip-172-31-14-46:~  
[hadoop@ip-172-31-14-46 ~]$ hadoop distcp s3://sunilhivecasestudy/2019-Nov.csv /user/hive/hivecasestudy/2019-Nov.csv  
21/08/30 11:12:31 INFO tools.DistCp: Input Options: DistCpOptions(atomicCommit=false, syncFolder=false, deleteMissing=false, ignoreFailures=false, overwrite=false, skipCRC=false, blocking=true, numListStatusThreads=0, maxMaps=20, mapBandwidth=100, selConfigurationFile=null, copyStrategy='uniformize', preserveStatus=[], preserveRawXAttr=false, atomicWorkPath=null, logPath=null, sourceFileListing=null, sourcePaths=[s3://sunilhivecasestudy/2019-Nov.csv], targetPath=/user/hive/hivecasestudy/2019-Nov.csv, targetPathExists=false, filtersFile=null)  
21/08/30 11:12:31 INFO client.RMProxy: Connecting to ResourceManager at ip-172-31-14-46.ec2.internal/172.31.14.46:8032  
21/08/30 11:12:35 INFO tools.SimpleCopyListing: Paths (files+dirs) cnt = 1; dirCnt = 0  
21/08/30 11:12:35 INFO tools.SimpleCopyListing: Build file listing completed.  
21/08/30 11:12:35 INFO Configuration.deprecation: io.sort.mb is deprecated. Instead, use mapreduce.task.io.sort.mb  
21/08/30 11:12:35 INFO Configuration.deprecation: io.sort.factor is deprecated. Instead, use mapreduce.task.io.sort.factor  
21/08/30 11:12:35 INFO tools.DistCp: Number of paths in the copy list: 1  
21/08/30 11:12:35 INFO tools.DistCp: Number of paths in the copy list: 1  
21/08/30 11:12:35 INFO client.RMProxy: Connecting to ResourceManager at ip-172-31-14-46.ec2.internal/172.31.14.46:8032  
21/08/30 11:12:35 INFO mapreduce.JobSubmission: number of splits:1  
21/08/30 11:12:35 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1630319660287_0002  
21/08/30 11:12:35 INFO impl.YarnClientImpl: Submitted application application_1630319660287_0002  
21/08/30 11:12:35 INFO mapreduce.Job: The url to track the job: http://ip-172-31-14-46.ec2.internal:20888/proxy/application_1630319660287_0002/  
21/08/30 11:12:35 INFO tools.DistCp: DistCp job-id: job_1630319660287_0002  
21/08/30 11:12:35 INFO mapreduce.Job: Running job: job_1630319660287_0002  
21/08/30 11:12:44 INFO mapreduce.Job: Job job_1630319660287_0002 running in uber mode : false  
21/08/30 11:12:44 INFO mapreduce.Job: map 0% reduce 0%  
21/08/30 11:13:01 INFO mapreduce.Job: map 100% reduce 0%  
21/08/30 11:13:03 INFO mapreduce.Job: Job job_1630319660287_0002 completed successfully  
21/08/30 11:13:03 INFO mapreduce.Job: Counters: 38  
  File System Counters  
    FILE: Number of bytes read=0  
    FILE: Number of bytes written=172487  
    FILE: Number of read operations=0  
    FILE: Number of large read operations=0  
    FILE: Number of write operations=0  
    HDFS: Number of bytes read=357  
    HDFS: Number of bytes written=545839412  
    HDFS: Number of read operations=12  
    HDFS: Number of large read operations=0  
    HDFS: Number of write operations=0  
    S3: Number of bytes read=545839412  
    S3: Number of bytes written=0  
    S3: Number of read operations=0  
    S3: Number of large read operations=0  
    S3: Number of write operations=0  
  Job Counters  
    Launched map tasks=1  
    Other local map tasks=1  
    Total time spent by all maps in occupied slots (ms)=539328  
hadoop@ip-172-31-14-46:~$
```

Both the files are successfully copied into the HDFS System.

hadoop fs -ls /user/hive/hivecasestudy

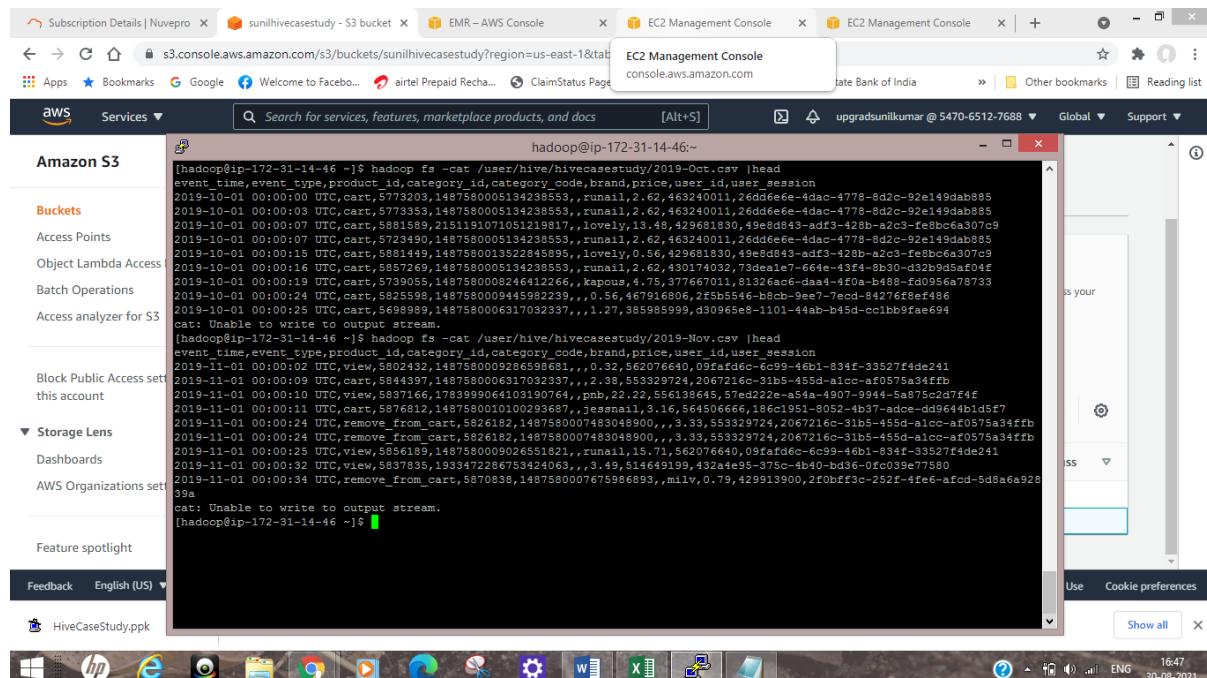


```
hadoop@ip-172-31-14-46:~$ hadoop fs -ls /user/hive/hivecasestudy
Found 2 items
-rw-r--r-- 1 hadoop hadoop 545839412 2021-08-30 11:13 /user/hive/hivecasestudy/2019-Nov.csv
-rw-r--r-- 1 hadoop hadoop 482542278 2021-08-30 11:10 /user/hive/hivecasestudy/2019-Oct.csv
```

Checking the top rows of the copied files in HDFS using

hadoop fs -cat /user/hive/hivecasestudy/2019-Oct.csv | head

hadoop fs -cat /user/hive/hivecasestudy/2019-Nov.csv | head



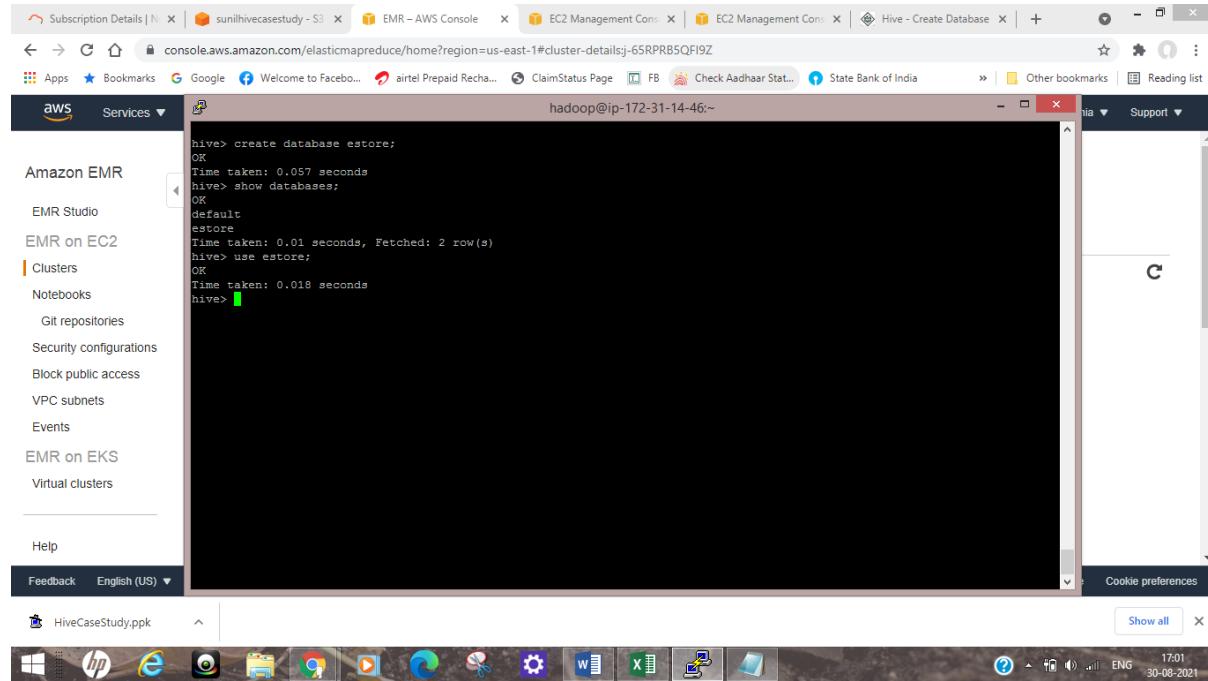
```
[hadoop@ip-172-31-14-46 ~]$ hadoop fs -cat /user/hive/hivecasestudy/2019-Oct.csv | head
event_time,event_type,product_id,category_id,category_code,brand,price,user_id,user_session
2019-10-01 00:00:00 UTC,cart,5773203,1487580005134238553,,runmail,2,62,463240011,2eddf6e6e-adac-4778-8d2c-92e149dab885
2019-10-01 00:00:03 UTC,cart,5773353,1487580005134238553,,runmail,2,62,463240011,2eddf6e6e-adac-4778-8d2c-92e149dab885
2019-10-01 00:00:07 UTC,cart,5881589,2151191071051219817,,lovelys,13,48,429681830,49e8d843-adf3-428b-a2c3-fe8bc6a307c9
2019-10-01 00:00:07 UTC,cart,5723490,1487580005134238553,,runmail,2,62,463240011,2eddf6e6e-adac-4778-8d2c-92e149dab885
2019-10-01 00:00:15 UTC,cart,5881449,148758000513522845895,,lovelys,0,56,429681830,49e8d843-adf3-428b-a2c3-fe8bc6a307c9
2019-10-01 00:00:16 UTC,cart,5857269,1487580005134238553,,runmail,2,62,430174032,73deale7-664e-43f4-9b30-d32995af04f
2019-10-01 00:00:19 UTC,cart,5739055,1487580005246412266,,kaptops,4,75,377667011,81326cc6-3aa4-4f0a-9488-fd0956a78733
2019-10-01 00:00:24 UTC,cart,5825598,1487580009445982239,,0,16,467916806,265b5546-18cb-9ee7-7ecd-84276f8ef486
2019-10-01 00:00:25 UTC,cart,5698989,1487580006317032377,,1,27,385985999,d209395e-1101-44a1-b45d-c01bb59fe694
cat: Unable to write to output stream
[hadoop@ip-172-31-14-46 ~]$ hadoop fs -cat /user/hive/hivecasestudy/2019-Nov.csv | head
event_time,event_type,product_id,category_id,category_code,brand,price,user_id,user_session
2019-11-01 00:00:02 UTC,view,5802432,148758000926598681,,,0,32,562076640,09fafd6c-6c99-46b1-834f-33527f4de241
2019-11-01 00:00:09 UTC,view,5844397,1487580006317032337,,2,18,553329724,2067216c-31b5-455d-a1cc-a0f0575a54ff
2019-11-01 00:00:10 UTC,view,5837166,1783999064103190764,,22,22,556138645,57ed222c-a54a-4907-9944-5a875e22d4f4
2019-11-01 00:00:11 UTC,view,5876812,14875800010100293687,,runmail,3,16,564506666,186c1951-8052-4b37-8dce-d19644b1d5f7
2019-11-01 00:00:24 UTC,remove_from_cart,5825182,1487580007483048900,,3,38,553329724,2067216c-31b5-455d-a1cc-a10575a54ff
2019-11-01 00:00:24 UTC,remove_from_cart,5826182,1487580007483048900,,3,33,553329724,2067216c-31b5-455d-a1cc-a0f0575a54ff
2019-11-01 00:00:25 UTC,view,5856169,148758000926551821,,runmail,15,71,562076640,09fafd6c-6c99-46b1-834f-33527f4de241
2019-11-01 00:00:32 UTC,view,5837835,1933472286753424063,,3,49,514649199,432a4e95-375c-4b40-8d36-0fc039e77380
2019-11-01 00:00:34 UTC,remove_from_cart,5870858,1487580007675986893,,mlnv,0,79,429913900,2f0bbff3c-252f-4fe6-afcd-5d0a6a928
39a
cat: Unable to write to output stream
[hadoop@ip-172-31-14-46 ~]$
```

We will be entering the Hive CLI and creating database to store data

Create database estore;

Use estore;

We will use estore database to load data and do the analysis.

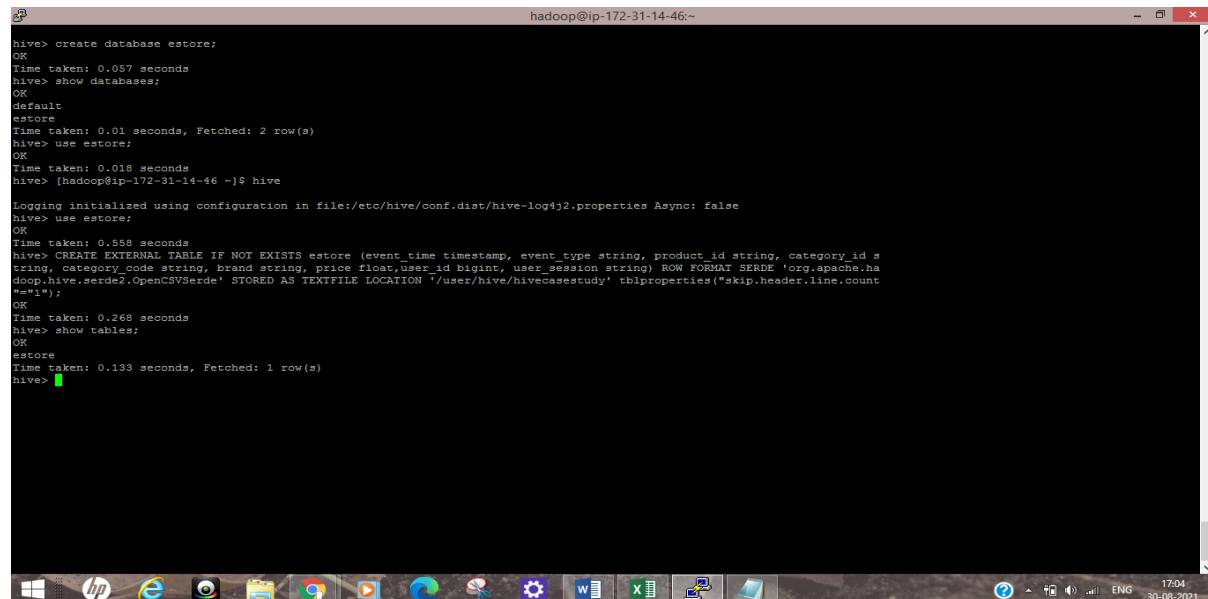


A screenshot of a web browser window titled "Hive - Create Database". The address bar shows "console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#cluster-detailsj-65RPRB5QF19Z". The main content area is a terminal window with the following Hive session:

```
hive> create database estore;
OK
Time taken: 0.057 seconds
hive> show databases;
OK
default
estore
Time taken: 0.01 seconds, Fetched: 2 row(s)
hive> use estore;
OK
Time taken: 0.018 seconds
hive>
```

Creating External Table Using CSVSerde:

```
CREATE EXTERNAL TABLE IF NOT EXISTS estore (event_time timestamp, event_type string,
product_id string, category_id string, category_code string, brand string, price float, user_id
bigint, user_session string) ROW FORMAT SERDE
'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS TEXTFILE LOCATION
'/user/hive/hivecasestudy' tblproperties("skip.header.line.count"="1");
```



A screenshot of a terminal window titled "hadoop@ip-172-31-14-46:~". The session shows the creation of a database and an external table:

```
hive> create database estore;
OK
Time taken: 0.057 seconds
hive> show databases;
OK
default
estore
Time taken: 0.01 seconds, Fetched: 2 row(s)
hive> use estore;
OK
Time taken: 0.018 seconds
hive> [hadoop@ip-172-31-14-46 ~]$ hive
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: false
hive> use estore;
OK
Time taken: 0.558 seconds
hive> CREATE EXTERNAL TABLE IF NOT EXISTS estore (event_time timestamp, event_type string, product_id string, category_id s
tring, category_code string, brand string, price float, user_id bigint, user_session string) ROW FORMAT SERDE 'org.apache.ha
doop.hive.serde2.OpenCSVSerde' STORED AS TEXTFILE LOCATION '/user/hive/hivecasestudy' tblproperties("skip.header.line.count"
="1");
OK
Time taken: 0.268 seconds
hive> show tables;
OK
estore
Time taken: 0.193 seconds, Fetched: 1 row(s)
hive>
```

To display the header columns

```
set hive.cli.print.header = true;
```

We are checking the top 5 rows in the table using

```
select * from estore limit 5;
```

```
hive> use estore;
OK
Time taken: 0.558 seconds
hive> CREATE EXTERNAL TABLE IF NOT EXISTS estore (event_time timestamp, event_type string, product_id string, category_id string, category_code string, brand string, price float, user_id bigint, user_session string) ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS TEXTFILE LOCATION '/user/hive/hivecasestudy' tblproperties('skip.header.line.count'='1');
OK
Time taken: 0.268 seconds
hive> show tables;
OK
estore
Time taken: 0.133 seconds, Fetched: 1 row(s)
hive> set hive.cli.print.header = true;
hive> select * from estore limit 5;
OK
estore.event_time      estore.event_type      estore.product_id      estore.category_id      estore.category_code      estore.brand      estore.price      estore.user_id
estore.user_session
2019-11-01 00:00:02 UTC view      5802432 1487580009286598681      0.32      562076640      09faf06c-6c99-46b1-834f-33527ff4de241
2019-11-01 00:00:09 UTC cart      5844397 1487580006317032337      2.38      553329724      2067216c-31b5-455d-a1cc-af0575a34ffb
2019-11-01 00:00:10 UTC view      5837166 1783999064103190764      pnb      22.22      556138645      57ed222e-a5fa-4907-9944-5a875c2d7ff
2019-11-01 00:00:11 UTC cart      5876912 1487580010100293687      jessnail      3.16      564506666      186c1951-8052-4b37-adce-dd9644b1d5f7
2019-11-01 00:00:24 UTC remove_from_cart  5826182 1487580007483048900      3.33      553329724      2067216c-31b5-455d-a1cc-af0575a34ffb
Time taken: 2.202 seconds, Fetched: 5 row(s)
hive> █
```

Creating Optimised Table:

We are creating the optimised tables using partitioning and Bucketing Techniques:

```
set hive.exec.dynamic.partition.mode = nonstrict;
```

```
set hive.exec.dynamic.partition = true;
```

```
set hive.enforce.bucketing = true;
```

Creating an optimized table by partitioning on “event_type” and bucketing on “price”

```
CREATE TABLE IF NOT EXISTS dynpart_bucket_estore(event_time timestamp, product_id string, category_id string, category_code string, brand string, price float, user_id bigint, user_session string) PARTITIONED BY (event_type string) CLUSTERED BY (price) INTO 10 BUCKETS ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS TEXTFILE LOCATION '/user/hive/hivecasestudy' tblproperties('skip.header.line.count' = '1');
```

Copying the data from the estore table:

```
INSERT INTO TABLE dynpart_bucket_estore PARTITION (event_type) SELECT event_time,product_id,category_id,category_code,brand,price,user_id,user_session,event_type FROM estore;
```

```

hive> CREATE TABLE IF NOT EXISTS dynpart_bucket_estore(event_time timestamp, product_id string, category_id string, category_code string, brand string, price float, user_id bigint, user_session string) PARTITIONED BY (event_type string) CLUSTERED BY (price) INTO 10 BUCKETS ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS TEXTFILE LOCATION '/user/hive/hivecasestudy' tblproperties('skip.header.line.count' = '1');
OK
Time taken: 0.078 seconds
hive> INSERT INTO TABLE dynpart_bucket_estore PARTITION (event_type) SELECT event_time,product_id,category_id,category_code,brand,price,user_id,user_session,even
t_type FROM estore;
Query ID = hadoop_20210830120840_0792a952-8bb6-4c6c-b82a-b825f2015bb4
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1630319660287_0005)

-----  

      VERTICES    MODE     STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  

-----  

Map 1 ..... container SUCCEEDED 2 2 0 0 0 0  

Reducer 2 ..... container SUCCEEDED 5 5 0 0 0 0  

-----  

VERTICES: 02/02 [=====>>>] 100% ELAPSED TIME: 160.49 s  

-----  

Loading data to table estore.dynpart_bucket_estore partition (event_type=null)  

Loaded : 4/4 partitions.  

  Time taken to load dynamic partitions: 1.043 seconds  

  Time taken for adding to write entity : 0.003 seconds  

OK
event_time      product_id      category_id      category_code   brand   price   user_id user_session   event_type
Time taken: 172.636 seconds
hive> 

```

Checking the tables created

```

hive> show tables;
OK
tab_name
dynpart_bucket_estore
estore
Time taken: 0.043 seconds, Fetched: 2 row(s)
hive> 

```

Verifying the partitioned files in the Hadoop file system

hadoop fs -ls /user/hive/hivecasestudy

Verifying the bucketed files in the Hadoop file system

hadoop fs -ls /user/hive/hivecasestudy/event_type=cart

```

[hadoop@ip-172-31-14-46 ~]$ hadoop fs -ls /user/hive/hivecasestudy
Found 6 items
-rw-r--r--  1 hadoop hadoop  545839412 2021-08-30 11:13 /user/hive/hivecasestudy/2019-Nov.csv
-rw-r--r--  1 hadoop hadoop  482542278 2021-08-30 11:10 /user/hive/hivecasestudy/2019-Oct.csv
drwxr-xr-x  - hadoop hadoop          0 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=cart
drwxr-xr-x  - hadoop hadoop          0 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=purchase
drwxr-xr-x  - hadoop hadoop          0 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=remove_from_cart
drwxr-xr-x  - hadoop hadoop          0 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=view
[hadoop@ip-172-31-14-46 ~]$ hadoop fs -ls /user/hive/hivecasestudy/event_type=cart
Found 10 items
-rwxr-xr-x  1 hadoop hadoop  27512579 2021-08-30 12:10 /user/hive/hivecasestudy/event_type=cart/000000_0
-rwxr-xr-x  1 hadoop hadoop  32190447 2021-08-30 12:10 /user/hive/hivecasestudy/event_type=cart/000001_0
-rwxr-xr-x  1 hadoop hadoop  33302805 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=cart/000002_0
-rwxr-xr-x  1 hadoop hadoop  32602023 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=cart/000003_0
-rwxr-xr-x  1 hadoop hadoop  34104132 2021-08-30 12:10 /user/hive/hivecasestudy/event_type=cart/000004_0
-rwxr-xr-x  1 hadoop hadoop  32538513 2021-08-30 12:10 /user/hive/hivecasestudy/event_type=cart/000005_0
-rwxr-xr-x  1 hadoop hadoop  39257340 2021-08-30 12:10 /user/hive/hivecasestudy/event_type=cart/000006_0
-rwxr-xr-x  1 hadoop hadoop  24825787 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=cart/000007_0
-rwxr-xr-x  1 hadoop hadoop  28504487 2021-08-30 12:11 /user/hive/hivecasestudy/event_type=cart/000008_0
-rwxr-xr-x  1 hadoop hadoop  35410315 2021-08-30 12:10 /user/hive/hivecasestudy/event_type=cart/000009_0
[hadoop@ip-172-31-14-46 ~]$ 

```

Checking the optimised tables using

select * from estore limit 5;

select * from dynpart_bucket_estore limit 5;

```
hive> use estore;
OK
Time taken: 0.069 seconds
hive> select * from estore limit 5;
OK
2019-11-01 00:00:02 UTC view      5802432 1487580009286598691          0.32    562076640      09fafd6c-6c99-46b1-834f-33527f4de241
2019-11-01 00:00:09 UTC cart      5844397 1487580006317032337          2.38    553329724      2067216c-31b5-455d-a1cc-af0575a34ffb
2019-11-01 00:00:10 UTC view      5837166 1783999064103190764          pnb     22.22   556198645      57ed222e-a5fa-4907-9944-5a875c2d7f4f
2019-11-01 00:00:11 UTC cart      5876812 14875800010100293687          jessnail  3.16    554506666      186c1951-8052-4b37-adce-dd9e644b1d5f7
2019-11-01 00:00:24 UTC remove_from_cart  5826182 1487580007483048900          3.33    553329724      2067216c-31b5-455d-a1cc-af0575a34ffb
Time taken: 2.143 seconds, Fetched: 5 row(s)
hive> select * from dynpart_bucket_estore limit 5;
OK
2019-10-08 09:19:19 UTC 89350  1487580011652186237      runail  1.27    232701853      3f1469f5-d926-44ce-a9f6-dff5ae276c9c      cart
2019-10-10 05:29:47 UTC 5866208 1487580013841613016      concept 3.16   493381333      535bb6b7-08f4-4021-ac66-b340178f7a37      cart
2019-10-08 12:25:50 UTC 5821183 1487580007717929935          1.27   546703849      3daf4d64-5ffa-46cc-827b-59760ebd819b      cart
2019-10-10 08:19:06 UTC 5848901 1487580007675986893      bpw.style 1.27   439370683      9aeb4d9a-1bed-4f42-b12d-88be1148d3a9      cart
2019-10-09 18:32:50 UTC 5869152 1487580005268456287      cosmoprofi 7.94   558533352      cfde0f74-8705-4a2f-ba83-a5b99581c294      cart
Time taken: 0.203 seconds, Fetched: 5 row(s)
```

Time taken by normal estore table is 2.143 Seconds, whereas partitioned and bucketed table could retrieve the data faster in .203 Seconds.

Optimised Tables are used for better performance and faster analysis.

Answers to the questions given below.

- 1) Find the total revenue generated due to purchases made in October.

Base Table: Estore

```
SELECT SUM(price) AS tot_revenue_oct FROM estore WHERE MONTH(event_time) = '10' AND event_type = 'purchase';
```

Optimised Table: Dynpart_bucket_estore

```
SELECT SUM(price) AS tot_revenue_oct FROM dynpart_bucket_estore WHERE MONTH(event_time) = '10' AND event_type = 'purchase';
```

```
hive> SELECT SUM(price) AS tot_revenue_oct FROM dynpart_bucket_estore WHERE MONTH(event_time) = '10' AND event_type = 'purchase';
Query ID = hadoop_20210830131019_7df1b9a2-368a-44e6-b660-31546468606b
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0007)

-----  

      VERTICES    MODE     STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED  

-----  

Map 1 ..... container  SUCCEEDED   3       3       0       0       0       0  

Reducer 2 ..... container  SUCCEEDED   1       1       0       0       0       0  

-----  

VERTICES: 02/02  [=====>>>] 100%  ELAPSED TIME: 22.38 s  

-----  

OK  

tot_revenue_oct  

1211532.4500002791  

Time taken: 23.489 seconds, Fetched: 1 row(s)
hive> SELECT SUM(price) AS tot_revenue_oct FROM estore WHERE MONTH(event_time) = '10' AND event_type = 'purchase';
Query ID = hadoop_20210830131143_aa3b800f-db98-4946-a44a-4ec8a1a9c635
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0007)

-----  

      VERTICES    MODE     STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED  

-----  

Map 1 ..... container  SUCCEEDED   5       5       0       0       0       0  

Reducer 2 ..... container  SUCCEEDED   1       1       0       0       0       0  

-----  

VERTICES: 02/02  [=====>>>] 100%  ELAPSED TIME: 125.93 s  

-----  

OK  

tot_revenue_oct  

1211538.4299997438  

Time taken: 126.554 seconds, Fetched: 1 row(s)
hive>
```

Output/Observations:

Total revenue generated based on purchase made in the month of October is 1,211,538.43 rupees.

Base table query took the execution time of 126 seconds whereas optimized table query took execution time of 23 seconds.

We can observe that there is drop in the execution time of the same query. Optimized table gives better performance in execution time.

-
- 2) Write a query to yield the total sum of purchases per month in a single output.

Base Table: Estore

```
SELECT MONTH(event_time) AS month, COUNT(event_type) AS total_purchases FROM estore
WHERE event_type = 'purchase' GROUP BY MONTH(event_time);
```

Optimised Table: Dynpart_bucket_estore

```
SELECT MONTH(event_time) AS month, COUNT(event_type) AS total_purchases FROM
dynpart_bucket_estore WHERE event_type = 'purchase' GROUP BY MONTH(event_time);
```

```

hadoop@ip-172-31-14-46:~ 
hive> SELECT MONTH(event_time) AS month, COUNT(event_type) AS total_purchases FROM estore WHERE event_type = 'purchase' GROUP BY MONTH(event_time);
Query ID = hadoop_20210830131817_c5468ebc-dda0-4ba7-80f5-d5fa46e29021
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0007)

-----  

    VERTICES      MODE      STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  

Map 1 ..... container    SUCCEEDED     5      5      0      0      0      0      0  

Reducer 2 ..... container    SUCCEEDED     3      3      0      0      0      0      0  

-----  

VERTICES: 02/02 [=====>>>] 100% ELAPSED TIME: 100.72 s  

-----  

OK  

month  total_purchases  

10      245624  

11      322417  

Time taken: 101.458 seconds, Fetched: 2 row(s)
hive> SELECT MONTH(event_time) AS month, COUNT(event_type) AS total_purchases FROM dynpart_bucket_estore WHERE event_type = 'purchase' GROUP BY MONTH(event_time);
Query ID = hadoop_20210830132221_f7a7b5ca-ac04-4e03-8267-28b8559e2052
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0007)

-----  

    VERTICES      MODE      STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  

Map 1 ..... container    SUCCEEDED     3      3      0      0      0      0      0  

Reducer 2 ..... container    SUCCEEDED     1      1      0      0      0      0      0  

-----  

VERTICES: 02/02 [=====>>>] 100% ELAPSED TIME: 23.01 s  

-----  

OK  

month  total_purchases  

10      245619  

11      322412  

Time taken: 23.846 seconds, Fetched: 2 row(s)
hive>

```

Output/Observations:

Count of purchases made in the month of October is 245624 and in the month of November 322417, we observe that there is 31% Increase in the November Purchases compared to October Purchases.

Base table query took the execution time of 101 seconds whereas optimized table query took execution time of 24 seconds.

We also observe that partitioning and bucketing techniques on table can reduce execution time substantially.

3) Write a query to find the change in revenue generated due to purchases from October to November.

Base Table: Estore

```
SELECT (SUM(CASE WHEN MONTH(event_time)=11 THEN price ELSE 0 END) - SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END)) AS diff_rev FROM dynpart_bucket_estore WHERE event_type = 'purchase' AND MONTH(event_time) in ('10','11');
```

Optimised Table: Dynpart_bucket_estore

```
SELECT (SUM(CASE WHEN MONTH(event_time)=11 THEN price ELSE 0 END) - SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END)) AS diff_rev FROM estore WHERE event_type = 'purchase' AND MONTH(event_time) in ('10','11');
```

```

hadoop@ip-172-31-14-46:~  

10      245619  

11      322412  

Time taken: 23.846 seconds, Fetched: 2 row(s)  

hive> SELECT (SUM(CASE WHEN MONTH(event_time)=11 THEN price ELSE 0 END) - SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END)) AS diff_rev FROM dynpart_bucket_estore WHERE event_type = 'purchase' AND MONTH(event_time) in ('10','11');  

Query ID = hadoop_20210830133734_f2e0f3be-ba0c-41f1-b600-b11233fdf958  

Total jobs = 1  

Launching Job 1 out of 1  

Tez session was closed. Reopening...  

Session re-established.  

Status: Running (Executing on YARN cluster with App id application_1630319660287_0008)  

-----  

   VERTICES    MODE     STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  

Map 1 ..... container SUCCEEDED   3      3      0      0      0      0  

Reducer 2 ..... container SUCCEEDED   1      1      0      0      0      0  

-----  

VERTICES: 02/02 [=====>>>] 100% ELAPSED TIME: 26.69 s  

OK  

diff_rev  

319437.789997565  

Time taken: 34.785 seconds, Fetched: 1 row(s)  

hive> SELECT (SUM(CASE WHEN MONTH(event_time)=11 THEN price ELSE 0 END) - SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END)) AS diff_rev FROM estore WHERE event_type = 'purchase' AND MONTH(event_time) in ('10','11');  

Query ID = hadoop_20210830133849_9ce7f2c5-eb59-43d1-b5e3-47c3b3j0369e0  

Total jobs = 1  

Launching Job 1 out of 1  

Status: Running (Executing on YARN cluster with App id application_1630319660287_0008)  

-----  

   VERTICES    MODE     STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  

Map 1 ..... container SUCCEEDED   5      5      0      0      0      0  

Reducer 2 ..... container SUCCEEDED   1      1      0      0      0      0  

-----  

VERTICES: 02/02 [=====>>>] 100% ELAPSED TIME: 102.84 s  

OK  

diff_rev  

319478.4700003781  

Time taken: 103.371 seconds, Fetched: 1 row(s)

```

Output/Observations:

We can observe that the difference in revenue between October purchases and November Purchases is 319478.

Base table query took the execution time of 103 seconds whereas optimized table query took execution time of 34 seconds

We also observe that partitioning and bucketing techniques on table can reduce execution time substantially.

We will be using the Optimised Table for further analysis.

4) Find distinct categories of products. Categories with null category code can be ignored.

Optimised Table: Dynpart_bucket_estore

```
SELECT DISTINCT SPLIT(category_code,'\\')[0] AS Category FROM dynpart_bucket_estore WHERE category_code != '';
```

```

hive> SELECT DISTINCT SPLIT(category_code,'\\.')[0] AS Category FROM dynpart_bucket_estore WHERE category_code != '';
Query ID = hadoop_20210830135316_d2ff13c8-ca6c-415e-a12e-53bf940149ab
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0008)

-----  

      VERTICES    MODE      STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  

-----  

Map 1 ..... container    SUCCEEDED   6       6       0       0       0       0  

Reducer 2 ..... container    SUCCEEDED   5       5       0       0       0       0  

-----  

VERTICES: 02/02  [=====>>] 100% ELAPSED TIME: 62.62 s  

-----  

OK  

category  

furniture  

appliances  

accessories  

apparel  

sport  

stationery  

Time taken: 63.599 seconds, Fetched: 6 row(s)
hive> 

```

Output/Observations:

We have 6 unique categories are –

Furniture	Apparel
Appliances	Sport
Accessories	Stationary

Time taken to execute the query is 63 seconds

5) Find the total number of products available under each category.

Optimised Table: Dynpart_bucket_estore

```

SELECT SPLIT(category_code,'\\.')[0] AS Category, COUNT(product_id) AS number_of_products
FROM dynpart_bucket_estore WHERE category_code != '' GROUP BY SPLIT(category_code,'\\.')[0]
ORDER BY number_of_products DESC;

```

```

hive> SELECT SPLIT(category_code,'\\.')[0] AS Category, COUNT(product_id) AS number_of_products FROM dynpart_bucket_estore WHERE category_code != '' GROUP BY SPLIT(category_code,'\\.')[0] ORDER BY number_of_products DESC;
Query ID = hadoop_20210830140746_7f879eca-be5b-4a28-9533-13da5efa00d
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0009)

-----  

      VERTICES    MODE      STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  

-----  

Map 1 ..... container    SUCCEEDED   4       4       0       0       0       0  

Reducer 2 ..... container    SUCCEEDED   5       5       0       0       0       0  

Reducer 3 ..... container    SUCCEEDED   1       1       0       0       0       0  

-----  

VERTICES: 03/03  [=====>>] 100% ELAPSED TIME: 63.96 s  

-----  

OK  

category      number_of_products  

appliances     61736  

stationery     26722  

furniture      23604  

apparel        18232  

accessories    12928  

sport          2  

Time taken: 64.794 seconds, Fetched: 6 row(s)
hive> 

```

Output/Observations:

Appliances	61736
Stationary	26722
Furniture	23604
Apparel	18232
Accessories	12928
Sport	2

Appliances are the most popular category with 61K Products, Stationary and Furniture are in the range of 26K to 23 K Products, Followed by Apparel with 18K Products, Accessories with almost 13K Product and Sports with 2 products.

Time Taken for execution of query is 65 Seconds

6) Which brand had the maximum sales in October and November combined?

Optimised Table: Dynpart_bucket_estore

WITH total_sales_summary AS(

```
SELECT brand, round((SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END) +  
SUM(CASE WHEN MONTH(event_time)=11 THEN PRICE ELSE 0 END)),2) AS total_sales FROM  
dynpart_bucket_estore WHERE event_type = 'purchase' AND MONTH(event_time) in ('10','11') AND  
brand != '' GROUP BY brand)
```

```
SELECT brand, total_sales FROM total_sales_summary ORDER BY total_sales DESC LIMIT 1;
```

```
hive> WITH total_sales_summary AS(  
>   SELECT brand, round((SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END) + SUM(CASE WHEN MONTH(event_time)=11 THEN PRICE ELSE 0 END)),2) AS total_sales  
>   FROM dynpart_bucket_estore  
>   WHERE event_type = 'purchase' AND MONTH(event_time) in ('10','11') AND brand != '' GROUP BY brand)  
>   SELECT brand, total_sales FROM total_sales_summary ORDER BY total_sales DESC LIMIT 1;  
Query ID = hadoop_20210830141719_88610c8c-5d63-4537-889a-68f0b8269e1a  
Total jobs = 1  
Launching Job 1 out of 1  
Status: Running (Executing on YARN cluster with App id application_1630319660287_0010)  
  
VERTICES      MODE      STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED  
Map 1 ..... container SUCCEEDED    3      3      0      0      0      0  
Reducer 2 .... container SUCCEEDED   1      1      0      0      0      0  
Reducer 3 .... container SUCCEEDED   1      1      0      0      0      0  
  
VERTICES: 03/03 [=====>>>] 100% ELAPSED TIME: 25.80 s  
  
OK  
brand      total_sales  
runail    148292.46  
Time taken: 26.876 seconds, Fetched: 1 row(s)
```

Runail is the brand that has the highest sales in total of both the months October and November

It seems that Runail brand has high popularity among the buyers and could help in increasing their profit.

Time Taken for execution of query is 27 Seconds

7) Which brands increased their sales from October to November?

Optimised Table: Dynpart_bucket_estore

WITH brand_sales_summary AS(

```
SELECT brand, round(SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END),2) AS  
Oct_sales,round(SUM(CASE WHEN MONTH(event_time)=11 THEN PRICE ELSE 0 END),2) AS  
Nov_sales FROM dynpart_bucket_estore WHERE event_type = 'purchase' AND MONTH(event_time)  
in ('10','11') AND brand != '' GROUP BY brand)
```

```
SELECT brand, Oct_sales, Nov_sales, round((Nov_sales-Oct_sales),2)AS Change_in_Sales FROM  
brand_sales_summary WHERE Nov_sales-Oct_sales > 0 ORDER BY Change_in_Sales DESC;
```

```

hadoop@ip-172-31-14-46:~ 
hive> WITH brand_sales_summary AS( SELECT brand, round(SUM(CASE WHEN MONTH(event_time)=10 THEN price ELSE 0 END),2) AS Oct_sales,round(SUM(CASE WHEN MONTH(event_time)=11 THEN price ELSE 0 END),2) AS Nov_sales,round((Nov_sales-Oct_sales),2)AS Change_in_Sales FROM brand_sales_summary WHERE Nov_sales-Oct_sales > 0 ORDER BY Change_in_Sales DESC );
Query ID = hadoop_20210830144353_a4aeb7c6-7f7f-451f-a135-cda180fc4d5
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0011)

----- VERTICES      MODE      STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED -----
Map 1 ..... container SUCCEEDED   3     3     0     0     0     0
Reducer 2 ..... container SUCCEEDED   1     1     0     0     0     0
Reducer 3 ..... container SUCCEEDED   1     1     0     0     0     0
----- VERTICES: 03/03 [=====>>>] 100% ELAPSED TIME: 26.48 s
----- OK
----- brand    oct_sales    nov_sales    change_in_sales
gratotol 35445.54    71472.71    36027.17
uno       35302.03    51039.75    15737.72
lilanail 5892.84     16394.24    10501.4
ingarden  23161.39    33566.21    10404.82
strong   29196.63     38671.27    9474.64
jessmail 26287.84     33345.23    7057.39
comoprofci 8322.81     14536.99    6214.18
polarus  6013.72     11371.93    5358.21
runail   71537.77    76754.69    5216.92
freedecor 3421.78     7671.8     4250.02
staleks  8519.73     11875.61    3355.88
bpw.style 11572.15    14837.44    3265.29
lovelyc  8704.38     11939.06    3234.68
marathon  7280.75     10273.1    2992.35
haruyama 9390.69     12352.91    2962.22
yoko     8756.91     11707.88    2950.97
italwax  21940.24    24799.37    2859.13
benovyy 409.62      3259.97    2850.35
kaypro   881.34      3268.7     2387.36
estel    21756.75    24142.67    2385.92
concept  11032.14    13380.4    2348.26
kapous  11927.16    14093.08    2165.92
f.o.x   6624.23     8577.28    1953.05
masura  31266.08    33058.47    1792.39
----- 2016 30-08-2021

```

```

hadoop@ip-172-31-14-46:~ 
masura  31266.08    33058.47    1792.39
milv   3904.94     5642.01    1737.07
beautix 10493.95    12222.95    1729.0
artex  2730.64     4327.25    1596.61
domix  10472.05    12009.17    1537.12
shilk  3341.2      4839.72    1498.52
smart  4457.26     5902.14    1444.88
roublöff 3491.36     4913.77    1422.41
levranz 2243.56     3664.1     1420.54
onig   8425.41     9841.65    1416.24
irisk  45591.96    46946.04    1354.08
severina 4775.88     6120.48    1344.6
joico  705.52      2015.1     1309.58
zeitun 708.66      2009.63    1300.97
beauty-free 554.17     1782.86    1228.69
swarovski 1887.93     3043.16    1155.23
de.lux 1659.7      2775.51    1115.81
metzger 5373.45     6457.16    1083.71
markell 1768.75     2834.43    1065.68
sanoto  157.14     1209.68    1052.54
nagaraku 4369.74     5327.68    957.94
ecolab  262.85     1214.3     951.45
art-visage 2092.71     2997.8     905.09
levissime 2227.5     3085.31    857.81
missa  1293.83     2150.28    856.45
solomeya 1899.7      2685.8     786.1
rosi   3077.04     3841.56    764.52
refectocil 2716.18     3475.58    759.4
kaaral  4412.43     5086.07    673.64
kosmekka 1181.44     1813.37    631.93
kinetics 6334.25     6945.26    611.01
brownxenna 14331.37    14916.73    585.36
airnails 5118.9      5691.52    572.62
uskuusi 5142.27     5690.31    548.04
coifin 903.0       1428.49    525.49
s.care 412.68      913.07    500.39
limoni 1308.9      1796.6     487.7
matrix  3243.25     3726.74    483.49
gehwol 1089.07     1557.68    468.61
greymy 29.21       489.49    460.28
bioagua 942.89      1398.12    455.23
farmavita 837.37      1291.97    454.6
sophinin 1067.86     1515.52    447.66
yu-r   271.41      673.71     402.3
----- 2017 30-08-2021

```

```
yu-x 271.41 673.71 402.3
kiss 421.55 817.33 395.78
naomi 0.0 389.0 389.0
lador 2083.61 2471.53 387.92
ellips 245.85 606.04 360.19
jas 3318.96 3657.43 338.47
lowence 242.84 567.75 324.91
nittile 847.28 1162.68 315.4
shary 871.96 1176.49 304.53
kims 330.04 632.04 302.0
happyfons 801.92 1091.59 289.67
kocostar 310.85 594.93 284.08
insight 1443.7 1721.96 278.26
candy 534.96 799.38 264.42
bluesky 10307.24 10565.53 258.29
beaugreen 511.51 768.35 256.84
protokeratin 201.25 456.79 255.54
trind 298.07 542.96 244.89
entity 479.71 719.26 239.55
skinlite 651.94 890.45 238.51
prococ 827.99 1063.82 235.83
fedua 52.38 263.81 211.43
ecocraft 41.16 241.95 200.79
keen 236.35 435.62 199.27
mane 66.79 260.26 193.47
freshbubble 318.7 502.34 183.64
matreshka 0.0 182.67 182.67
chi 358.94 538.61 179.67
cristalinas 427.63 584.95 157.32
farmona 1692.46 1843.43 150.97
latinol 249.52 384.59 135.07
miskin 158.04 293.07 135.03
elizavecca 70.53 204.3 133.77
nefertiti 233.52 366.64 133.12
finish 98.38 230.38 132.0
igrobeauty 513.66 645.07 131.41
dizao 819.13 945.51 126.38
osmo 645.58 762.31 116.73
batiste 772.4 874.17 101.77
carmex 145.08 243.36 98.28
eos 54.34 152.61 98.27
depiliflax 2707.07 2803.78 96.71
enjoy 41.35 136.57 95.22
kerasys 430.91 525.2 94.29
```

hadoop@ip-172-31-14-46:~

```
lmm 288.02 351.21 63.19
dewal 0.0 61.29 61.29
marutaka-foot 49.22 109.33 60.11
kares 0.0 59.45 59.45
profenhna 679.23 736.85 57.62
koelcia 55.5 112.75 57.25
balbcare 155.33 212.38 57.05
elskin 251.09 307.65 56.56
foamie 35.04 80.49 45.45
ladykin 125.65 170.57 44.92
likato 296.06 340.97 44.91
mavala 409.04 446.32 37.28
vilenta 197.6 231.21 33.61
beautyblender 78.74 109.41 30.67
biore 60.65 90.31 29.66
orly 902.38 931.09 28.71
estelare 444.81 471.87 27.06
profepil 93.36 118.02 24.66
blizix 38.95 63.4 24.45
binacil 0.0 24.26 24.26
godefroy 401.22 425.12 23.9
glysolid 69.73 91.59 21.86
veraclara 50.11 71.21 21.1
juno 0.0 21.08 21.08
kamili 63.01 81.49 18.48
treaclemon 163.37 181.49 18.12
supertran 50.37 66.51 16.14
barbie 0.0 12.39 12.39
deoprocce 316.84 329.17 12.33
rasyan 18.8 28.94 10.14
fly 17.14 27.17 10.03
terto 236.16 245.8 9.64
jaguar 1102.11 1110.65 8.54
soleo 204.2 212.53 8.33
neolecr 43.41 51.7 8.29
mayon 5.71 10.28 4.57
bodyton 1376.34 1380.64 4.3
skinity 8.88 12.44 3.56
hellologic 0.0 3.1 3.1
grace 100.92 102.61 1.69
cosima 20.23 20.93 0.7
ovale 2.54 3.1 0.56
```

Time taken: 27.098 seconds, Fetched: 160 row(s)

hive> 

Output/Observations:

160 brands have increased the selling from October to November.

'Grattol' brand has the highest total increment i.e., 36,027 /- and 'Ovale' seems to have the least increment of 0.56 /- from October to November.

'Runail' is the best and popular brand among the buyers with highest sales of around 70K in October and November.

Time Taken for execution of query is 27 Seconds

8) Your company wants to reward the top 10 users of its website with a Golden Customer plan. Write a query to generate a list of top 10 users who spend the most.

```
SELECT user_id, round(SUM(price),2) AS total_spends FROM dynpart_bucket_estore WHERE event_type = 'purchase' GROUP BY user_id ORDER BY total_spends DESC LIMIT 10;
```

```
hive> SELECT user_id, round(SUM(price),2) AS total_spends FROM dynpart_bucket_estore WHERE event_type = 'purchase' GROUP BY user_id ORDER BY total_spends DESC LIMIT 10;
Query ID = hadoop_20210830150031_aac74086-3154-43c2-9b2f-54b97b544353
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630319660287_0012)

-----  
 VERTICES      MODE      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED  
-----  
Map 1 ..... container  SUCCEEDED   3       3       0       0       0       0  
Reducer 2 ..... container  SUCCEEDED   1       1       0       0       0       0  
Reducer 3 ..... container  SUCCEEDED   1       1       0       0       0       0  
-----  
VERTICES: 03/03 [=====>>>] 100% ELAPSED TIME: 26.19 s  
-----  
OK  
user_id total_spends  
557790271  2715.87  
150318419  1645.97  
562167663  1352.85  
531900924  1329.45  
557850743  1295.48  
522130011  1185.39  
561592095  1109.7  
431950134  1097.59  
566576008  1056.36  
521347209  1040.91  
Time taken: 27.111 seconds, Fetched: 10 row(s)
hive>
```

Output/Observations:

Extracted the list of Top 10 Customers who has spent most in the website and shall be rewarded with Golden Customers to improve the customer engagement.

Time Taken for execution of query is 27 Seconds

We have completed the analysis, Let us terminate the EMR Cluster.

Click on the Terminate Protection and Disable the option and save it.

The screenshot shows the AWS EMR Cluster details page for a cluster named 'HiveCaseStudy'. The cluster status is 'Waiting' and it is described as 'Cluster ready after last step completed.' The 'Summary' tab is selected, showing the cluster ID (j-65RPRB5QF192), creation date (2021-08-30 15:56 UTC+5:30), and elapsed time (5 hours, 3 minutes). The 'Termination protection' setting is set to 'Off'. The 'Configuration details' section includes the release label (emr-5.29.0), Hadoop distribution (Amazon 2.8.5), applications (Hive 2.3.6, Pig 0.17.0, Hue 4.4.0), and log URI (S3://aws-lcqs-547065127688-us-east-1). The bottom of the page includes standard AWS navigation links and a footer with copyright information.

After saving, click on the Terminate

The screenshot shows the AWS EMR Cluster details page for 'Cluster: HiveCaseStudy'. The status is 'Waiting' with the message 'Cluster ready after last step completed.' The 'Summary' tab is selected, showing the cluster ID (j-65RPRB5QF19Z), creation date (2021-08-30 15:56 UTC+5:30), and elapsed time (5 hours, 3 minutes). It also shows the termination protection status ('Off') and the master public DNS (ec2-3-236-37-163.compute-1.amazonaws.com). The configuration details section includes the release label (emr-5.29.0), Hadoop distribution (Amazon 2.8.5), and applications (Hive 2.3.6, Pig 0.17.0, Hue 4.4.0). The log URI is s3://aws-logs-547065127688-us-east-1/elasticmapreduce/.

EMR Cluster status has changed from waiting to Terminating

The screenshot shows the AWS EMR Cluster details page for 'Cluster: HiveCaseStudy'. The status is now 'Terminating' with the message 'Terminated by user request'. The rest of the information remains the same as in the previous screenshot, including the cluster ID, creation date, elapsed time, termination protection status, master public DNS, configuration details, and log URI.

The screenshot shows the AWS EMR Management Console interface. On the left, there's a sidebar with navigation links for Amazon EMR, EMR Studio, EMR on EC2 (selected), Clusters, Notebooks, Git repositories, Security configurations, Block public access, VPC subnets, Events, EMR on EKS, and Virtual clusters. The main content area displays a table of clusters. The table has columns for Name, ID, Status, Creation time (UTC+5:30), Elapsed time, and Normalized instance hour. There are two entries:

	Name	ID	Status	Creation time (UTC+5:30)	Elapsed time	Normalized instance hour
<input type="checkbox"/>	HiveCaseStudy	J-65RPRB5QF19Z	Terminating User request	2021-08-30 15:56 (UTC+5:30)	5 hours, 56 minutes	48
<input type="checkbox"/>	demo_emr_cluster	J-S4KJP9B1Z91J	Terminated User request	2021-08-19 14:36 (UTC+5:30)	5 hours, 27 minutes	48

Cluster has Terminated Sccesfully

This screenshot is identical to the one above, showing the AWS EMR Management Console with the same cluster list and status information. Both clusters are now explicitly labeled as 'Terminated User request'.

All the Instances are also Terminated Successfully

The screenshot shows the AWS EC2 Management Console interface. On the left, there's a sidebar with navigation links like 'New EC2 Experience', 'EC2 Dashboard', 'Events', 'Tags', 'Limits', 'Instances' (selected), 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Scheduled Instances', 'Capacity Reservations', and 'Images'. The main content area has a title 'Instances (3) Info' with a search bar and filters. A table lists three instances: one terminated m4.large instance and two terminated t2.micro instances. Below the table is a modal window titled 'Select an instance above'.

Links of Web Interface Applications of HDFS, MasterNode, Hue

The screenshot shows the AWS EMR Cluster details page for 'HiveCaseStudy'. The sidebar includes links for 'Amazon EMR', 'EMR Studio', 'EMR on EC2' (selected), 'Clusters', 'Notebooks', 'Git repositories', 'Security configurations', 'Block public access', 'VPC subnets', 'Events', 'EMR on EKS', 'Virtual clusters', 'Help', and 'What's new'. The main content shows a cluster status of 'Waiting' and a table of 'On-cluster application user interfaces'. It also lists 'Web interfaces you can view on the task nodes' and a 'High-level application history' table.

List of Events that has done during the analysis:

Time	Event description	Source ID	Source type	Event type	Severity	Full date & time
Aug 30 04:08 PM	Amazon EMR cluster j-65RPRB5QF19Z (HiveCaseStudy) finished running all pending steps at 2021-08-30 10:37 UTC.	J-65RPRB5QF19Z	Cluster	Cluster State Change	INFO	August 30, 2021 at 04:08:34 PM (UTC+5:30)
Aug 30 04:07 PM	Amazon EMR cluster j-65RPRB5QF19Z (HiveCaseStudy) began running steps at 2021-08-30 10:37 UTC.	J-65RPRB5QF19Z	Cluster	Cluster State Change	INFO	August 30, 2021 at 04:07:48 PM (UTC+5:30)
Aug 30 03:56 PM	Amazon EMR cluster j-65RPRB5QF19Z (HiveCaseStudy) was requested at 2021-08-30 10:26 UTC and is being created.	J-65RPRB5QF19Z	Cluster	Cluster State Change	INFO	August 30, 2021 at 03:56:59 PM (UTC+5:30)
Aug 30 04:08 PM	Step s-1SXTZAG8DH3SF (Setup hadoop debugging) in Amazon EMR cluster j-65RPRB5QF19Z (HiveCaseStudy) completed execution at 2021-08-30 10:37 UTC. The step started running at 2021-08-30 10:37 UTC and took 0 minutes to complete.	s-1SXTZAG8DH3SF	Step	Step State Change	INFO	August 30, 2021 at 04:08:33 PM (UTC+5:30)

List of Activities that have taken place in Yarn Applications

Application ID	Type	Action	Status	Start time (UTC+5:30)	Duration	Finish time (UTC+5:30)
application_1630319660287_0001	TEZ	978f-ebf153007248	Succeeded	(UTC+5:30)	23 min	(UTC+5:30)
application_1630319660287_0006	TEZ	HIVE-311452df-82ee-49b3-978f-ebf153007248	Succeeded	2021-08-30 18:26 (UTC+5:30)	5.2 min	2021-08-30 18:31 (UTC+5:30)
application_1630319660287_0005	TEZ	HIVE-90844369-7ea2-45be-aa80-636aa306a2bf9	Succeeded	2021-08-30 17:38 (UTC+5:30)	8.2 min	2021-08-30 17:46 (UTC+5:30)
application_1630319660287_0004	TEZ	HIVE-90844369-7ea2-45be-aa80-636aa306a2bf9	Succeeded	2021-08-30 17:03 (UTC+5:30)	5.2 min	2021-08-30 17:08 (UTC+5:30)
application_1630319660287_0003	TEZ	HIVE-23e41015-e21a-41f4-9d47-95c3e47c5888	Succeeded	2021-08-30 16:48 (UTC+5:30)	5.2 min	2021-08-30 16:53 (UTC+5:30)
application_1630319660287_0002	MapReduce	distcp	Succeeded	2021-08-30 16:42 (UTC+5:30)	26 s	2021-08-30 16:43 (UTC+5:30)
application_1630319660287_0001	MapReduce	distcp	Succeeded	2021-08-30 16:39 (UTC+5:30)	30 s	2021-08-30 16:40 (UTC+5:30)

The screenshot shows the AWS EMR console with the 'High-level application history' page open. The left sidebar shows navigation links for Amazon EMR, EMR Studio, EMR on EC2 (Clusters, Notebooks, Git repositories, Security configurations, Block public access, VPC subnets, Events), EMR on EKS (Virtual clusters), Help, and What's new. The main content area displays a table of YARN applications:

Application ID	Type	Action	Status	Start time (UTC+5:30)	Duration	Finish time (UTC+5:30)
application_1630319660287_0012	TEZ	HIVE-8a4bcd35-455f-4658-aea6-bc022a9a211f	Succeeded	2021-08-30 20:29 (UTC+5:30)	7.2 min	2021-08-30 20:36 (UTC+5:30)
application_1630319660287_0011	TEZ	HIVE-8a4bcd35-455f-4658-aea6-bc022a9a211f	Succeeded	2021-08-30 20:10 (UTC+5:30)	9.2 min	2021-08-30 20:19 (UTC+5:30)
application_1630319660287_0010	TEZ	HIVE-8a4bcd35-455f-4658-aea6-bc022a9a211f	Succeeded	2021-08-30 19:44 (UTC+5:30)	8.2 min	2021-08-30 19:53 (UTC+5:30)
application_1630319660287_0009	TEZ	HIVE-8a4bcd35-455f-4658-aea6-bc022a9a211f	Succeeded	2021-08-30 19:28 (UTC+5:30)	15 min	2021-08-30 19:44 (UTC+5:30)
application_1630319660287_0008	TEZ	HIVE-311452df-82ee-49b3-9781-ebf153007248	Succeeded	2021-08-30 19:07 (UTC+5:30)	22 min	2021-08-30 19:29 (UTC+5:30)
application_1630319660287_0007	TEZ	HIVE-311452df-82ee-49b3-9781-ebf153007248	Succeeded	2021-08-30 18:35 (UTC+5:30)	23 min	2021-08-30 18:58 (UTC+5:30)
application_1630319660287_0006	TEZ	HIVE-311452df-82ee-49b3-9781-ebf153007248	Succeeded	2021-08-30 18:26 (UTC+5:30)	5.2 min	2021-08-30 18:31 (UTC+5:30)
		HIVE-90844369-7ea2-45be-	-	2021-08-30 17:38	-	2021-08-30 17:46

We have completed the analysis using Hive Successfully using EMR Cluster.

Thank You!

