

Cognizant Technology Solutions



A Project Report on

“EMPLOYEE PERFORMANCE TRACKING”

Prepared By

Ankita Shivane	2320390
Bhavesh Uprikar	2320803
Harshit Yadav	2320382
Mimansa Yadav	2320956
Nandhini T	2320539
Rajanya Kamilya	2320828
Shambhavi Shukla	2320529

REQUIREMENTS

Employee Performance Tracking:

- Import employee performance data into Amazon Redshift.
- Track employee performance metrics such as sales targets, customer satisfaction ratings, and productivity.
- Identify top-performing employees by analyzing performance metrics and comparing them to predefined targets or benchmarks.
- Identify areas for improvement by analyzing performance gaps and providing targeted training or support.
- Visualize employee performance data using charts and graphs (e.g., bar charts, radar charts) to identify trends and patterns.

INDEX

TOPIC

1. PURPOSE
2. PICTORIAL FLOWCHART
3. DESIGN & ARCHITECTURE

3.1. VPC

- VPC CREATION
- SUBNET
- INTERNET GATEWAY
- ROUTE TABLE

3.2. EC2

- EC2 INSTANCE
- SECURITY GROUP

3.3. S3

- BUCKET CREATION

3.4.IAM ROLE ASSOCIATION

- IAM ROLE

3.5. REDSHIFT

- CLUSTER CREATION
- CONFIGURE VPC, CLUSTER SUBNET GROUP, SECURITY GROUP

4. QUERY GENERATION

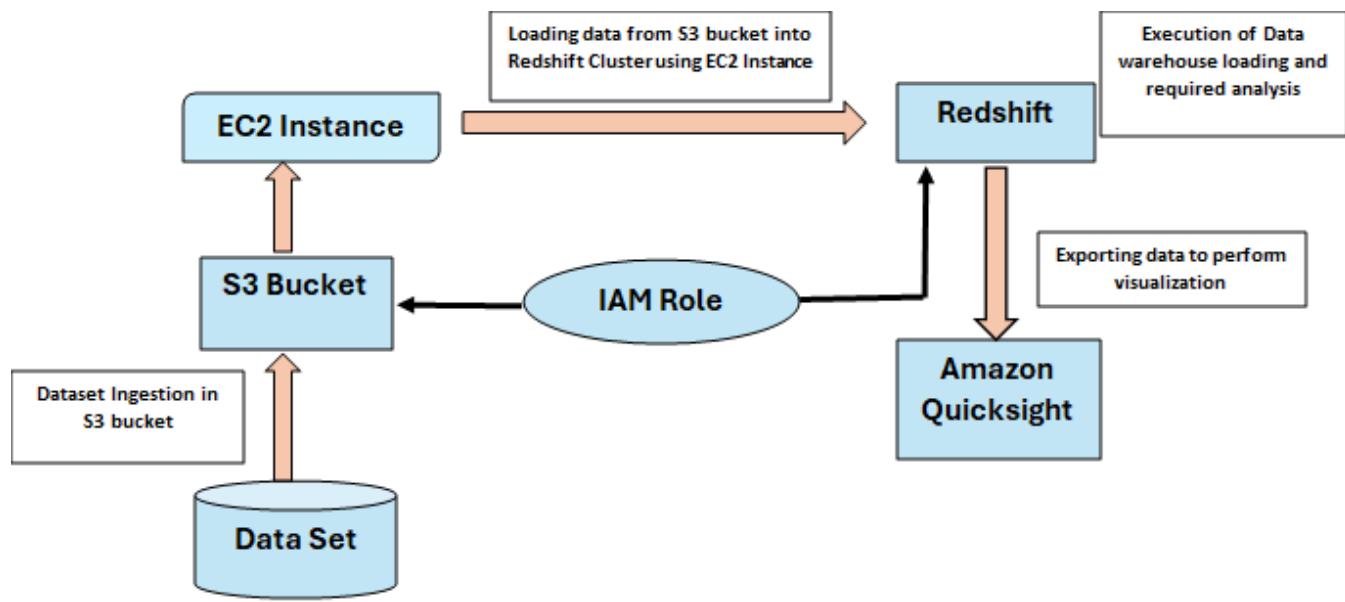
5. VIRTUALIZATION

6.CONCLUSION

PURPOSE:

- **Data Analysis:** Perform complex analytics on large datasets stored in S3 using the powerful querying capabilities of Redshift.
- **Business Intelligence (BI):** Create reports, dashboards, and visualizations from data stored in S3 by loading it into Redshift for BI purposes.
- **Data Warehousing:** Store and organize large volumes of structured and semi-structured data from S3 into Redshift for centralized data warehousing.
- **Data Integration:** Integrate data from multiple sources stored in S3 into a single data warehouse in Redshift for comprehensive analysis and reporting.
- **Data Migration:** Migrate data from on-premises data warehouses or other cloud storage solutions to Redshift for better scalability, performance, and cost-effectiveness.
- **ETL (Extract, Transform, Load):** Extract data from S3, transform it as needed, and load it into Redshift for further analysis and reporting.
- **Real-time Analytics:** Continuously load streaming data from S3 into Redshift to perform real-time analytics and gain insights into rapidly changing data.
- **Cost Optimization:** Optimize costs by storing raw or historical data in S3's cost-effective storage and transferring only relevant data into Redshift for analysis.
- **Data Archiving:** Archive historical data from S3 into Redshift for long-term storage and analysis while keeping it easily accessible for future reference.
- **Data Governance and Security:** Centralize sensitive data stored in S3 into Redshift for better governance, access control, and security management, ensuring compliance with regulatory requirements.

PICTORIAL FLOWCHART



DESIGN AND ARCHITECTURE

VPC CREATION

Step 1: Initiate the VPC creation process, provide the necessary details for your VPC, including the VPC name, CIDR block (the range of IP addresses for your VPC).

The screenshot shows the AWS VPC Dashboard. On the left, a sidebar lists options such as Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, and NAT gateways. The main area displays 'Resources by Region' for the Asia Pacific region, showing counts for VPCs, Subnets, Route Tables, Internet Gateways, NAT Gateways, VPC Peering Connections, Network ACLs, and Security Groups. A central note says, "Note: Your Instances will launch in the Asia Pacific region." To the right, there's a 'Service Health' section with a link to 'View complete service health details', and a 'Settings' section with links to 'Zones' and 'Console Experiments'. Below these are sections for 'Additional Information' (links to Documentation, All VPC Resources, Forums, and Report an Issue) and 'AWS Network Manager' (a brief description of its features). The bottom of the screen shows the Windows taskbar with various pinned icons.

The screenshot shows the 'Create VPC' wizard. The current step is 'VPC settings'. It includes fields for 'Resources to create' (set to 'VPC only'), 'Name tag - optional' (set to 'group5vpc'), 'IPv4 CIDR block' (set to '172.16.0.0/16'), and 'IPv6 CIDR block' (left empty). The top navigation bar shows the path 'VPC > Your VPCs > Create VPC'. The bottom of the screen shows the Windows taskbar.

The screenshot shows the AWS VPC Create VPC console. It includes sections for IPv6 CIDR block (radio buttons for 'No IPv6 CIDR block', 'IPAM-allocated IPv6 CIDR block', 'Amazon-provided IPv6 CIDR block', and 'IPv6 CIDR owned by me'), Tenancy (set to 'Default'), and Tags (with a key 'Name' set to 'group5vpc'). A 'Create VPC' button is at the bottom right.

Step 2: Define the subnets within your VPC and specify the CIDR block for subnet.

The screenshot shows the AWS VPC Create Subnet console. It displays the 'VPC' section with a VPC ID selected ('vpc-05eb10f7aaac9592c (group5vpc)'). In the 'Associated VPC CIDRs' section, an IPv4 CIDR is listed as '172.16.0.0/16'. The 'Subnet settings' section is partially visible. A 'Create subnet' button is located at the bottom left.

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs

Tags - optional

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="subnet5"/> X
Remove	

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

[Add new subnet](#)

Cancel **Create subnet**

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS VPC Management Console. A success message at the top states: "You have successfully created 1 subnet: subnet-00b9c47147ac501fc". The main area displays a table titled "Subnets (1/1) Info" with one item: "subnet5" (Subnet ID: subnet-00b9c47147ac501fc). A context menu is open over this item, with "View details" selected. Other options in the menu include: Create flow log, Edit subnet settings, Edit IPv6 CIDs, Edit network ACL association, Edit route table association, Edit CIDR reservations, Share subnet, Manage tags, and Delete subnet.

The screenshot shows the "Edit subnet settings" page for the subnet "subnet-00b9c47147ac501fc". The "Subnet" section shows the Subnet ID as "subnet-00b9c47147ac501fc" and the Name as "subnet5". The "Auto-assign IP settings" section contains two options: "Enable auto-assign public IPv4 address" (checked) and "Enable auto-assign customer-owned IPv4 address" (disabled, noted as "Option disabled because no customer owned pools found"). The "Resource-based name (RBN) settings" section is present but empty. The bottom of the screen shows the Windows taskbar with various application icons.

Enable auto-assign customer-owned IPv4 address [Info](#)
Option disabled because no customer owned pools found.

Resource-based name (RBN) settings [Info](#)
Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

Enable resource name DNS A record on launch [Info](#)

Enable resource name DNS AAAA record on launch [Info](#)

Hostname type [Info](#)
 Resource name
 IP name

DNS64 settings
Enable DNS64 to allow IPv6-only services in Amazon VPC to communicate with IPv4-only services and networks.

Enable DNS64 [Info](#)

Cancel **Save**

Step 3: Configure Internet Gateway: create a new internet gateway, give it a name, and attach it to your VPC.

Create internet gateway [Info](#)
An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="gatewaygr5"/> Remove

[Add new tag](#)
You can add 49 more tags.

Cancel **Create internet gateway**

Internet gateways (1/1) [Info](#)

Name	Internet gateway ID
<input checked="" type="checkbox"/> gatewaygr5	igw-062aa89731b4e51d0

Actions ▾ [Create internet gateway](#)

- [View details](#)
- [Attach to VPC](#)
- [Detach from VPC](#)
- [Manage tags](#)
- [Delete internet gateway](#)

igw-062aa89731b4e51d0 / gatewaygr5

[Details](#) Tags

Details

Internet gateway ID	State	VPC ID	Owner
igw-062aa89731b4e51d0	Detached	-	896334692798

VPC > Internet gateways > Attach to VPC (igw-062aa89731b4e51d0) [Info](#)

VPC

Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs

Attach the internet gateway to this VPC.

[▶ AWS Command Line Interface command](#)

[Cancel](#) [Attach internet gateway](#)

Step 4: Configure Route Table: create a route table, give it a name, and attach it to internet gateway.

The screenshot shows the AWS VPC Route Tables console. A context menu is open over the route table 'routegr5'. The 'Edit subnet associations' option is highlighted. The main pane displays the details of the route table 'rtb-0e979cefd75c9faaf'.

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-0e979cefd75c9faaf	Yes	-	-

The screenshot shows the 'Edit subnet associations' dialog box. It lists available subnets and selected subnets. The selected subnet is 'subnet-00b9c47147ac501fc / subnet5'. The 'Save associations' button is highlighted.

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
subnet5	subnet-00b9c47147ac501fc	172.16.1.0/24	-	Main (rtb-0e979cefd75c9faaf / rtb-0e979cefd75c9faaf)

The screenshot shows the AWS VPC Edit Routes page. It displays two routes:

Route 1

Destination	Target	Status
172.16.0.0/16	local	Active

Route 2

Destination	Target	Status
0.0.0.0/0	Internet Gateway	-

Below the routes, there is a "Propagated" section with the value "No".

The screenshot shows the AWS VPC Edit Routes page. It displays two routes:

Route 2

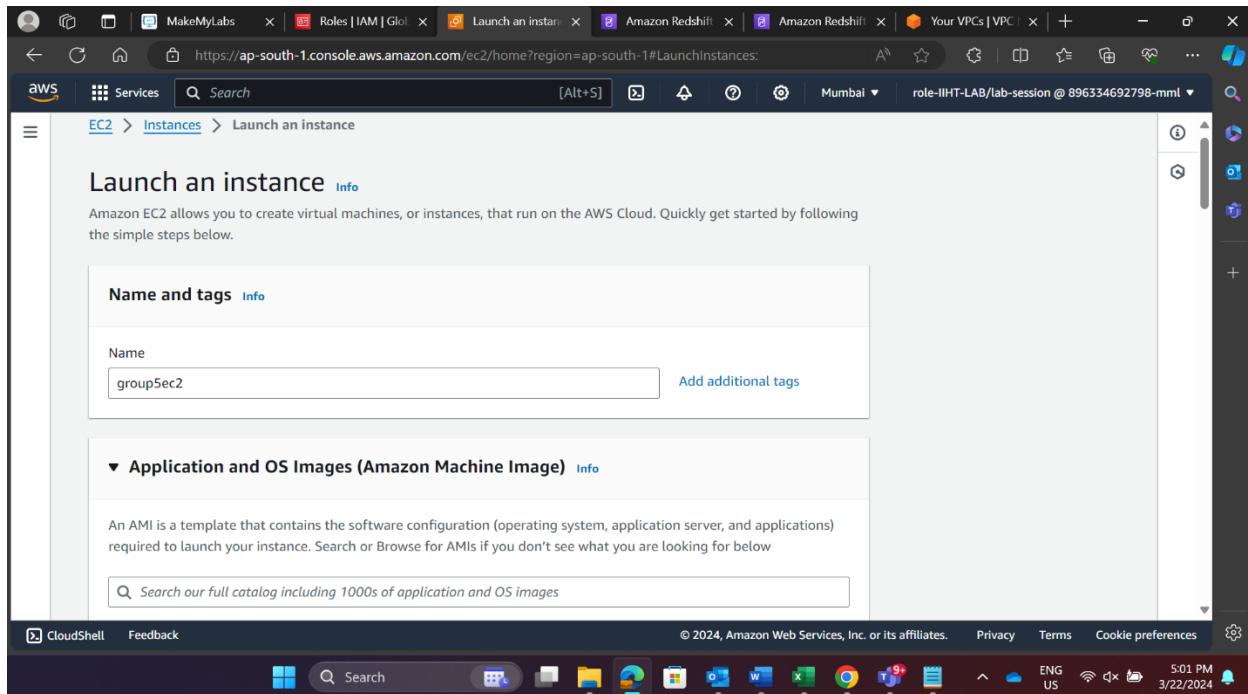
Destination	Target	Status
0.0.0.0/0	Internet Gateway	-

Below the routes, there is a "Propagated" section with the value "No".

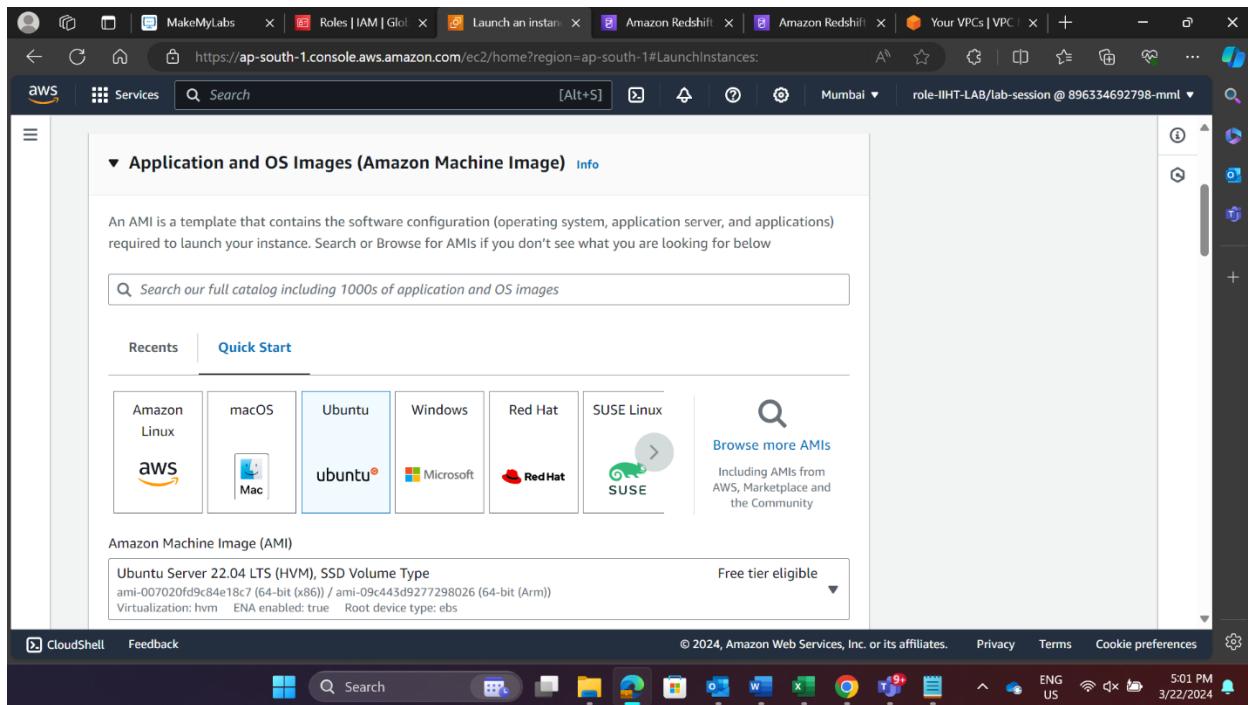
At the bottom right, there are "Cancel", "Preview", and "Save changes" buttons.

EC2 INSTANCE

Step 1: On the EC2 dashboard, click the “Launch Instance” button. This will start the instance creation wizard.



Step 2: Choose an Amazon Machine Image (AMI). It is a template for instance's operating system.



Step 3: In this step, select the type of instance based on your business requirements.

Description
Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2024-03-01

Architecture
64-bit (x86) AMI ID ami-007020fd9c84e18c7 Verified provider

Instance type
t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0724 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

All generations
Compare instance types

Additional costs apply for AMIs with pre-installed software

Step 4: If you don't have an existing key pair, create a new one. Download the private key file (.pem) and keep it secure.

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.
grkey

Key pair type
 RSA RSA encrypted private and public key pair
 ED25519 ED25519 encrypted private and public key pair

Private key file format
 .pem For use with OpenSSH
 .ppk For use with PuTTY

Cancel Create key pair

Step 5: Set up networking details like VPC, subnet, and security groups.

Network settings

Network: vpc-05eb10f7aaac9592c | group5vpc

Subnet: subnet-00b9c47147ac501fc | subnet5

Auto-assign public IP: Enable

Firewall (security groups)

We'll create a new security group called 'launch-wizard-1' with the following rules:

- Allow SSH traffic from Anywhere (0.0.0.0/0)
- Allow HTTPS traffic from the internet

Step 6: Add rules for SSH and Redshift in inbound rule for allowing the traffic to redshift cluster.

Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0073bc65420182e5b	SSH	TCP	22	A... 0.0.0.0/0 X	Delete
-	Redshift	TCP	5439	A... 0.0.0.0/0 X	Delete

Add rule

S3 BUCKET

Step 1: Create bucket by giving unique name and select the AWS region where you want to create bucket. Configure bucket permissions whether to keep the bucket private or public. After creating bucket add objects.

The screenshot shows the Amazon S3 homepage. On the left, there's a sidebar with a menu icon and the text "Storage". The main area features the heading "Amazon S3" and the subtext "Store and retrieve any amount of data from anywhere". Below this, a paragraph explains that Amazon S3 is an object storage service with industry-leading scalability, data availability, security, and performance. A large orange "Create bucket" button is prominently displayed. To the right, a "Pricing" section is visible with the text: "With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket." At the bottom of the page, there's a "How it works" section with a video thumbnail and some descriptive text. The browser's address bar shows the URL <https://s3.console.aws.amazon.com/s3/get-started?region=ap-south-1>. The status bar at the bottom right indicates the date and time as 3/22/2024 4:39 PM.

The screenshot shows the "Create bucket" configuration page. The top navigation bar includes the "Services" menu and a search bar. The main form has a title "Create bucket" with an "Info" link. It contains two sections: "General configuration" and "Object Ownership". In the "General configuration" section, the "AWS Region" dropdown is set to "Asia Pacific (Mumbai) ap-south-1". The "Bucket name" field is filled with "group-\$bucket". A note below states: "Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming." There's also a "Choose bucket" button and a note about copy settings. In the "Object Ownership" section, there's a note about controlling ownership of objects from other AWS accounts via ACLs. The bottom of the page includes standard AWS footer links like CloudShell, Feedback, and a copyright notice for 2024. The status bar at the bottom right shows the date and time as 3/22/2024 4:40 PM.

Step 2: Configure whether ACLs (Access Control Lists) are enabled for the bucket.
You can make the bucket accessible to all or restrict access based on your requirements.

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership
Bucket owner enforced

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG US 4:40 PM 3/22/2024

Step 3: Selecting SSE with Amazon S3 managed keys and Enabling bucket key.

Encryption type Info

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see DSSE-KMS pricing on the Storage tab of the [Amazon S3 pricing page](#).

Bucket Key
Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable
 Enable

Advanced settings

After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel **Create bucket**

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG US 4:40 PM 3/22/2024

Step 4: After creating bucket, upload the file in bucket.

The screenshot shows the AWS S3 console interface. At the top, there's a green success message: "Upload succeeded" with a link to "View details below.". Below this, there are two sections: "Destination" and "Succeeded". The "Destination" section shows "s3://group-5bucket" with "1 file, 1.0 MB (100.00%)". The "Succeeded" section shows "0 files, 0 B (0%)". Underneath these, there are tabs for "Files and folders" and "Configuration", with "Files and folders" being selected. A table titled "Files and folders (1 Total, 1.0 MB)" lists one file: "employee_p..." which is a "text/csv" file of size "1.0 MB" with a status of "Succeeded". The bottom of the screen shows the Windows taskbar with various pinned icons and system status indicators.

IAM ROLE ASSOCIATION

Step 1: Navigate the IAM console and create a role, choose redshift that will assume the role and then choose S3 as the service that will use this role.

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Select trusted entity

Trusted entity type

- AWS service

Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account

Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- Web identity

Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- SAML 2.0 federation

Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy

Create a custom trust policy to enable others to perform actions in this account.

others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

Redshift

Choose a use case for the specified service.

Use case

- Redshift - Customizable

Allows Redshift clusters to call AWS services on your behalf.
- Redshift

Allows Redshift clusters to call AWS services on your behalf.
- Redshift - Scheduler

Allow Redshift Scheduler to call Redshift on your behalf.

Cancel Next

Permissions policies (1/913) [Info](#)

Choose one or more policies to attach to your new role.

Filter by Type All types 9 matches

Policy name	Type	Description
<input type="checkbox"/> AmazonDMSRedsh...	AWS managed	
<input checked="" type="checkbox"/> AmazonS3FullAccess	AWS managed	
<input type="checkbox"/> AmazonS3ObjectL...	AWS managed	
<input type="checkbox"/> AmazonS3Outpost...	AWS managed	
<input type="checkbox"/> AmazonS3Outpost...	AWS managed	
<input type="checkbox"/> AmazonS3ReadOn...	AWS managed	
<input type="checkbox"/> AWSBackupService...	AWS managed	

Step 2: Provide the details and finally create the role.

IAM > Roles > Create role

Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.

Description
Add a short explanation for this role.

Step 1: Select trusted entities [Edit](#)

Step 1: Select trusted entities

Trust policy

```

1  {
2    "Version": "2012-10-17",
3    "Statement": [
4      {
5        "Effect": "Allow",
6        "Action": [
7          "sts:AssumeRole"
8        ],
9        "Principal": [
10          {
11            "Service": [
12              "redshift.amazonaws.com"
13            ]
14          }
15        ]
16      }
  
```

Step 2: Add permissions

CloudShell Feedback Search © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG US 4:44 PM 3/22/2024

Permissions policy summary

Policy name	Type	Attached as
AmazonS3FullAccess	AWS managed	Permissions policy

Step 3: Add tags

Add tags - optional [Info](#)
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

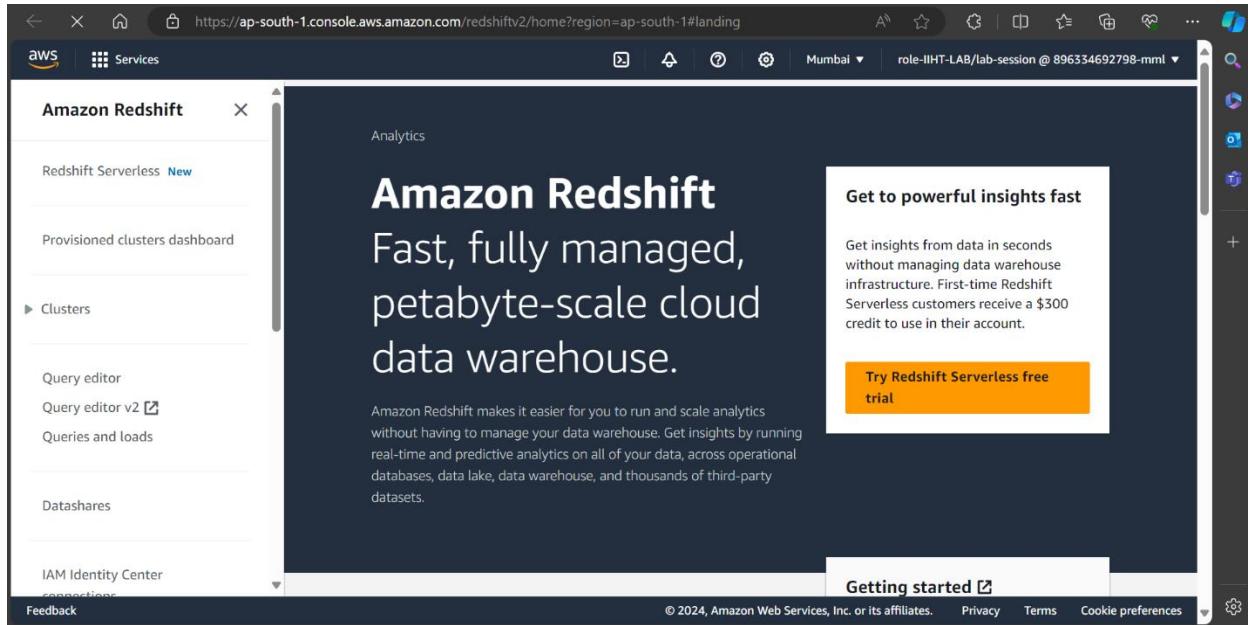
Add new tag
You can add up to 50 more tags.

Cancel Previous Create role

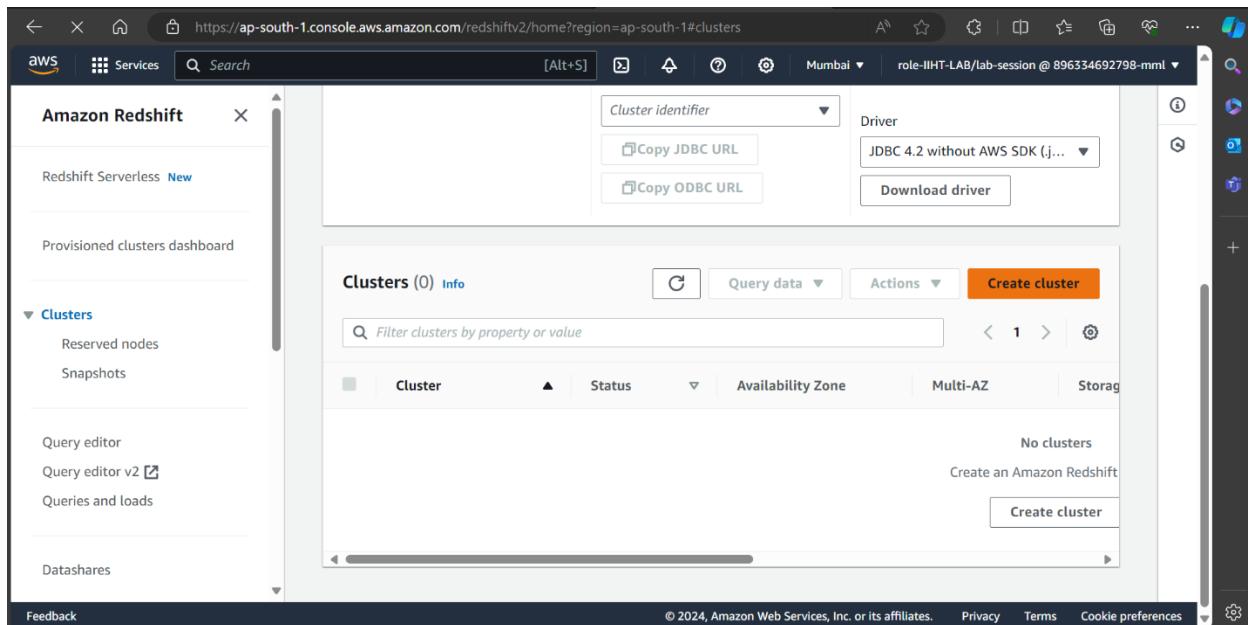
CloudShell Feedback Search © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG US 4:44 PM 3/22/2024

REDSHIFT CREATION

Step1: Sign in to the AWS Management Console and open the Amazon Redshift console and create the cluster. At upper right, choose the AWS Region where you want to create the cluster.



Choose **Clusters**, then choose **Create cluster** and start creating the cluster.



Step 2: In the Cluster configuration section, specify values for Cluster identifier and size of the cluster.

Amazon Redshift > Clusters > Create cluster

Create cluster Info

Looking for free trial? Try Redshift Serverless. First-time Redshift Serverless customers receive a \$300 credit to use in their account. [Launch Redshift Serverless](#)

Cluster configuration

Cluster identifier
This is the unique key that identifies a cluster.

The identifier must be from 1-63 characters. Valid characters are a-z (lowercase only) and - (hyphen).

Choose the size of the cluster
 I'll choose
 Help me choose

Node type Info
Choose a node type that meets your CPU, RAM, storage capacity, and drive type requirements.

Then choose the **Node type** and number of **Nodes** to size your cluster. Choose dc2.large Node type and 1 for Nodes.

Amazon Redshift > Clusters > Create cluster

Node type Info
Choose a node type that meets your CPU, RAM, storage capacity, and drive type requirements.

Number of nodes
Enter the number of nodes that you need.

Range (1-32)

Configuration summary Info
dc2.large | 1 node

\$229.95/month Estimated on-demand compute price Save more than 60% of your costs by purchasing reserved nodes. Learn more about pricing	160 GB Total compressed storage The total storage capacity for the cluster if you deploy the number of nodes that you chose.
---	---

Step 3: In the Database configuration section, specify a value for **Admin username**. Choose Admin password as follows:

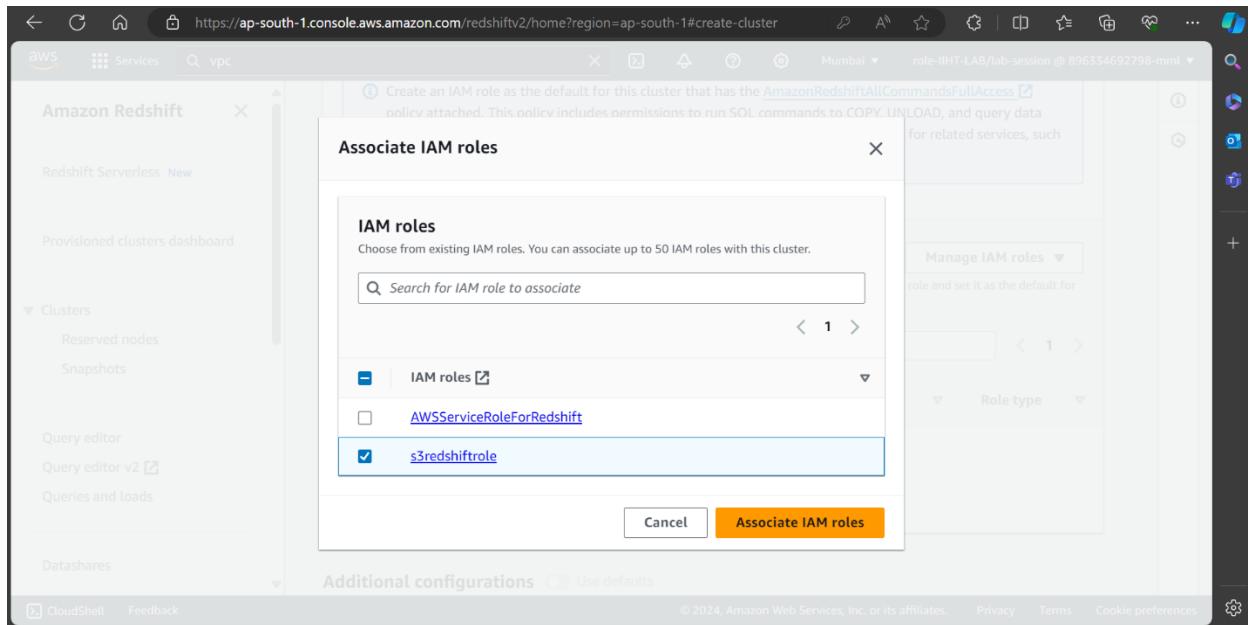
Manually add an admin password (Use your own password).

The screenshot shows the AWS Redshift console with the URL <https://ap-south-1.console.aws.amazon.com/redshiftv2/home?region=ap-south-1#create-cluster>. The left sidebar shows 'Amazon Redshift' selected. The main area is titled 'Database configurations' and contains fields for 'Admin user name' (set to 'awsuser') and 'Admin password' (set to 'Manually add the admin password' with password 'Group5').

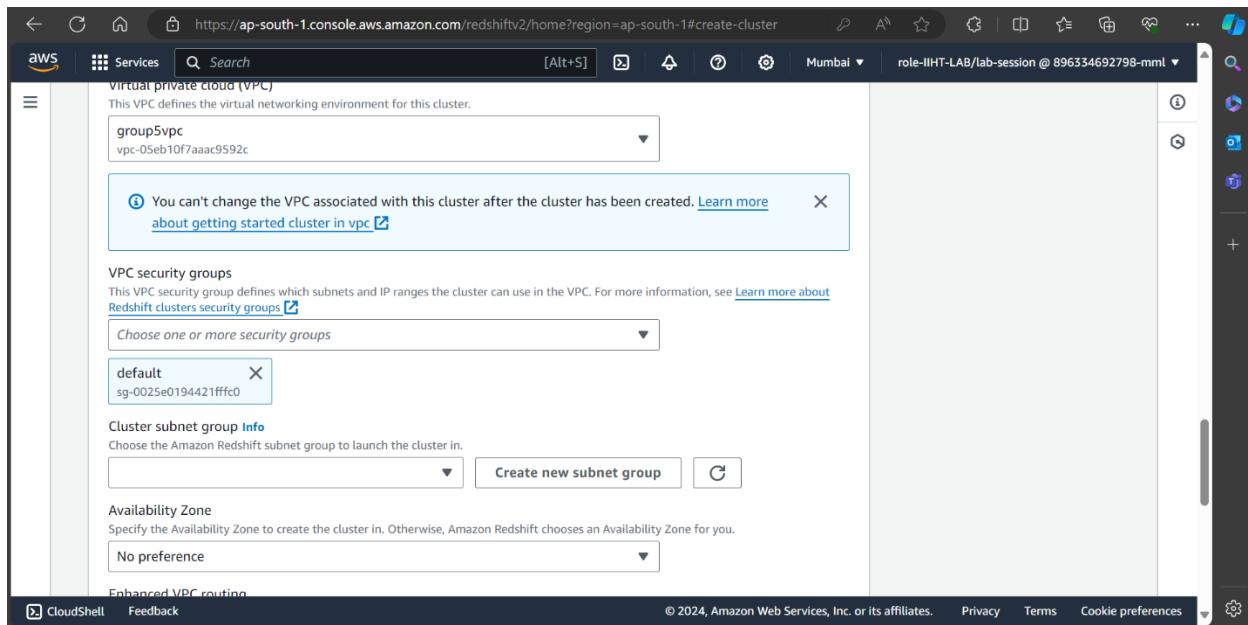
Step 4: Create an IAM role and choose the one you have created for your cluster. Under Cluster permissions, for Manage IAM roles.

The screenshot shows the AWS Redshift console with the URL <https://ap-south-1.console.aws.amazon.com/redshiftv2/home?region=ap-south-1#create-cluster>. The left sidebar shows 'Amazon Redshift' selected. The main area shows a callout box about the 'AmazonRedshiftAllCommandsFullAccess' policy and the 'Associated IAM roles (0)' section, which includes a search bar and a table with 'No resources' and an 'Associate IAM role' button.

Specify an Amazon S3 bucket for the IAM role to access then Associate IAM role.



Step 5: Choose the VPC that you have created previously and set the VPC security groups as default.



Step 6: Create a cluster subnet group. A cluster subnet group allows you to specify a set of subnets in your VPC.

The screenshot shows the 'Create cluster subnet group' page in the AWS Redshift console. The 'Cluster subnet group details' section is active, showing fields for 'Name' (cluster-subnet-group-1) and 'Description' (subnetforcluster). Below this is the 'Add subnets' section, which is currently inactive. At the bottom, there are 'CloudShell' and 'Feedback' links, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

The screenshot shows the 'Create cluster subnet group' configuration page. In the 'VPC' section, 'vpc-05eb10f7aac9592c' is selected from a dropdown. Below it, there's a button to 'Add all the subnets for this VPC'. The 'Availability Zone' and 'Subnet' selection dropdowns are shown. Under 'Subnets in this cluster subnet group (1)', a table lists one subnet: 'ap-south-1a' with 'subnet-00b9c4...' and CIDR block '172.16.1.0/24'. There are 'Remove' and 'Remove all' buttons for this list. At the bottom are 'Cancel' and 'Create cluster subnet group' buttons, with the latter being orange.

Step 7: Choose the Availability Zone in which location should subnet run.

This screenshot shows the AWS Redshift VPC configuration page. It displays the following settings:

- VPC:** A dropdown menu shows "group5vpc" and "vpc-05eb10f7aac9592c". A note states: "You can't change the VPC associated with this cluster after the cluster has been created." with links to "Learn more about getting started cluster in vpc".
- VPC security groups:** A dropdown menu shows "Choose one or more security groups" and "default" (sg-0025e0194421fffc0).
- Cluster subnet group:** A dropdown menu shows "cluster-subnet-group-5" and a "Create new subnet group" button.
- Availability Zone:** A dropdown menu shows "ap-south-1".
- Enhanced VPC routing:** A dropdown menu shows "CloudShell" and "Feedback".

Turn on “publicly accessible”. This allows public connection to redshift.

This screenshot shows the AWS Redshift cluster configuration page under the "Turn on" section. It displays the following settings:

- Publicly accessible:** A checkbox labeled "Turn on Publicly accessible" is checked, with a note: "Allow public connections to Amazon Redshift."
- Elastic IP address:** A dropdown menu is shown.
- Note:** A message states: "It can take about ten minutes for the setting to change and connections to succeed."
- Database configurations:** A section with a "Database configurations" link.
- Maintenance:** A section with a "Maintenance" link.
- Monitoring:** A section with a "Monitoring" link.
- CloudShell:** A "CloudShell" link at the bottom left.

After completing the configuration setup “Create the cluster”

The screenshot shows the AWS Redshift console with the URL <https://ap-south-1.console.aws.amazon.com/redshiftv2/home?region=ap-south-1#clusters>. The top navigation bar includes the AWS logo, Services, Search, and Mumbai region. The main content area displays a cluster named "redshift-cluster-gr5" with the status "Available". Below the cluster list is a "Clusters (1) Info" section with a "Create cluster" button. The bottom of the screen shows standard AWS footer links for CloudShell, Feedback, and legal information.

Step 8: Edit inbound rules under security groups and grant inbound access to cluster.

To access from an Amazon EC2 external, add a rule to the security group attached to your cluster that allows inbound traffic.

The screenshot shows the AWS EC2 console with the URL <https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#SecurityGroup:groupId=sg-0025e0194421fffc0>. The left sidebar lists various EC2 services like Dashboard, Global View, Instances, Launch Templates, and Images. The main content area shows a security group named "sg-0025e0194421fffc0 - default" with its details: Security group name is "default", Owner is "896334692798", and it has 1 inbound rule entry. The "Inbound rules" tab is selected. The bottom of the screen shows standard AWS footer links.

The screenshot shows the AWS EC2 Inbound Security Group Rules page. It lists three rules:

- sgr-0f6f7a2e95d01ac22**: Type: All traffic, Protocol: All, Port range: All, Source: sg-0025e0194421fff, Description: c0. This rule is highlighted with a blue box.
- : Type: SSH, Protocol: TCP, Port range: 22, Source: 0.0.0.0/0. This rule is marked with a red X.
- : Type: Redshift, Protocol: TCP, Port range: 5439, Source: 0.0.0.0/0. This rule is marked with a red X.

At the bottom left is an "Add rule" button. A yellow warning bar at the bottom states: "⚠️ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP." At the bottom right are links for CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

Step 9: After creating your cluster, you can immediately run queries using the Amazon Redshift console.

To query databases through Amazon Redshift cluster,

- Connect to your cluster and run queries on the AWS Management Console with one of the query editors.
- Connect to your cluster through an SQL client tool, such as SQL Workbench

Connection to the EC2 instance

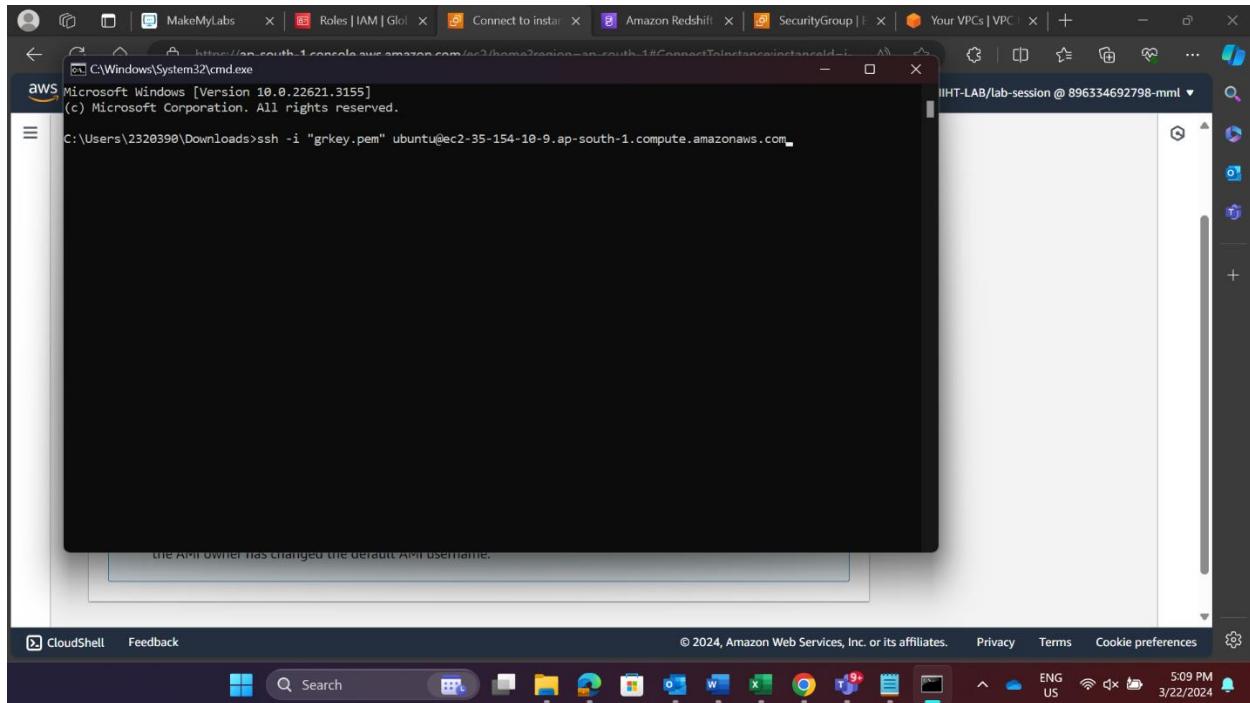
Step 1: Goto your EC2 dashboard and select the instance and click on Connect.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations (New), and Images (AMIs, AMI Catalog). The main area displays the 'Instance summary for i-07056ba2dcc970604 (group5ec2)'. It includes fields for Instance ID (i-07056ba2dcc970604), Public IPv4 address (35.154.10.9), Private IPv4 addresses (172.16.1.18), IPv6 address (-), Instance state (Running), Hostname type (IP name: ip-172-16-1-18.ap-south-1.compute.internal), Private IP DNS name (IPv4 only) (ip-172-16-1-18.ap-south-1.compute.internal), Answer private resource DNS name (-), Instance type (t2.micro), VPC ID (vpc-05eb10f7aaac9592c), Elastic IP addresses (-), Auto-assigned IP address (35.154.10.9 [Public IP]), and VPC Compute Optimizer finding (Opt-in to AWS Compute Optimizer for recommendations). At the top, there are 'Connect' and 'Actions' buttons. The 'Connect' button is highlighted in blue.

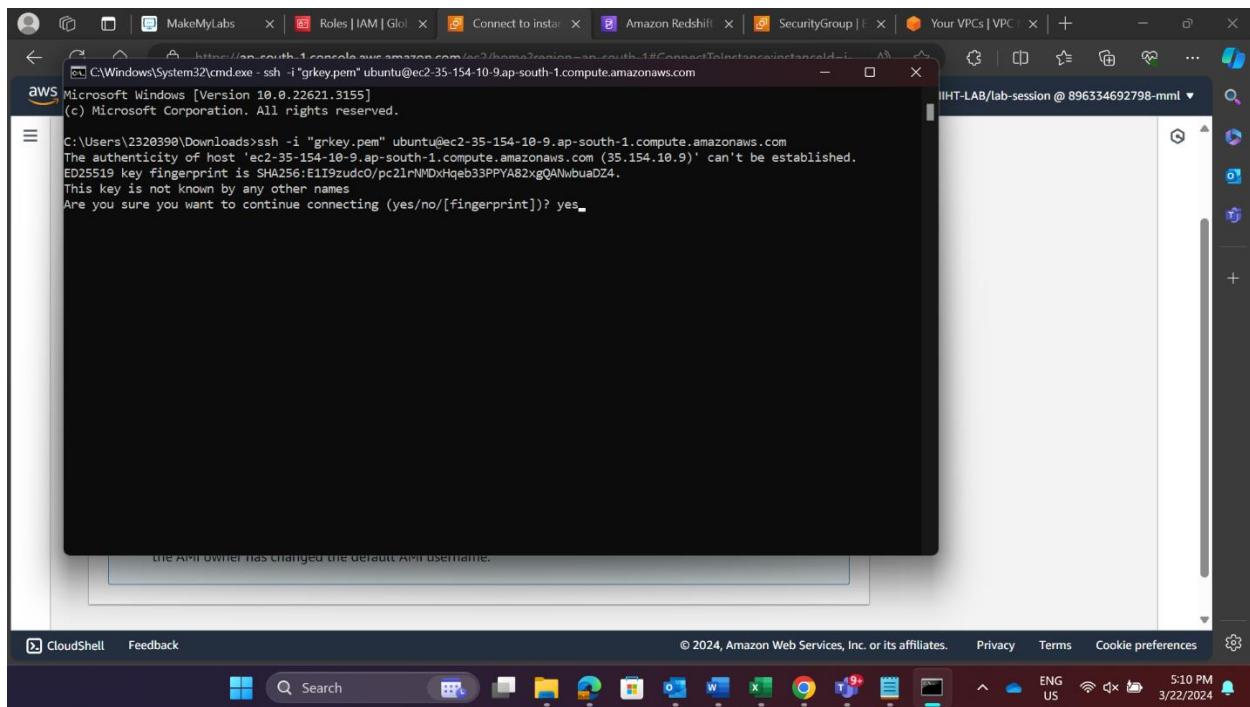
Step 2: Use SSH Client option to connect to your instance and copy the command given as an example.

The screenshot shows the AWS EC2 Instance Connect page. The top navigation bar includes tabs for EC2 Instance Connect, Session Manager, SSH client (which is selected and highlighted in blue), and EC2 serial console. Below the tabs, it shows the Instance ID (i-07056ba2dcc970604). A numbered list of steps for connecting via SSH is displayed: 1. Open an SSH client, 2. Locate your private key file (grkey.pem), 3. Run the command chmod 400 "grkey.pem", and 4. Connect to your instance using its Public DNS (ec2-35-154-10-9.ap-south-1.compute.amazonaws.com). A callout bubble indicates that the command has been copied to the clipboard. Below the list, a note states: 'Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.' The bottom of the screen shows a Windows taskbar with various icons.

Step 3: Open a terminal or the command prompt on the local machine and paste the command copied in the previous step. Make sure that you are inside the directory where your key pair generated during the instance creation is stored before executing the command.



A screenshot of a Windows desktop environment. The taskbar at the bottom shows several open applications, including a CloudShell window, a Feedback window, a search bar, and various system icons like battery level, signal strength, and volume. Above the taskbar, there are multiple browser tabs from the AWS Management Console. One tab is active, displaying a terminal window titled 'C:\Windows\System32\cmd.exe'. The command entered is 'ssh -i "grkey.pem" ubuntu@ec2-35-154-10-9.ap-south-1.compute.amazonaws.com'. A tooltip message 'The AWS Owner has changed the default API username.' is visible near the bottom left of the terminal window.



A screenshot of a Windows desktop environment, similar to the one above. The taskbar shows the same set of open applications. The terminal window in the foreground displays the following text:

```
C:\Users\2320390\Downloads>ssh -i "grkey.pem" ubuntu@ec2-35-154-10-9.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-35-154-10-9.ap-south-1.compute.amazonaws.com (35.154.10.9)' can't be established.
ED25519 key fingerprint is SHA256:E1I9zudc0/pclnNxDhqeB33PPYA82xgQANwbuadZ4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

The terminal window also shows the same tooltip message: 'The AWS Owner has changed the default API username.'

Step 4: Once the above command is successfully executed you will be logged into your ubuntu server. The next step is to install PostgreSQL on your virtual server by running command '**sudo apt-get install -y PostgreSQL-client**'

PostgreSQL is installed successfully on your ubuntu server.

```
ubuntu@ip-172-16-1-18: ~
System information as of Fri Mar 22 11:40:10 UTC 2024
Activity: 19 notifications
Chat: 3 notifications
Teams: 0 notifications
Calendar: 0 notifications
Calls: 0 notifications
Files: 0 notifications
Apps: 0 notifications

System load: 0.0 Processes: 98
Usage of /: 20.4% of 7.57GB Users logged in: 0
Memory usage: 20% IPv4 address for eth0: 172.16.1.18
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-16-1-18:~$ sudo apt-get install -y postgresql-client
```

Step 5: After successfully installing the PostgreSQL on your server, the next step is connecting to the redshift cluster. This is done by using the following command.

'psql -h redshift-cluster-1.cackughiw5hj.ap-south-1.redshift.amazonaws.com -p 5439 -U awsuser -d dev -W'

```
ubuntu@ip-172-16-1-18: ~
Unpacking libpq5:amd64 (14.11-0ubuntu0.22.04.1) ...
Selecting previously unselected package postgresql-client-common.
Preparing to unpack .../postgresql-client-common_238_all.deb ...
Unpacking postgresql-client-common (238) ...
Selecting previously unselected package postgresql-client-14.
Preparing to unpack .../postgresql-client-14_14.11-0ubuntu0.22.04.1_amd64.deb ...
Unpacking postgresql-client-14 (14.11-0ubuntu0.22.04.1) ...
Selecting previously unselected package postgresql-client.
Preparing to unpack .../postgresql-client_14+238_all.deb ...
Unpacking postgresql-client (14+238) ...
Setting up postgresql-client-common (238) ...
Setting up libpq5:amd64 (14.11-0ubuntu0.22.04.1) ...
Setting up postgresql-client-14 (14.11-0ubuntu0.22.04.1) ...
update-alternatives: using /usr/share/postgresql/14/man/man1/psql.1.gz to provide /usr/share/man/man1/psql.1.gz (psql.1.gz) in auto mode
Setting up postgresql-client (14+238) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

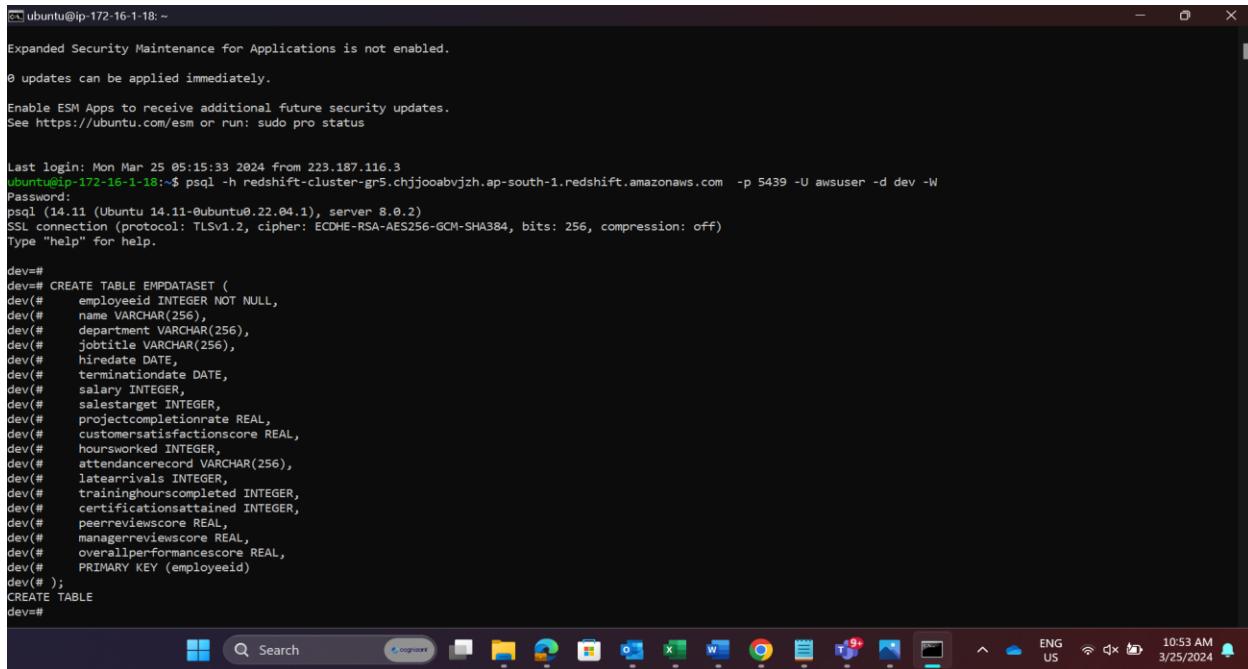
No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-16-1-18:~$ psql -h redshift-cluster-1.cackughiw5hj.ap-south-1.redshift.amazonaws.com -p 5439 -U awsuser -d dev -W
Password:
psql (14.11 (Ubuntu 14.11-0ubuntu0.22.04.1), server 8.0.1)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help.

dev#
```

Step 6: Once the above command is successfully executed, you can access your database. Now, create the required table (EMPDATASET) using the CREATE statement and define the column names and its attributes.

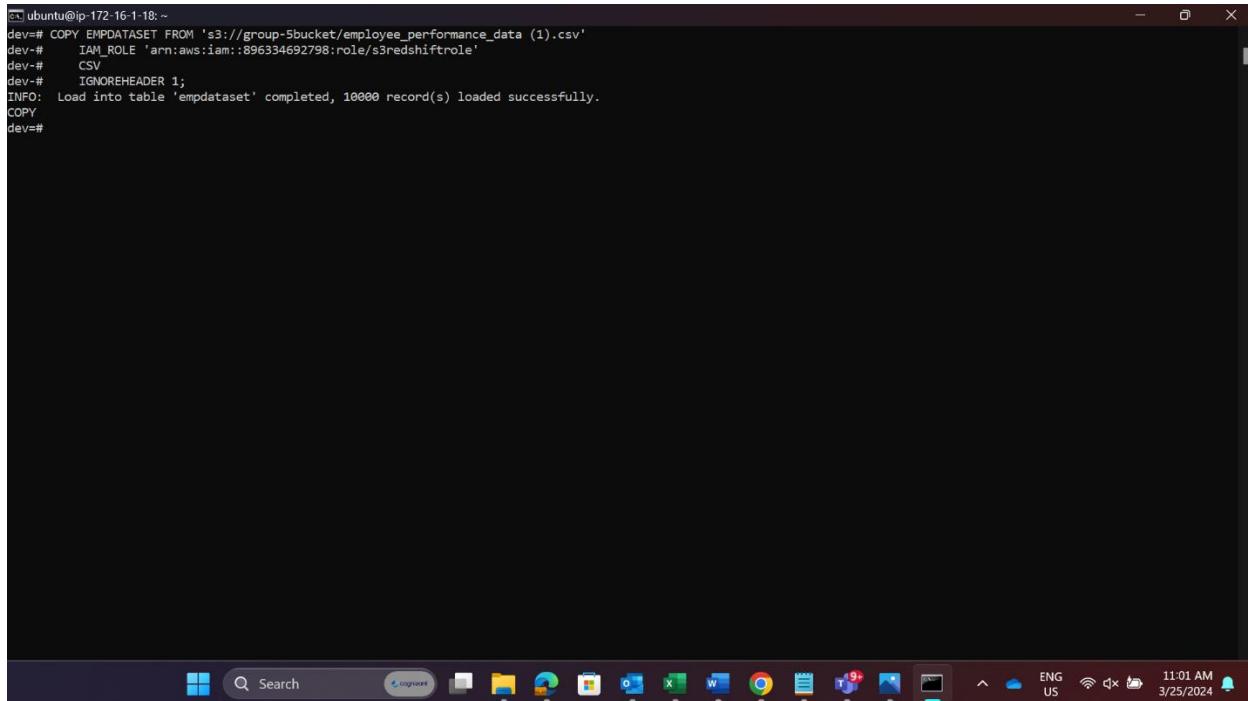


```
ubuntu@ip-172-16-1-18:~$ Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Mon Mar 25 05:33 2024 from 223.187.116.3
ubuntu@ip-172-16-1-18:~$ psql -h redshift-cluster-gr5.chjjooabvzh.ap-south-1.redshift.amazonaws.com -p 5439 -U awsuser -d dev -W
Password:
psql (14.11 (Ubuntu 14.11-0ubuntu0.22.04.1), server 8.0.2)
SSL connection (protocol: TLSV1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help.

dev# CREATE TABLE EMPDATASET (
dev#   employeeid INTEGER NOT NULL,
dev#   name VARCHAR(256),
dev#   department VARCHAR(256),
dev#   jobtitle VARCHAR(256),
dev#   hiredate DATE,
dev#   terminationdate DATE,
dev#   salary INTEGER,
dev#   salestarget INTEGER,
dev#   projectcompletionrate REAL,
dev#   customersatisfactionscore REAL,
dev#   hoursworked INTEGER,
dev#   attendancerecord VARCHAR(256),
dev#   latearrivals INTEGER,
dev#   traininghourscompleted INTEGER,
dev#   certificationsattained INTEGER,
dev#   peerreviewscore REAL,
dev#   managerreviewscore REAL,
dev#   overallperformance score REAL,
dev#   PRIMARY KEY (employeeid)
dev# );
CREATE TABLE
dev#
```

Step 7: The table has been created successfully. Now the next step is to load the data present in your S3 bucket into the table you have created in the redshift cluster. This is done by using the copy command.



```
ubuntu@ip-172-16-1-18:~$ COPY EMPDATASET FROM 's3://group-5bucket/employee_performance_data (1).csv'
dev#   IAM_ROLE 'arn:aws:iam::896334692798:role/s3redshiftrole'
dev#   CSV
dev#   IGNOREHEADER 1;
INFO: Load into table 'empdataset' completed, 10000 record(s) loaded successfully.
COPY
dev#
```

The data has been successfully loaded into the table.

Redshift Queries Generation and Visualization

1. Fetch the sales target, customer satisfaction ratings, and productivity for all the employees of the sales department.

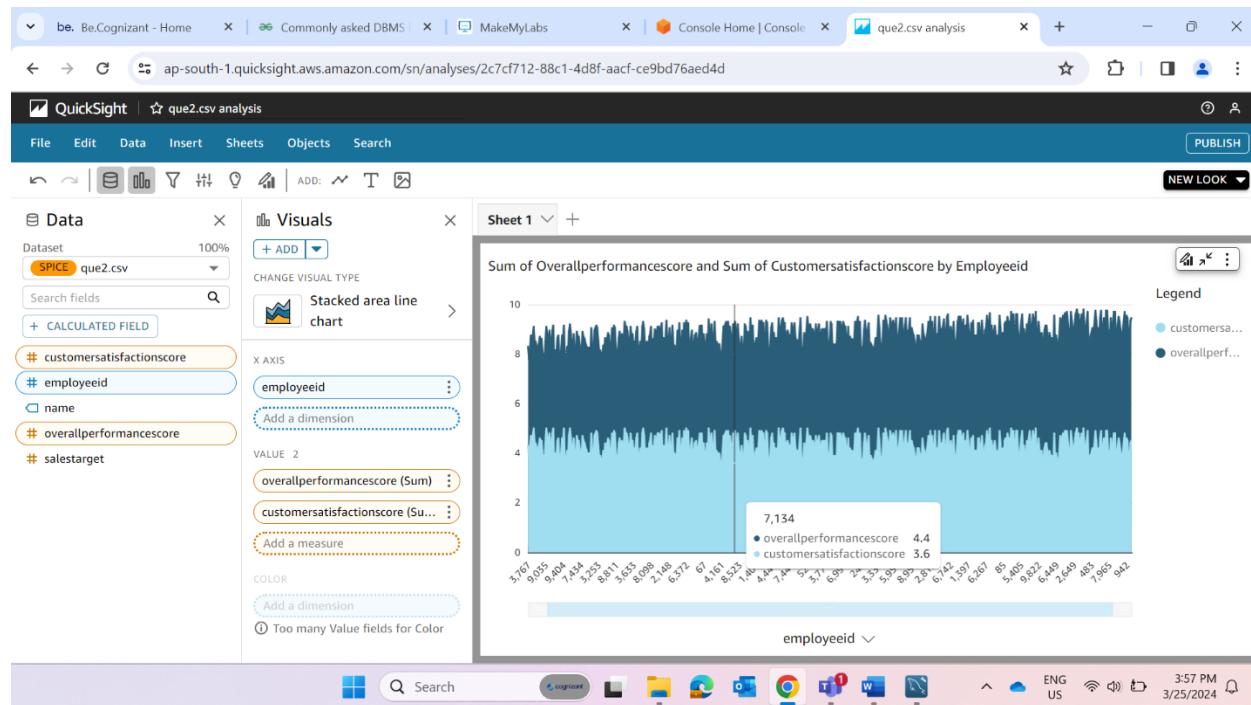
 - a. Query execution.

```

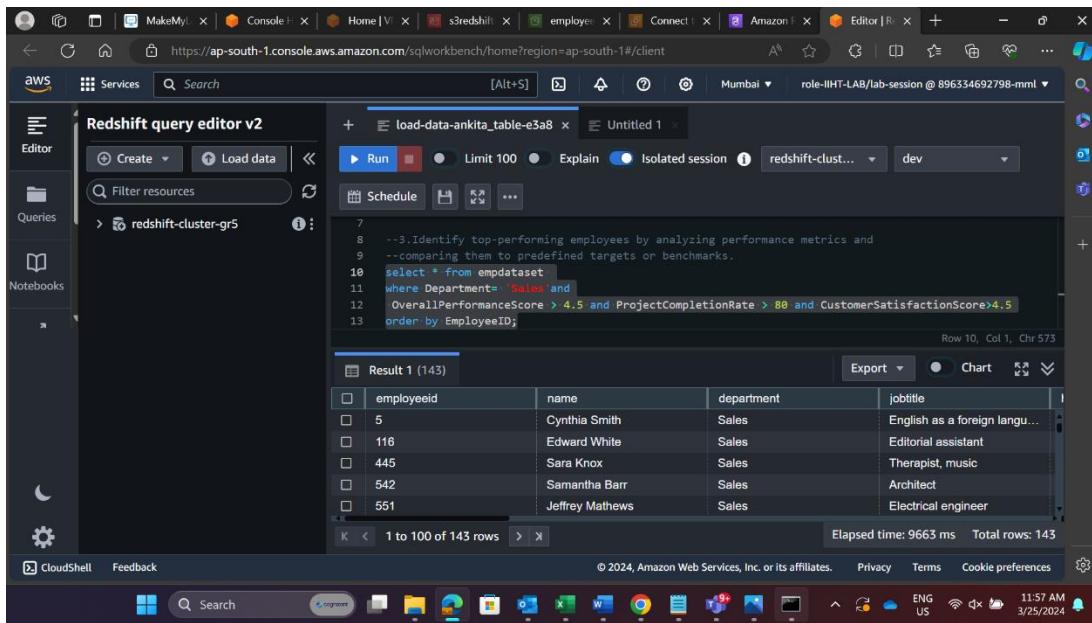
aws Services Search [Alt+S] Mumbai role-IIHT-LAB/lab-session @ 896354692798-mm1
Redshift query editor v2
+ load-data-ankita_table-e3a8 x Untitled 1
[Run] Limit 100 Explain Isolated session redshift-clust... dev
Filter resources
redshift-cluster-gr5
awsdatasatalog dev public
empdataset
Field employeeid name salestarget customersatisfactionsc...
employeeid integer character varying(2) character varying(2)
name character varying(2)
department character varying(2)
1 to 100 of 1922 rows
Elapsed time: 27 ms Total rows: 1922
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences FNG US 11:16 AM 3/25/2024

```

- b. Visualization



2. Identify top-performing employees by analyzing performance metrics and comparing them to predefined targets or benchmarks.
- a. Query Execution

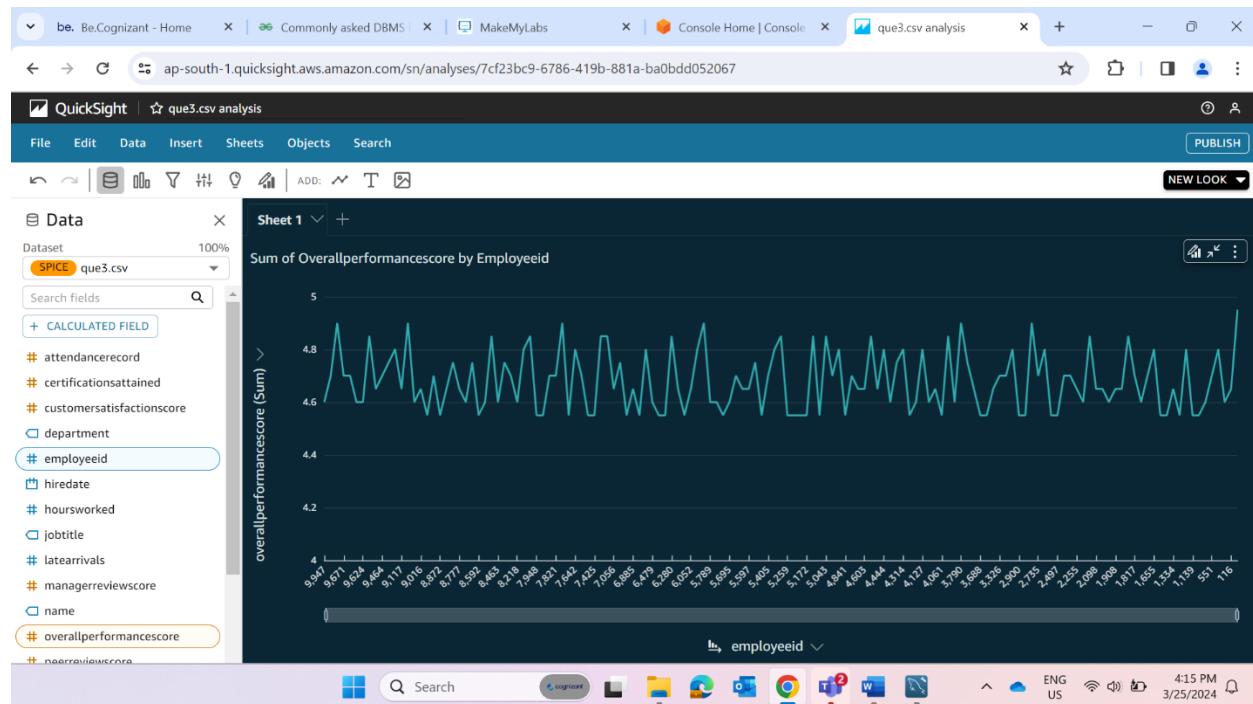


```

    7
    8 --3. Identify top-performing employees by analyzing performance metrics and
    9 --comparing them to predefined targets or benchmarks.
10 select * from empdataset
11 where Department='Sales' and
12 OverallPerformanceScore > 4.5 and ProjectCompletionRate > 80 and CustomerSatisfactionScore>4.5
13 order by EmployeeID;
  
```

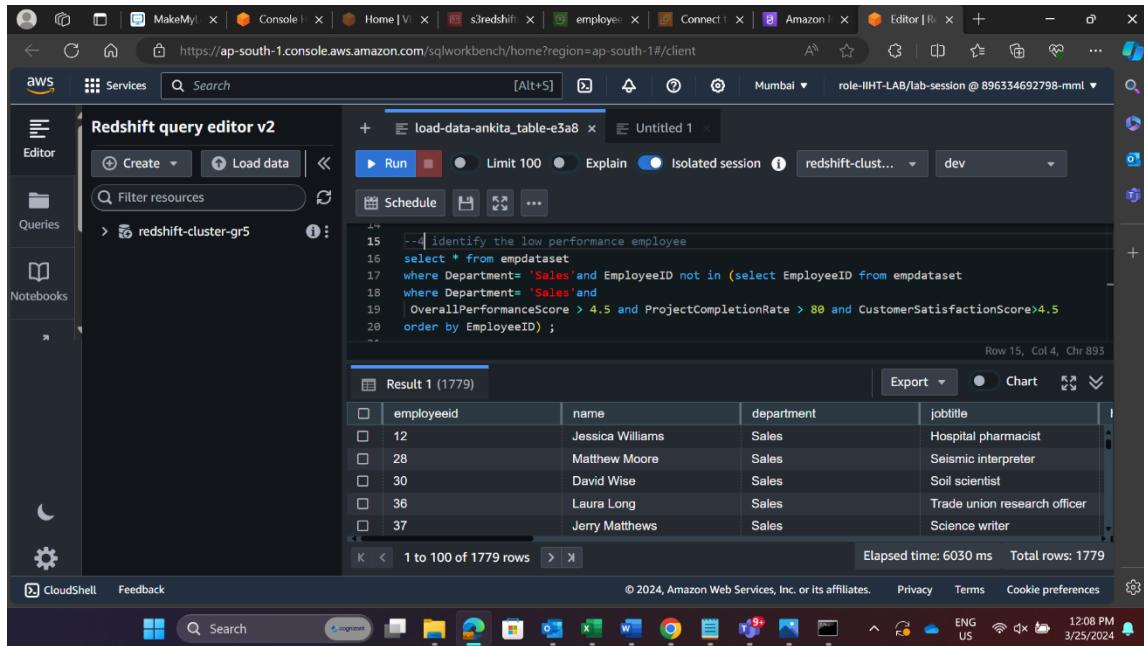
The screenshot shows the AWS Redshift query editor interface. A SQL query is being run to select all columns from the 'empdataset' table where the department is 'Sales', the overall performance score is greater than 4.5, the project completion rate is greater than 80, and the customer satisfaction score is greater than 4.5. The results are ordered by EmployeeID. The output table has columns: employeeid, name, department, and jobtitle. The results show 143 rows of data, with the first few rows being: employeeid 5 (Cynthia Smith, Sales, English as a foreign language teacher), employeeid 116 (Edward White, Sales, Editorial assistant), employeeid 445 (Sara Knox, Sales, Therapist, music), employeeid 542 (Samantha Barr, Sales, Architect), and employeeid 551 (Jeffrey Mathews, Sales, Electrical engineer).

b. Visualization



3. Identify the employees having low performance in the sales department.

a. Query Execution



```

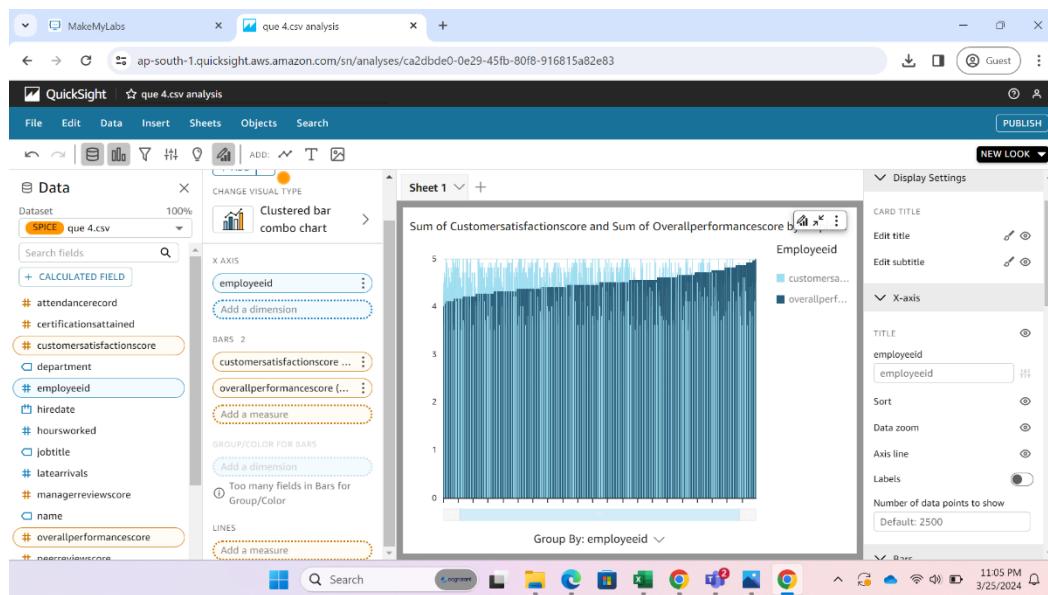
14
15  --4 identify the low performance employee
16  select * from empdataset
17  where Department= 'Sales'and EmployeeID not in (select EmployeeID from empdataset
18  where Department= 'Sales'and
19  | OverallPerformanceScore > 4.5 and ProjectCompletionRate > 80 and CustomerSatisfactionScore>4.5
20  order by EmployeeID) ;

```

The screenshot shows the AWS Redshift query editor interface. A query is being run to select all columns from the 'empdataset' table where the department is 'Sales'. It filters out employees whose EmployeeID is present in another subquery that selects EmployeeIDs from the same table where the department is 'Sales' and both the OverallPerformanceScore and ProjectCompletionRate are greater than 4.5, and the CustomerSatisfactionScore is also greater than 4.5. The results are ordered by EmployeeID. The result set contains 1779 rows, with the first few entries being:

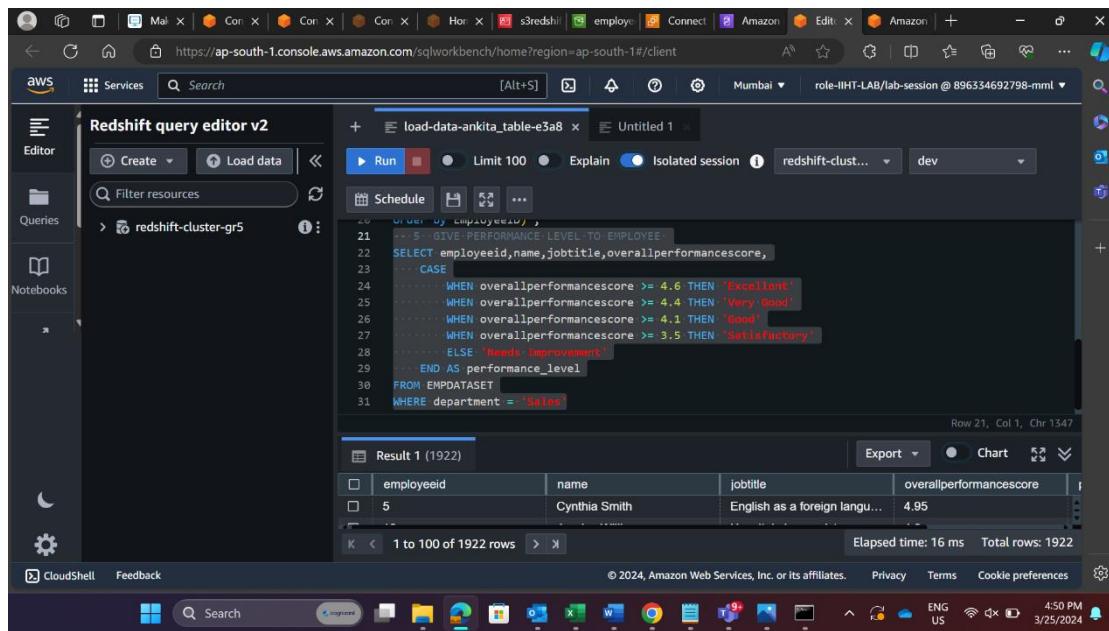
employeeid	name	department	jobtitle
12	Jessica Williams	Sales	Hospital pharmacist
28	Matthew Moore	Sales	Seismic interpreter
30	David Wise	Sales	Soil scientist
36	Laura Long	Sales	Trade union research officer
37	Jerry Matthews	Sales	Science writer

b. Visualization



4. Give a performance value to all the employees of the sales department based on some specified criteria.

a. Query Execution

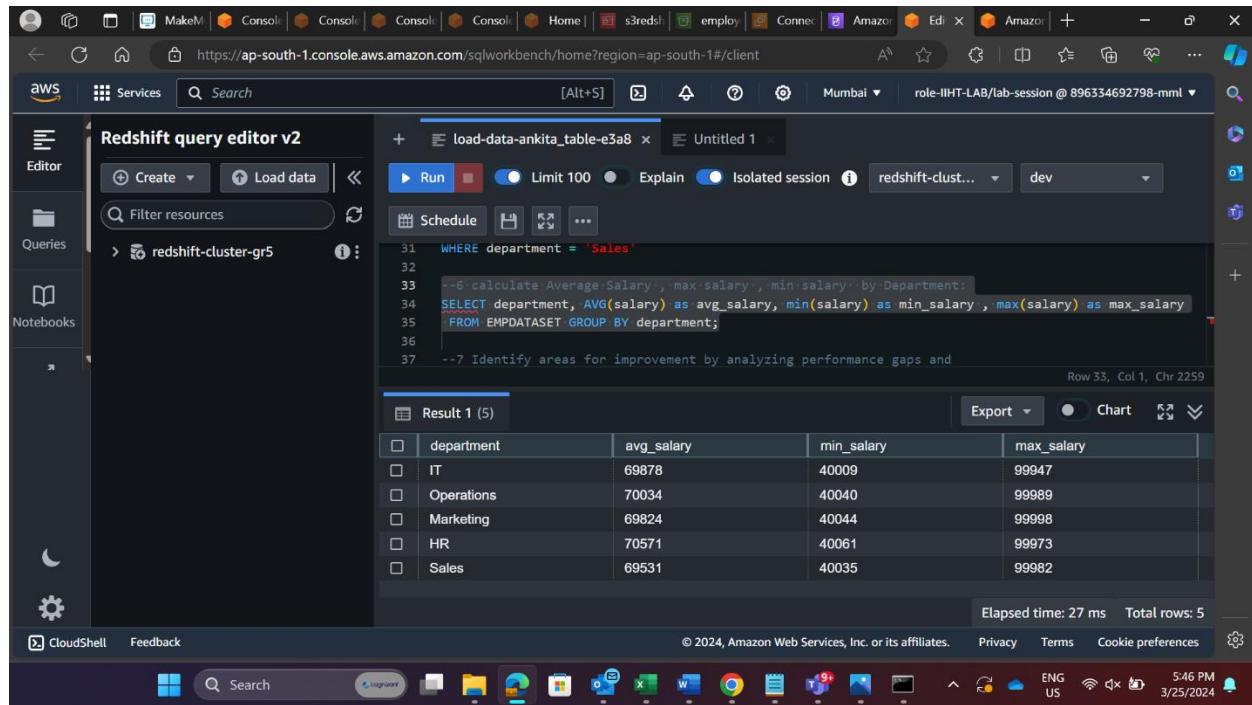


```
21 5 GIVE PERFORMANCE LEVEL TO EMPLOYEE
22 SELECT employeeid, name, jobtitle, overallperformancescore,
23     CASE
24         WHEN overallperformancescore >= 4.6 THEN 'Excellent'
25         WHEN overallperformancescore >= 4.4 THEN 'Very Good'
26         WHEN overallperformancescore >= 4.1 THEN 'Good'
27         WHEN overallperformancescore >= 3.5 THEN 'Satisfactory'
28     ELSE 'Needs Improvement'
29     END AS performance_level
30 FROM EMPTDATASET
31 WHERE department = 'Sales'
```

employeeid	name	jobtitle	overallperformancescore
5	Cynthia Smith	English as a foreign langu...	4.95

5. Calculate the average, minimum and maximum salary of each department in the employee table.

a. Query Execution

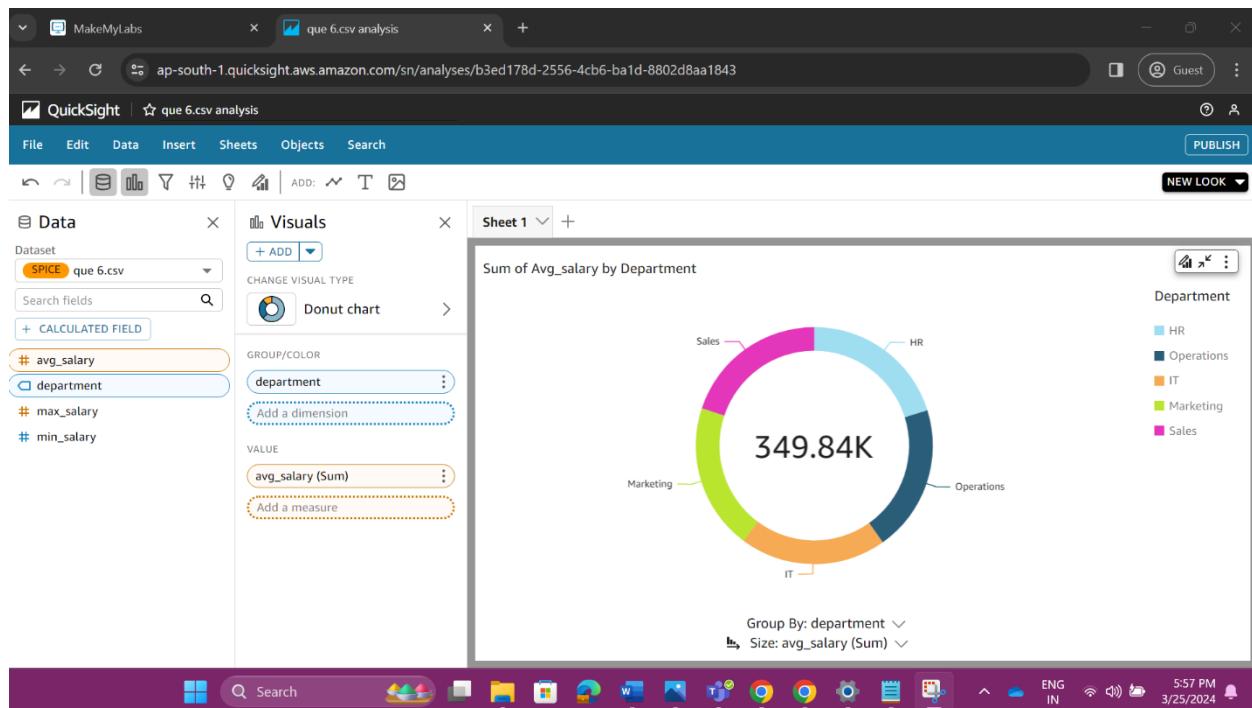


The screenshot shows the AWS Redshift query editor v2 interface. A query is being run to calculate the average, minimum, and maximum salary for each department. The results are displayed in a table:

department	avg_salary	min_salary	max_salary
IT	69878	40009	99947
Operations	70034	40040	99989
Marketing	69824	40044	99998
HR	70571	40061	99973
Sales	69531	40035	99982

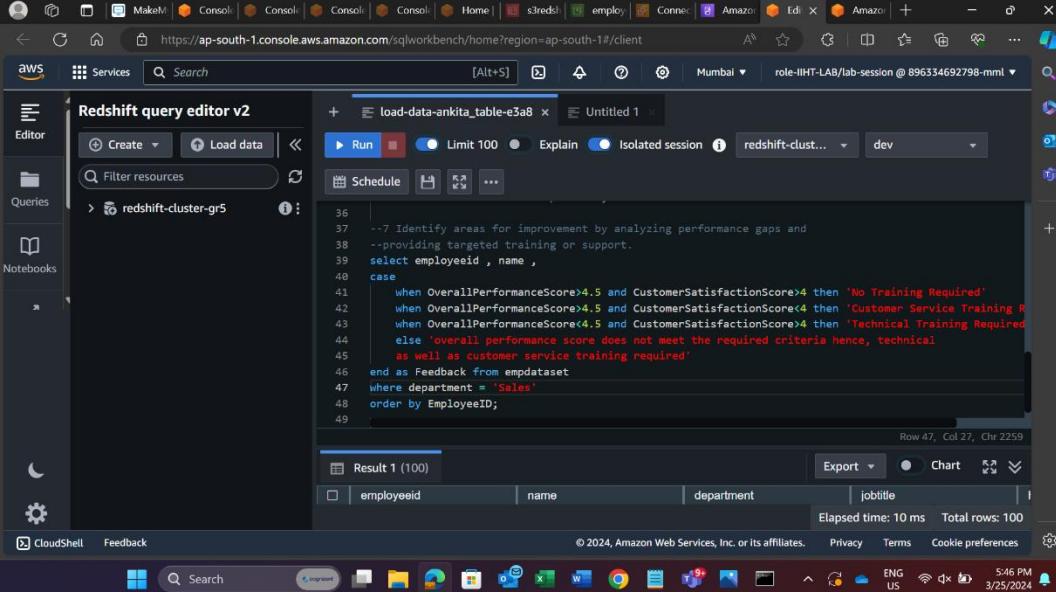
Elapsed time: 27 ms Total rows: 5

b. Visualization



6. Identify areas for improvement by analyzing performance gaps and providing targeted training or support.

a. Query Execution



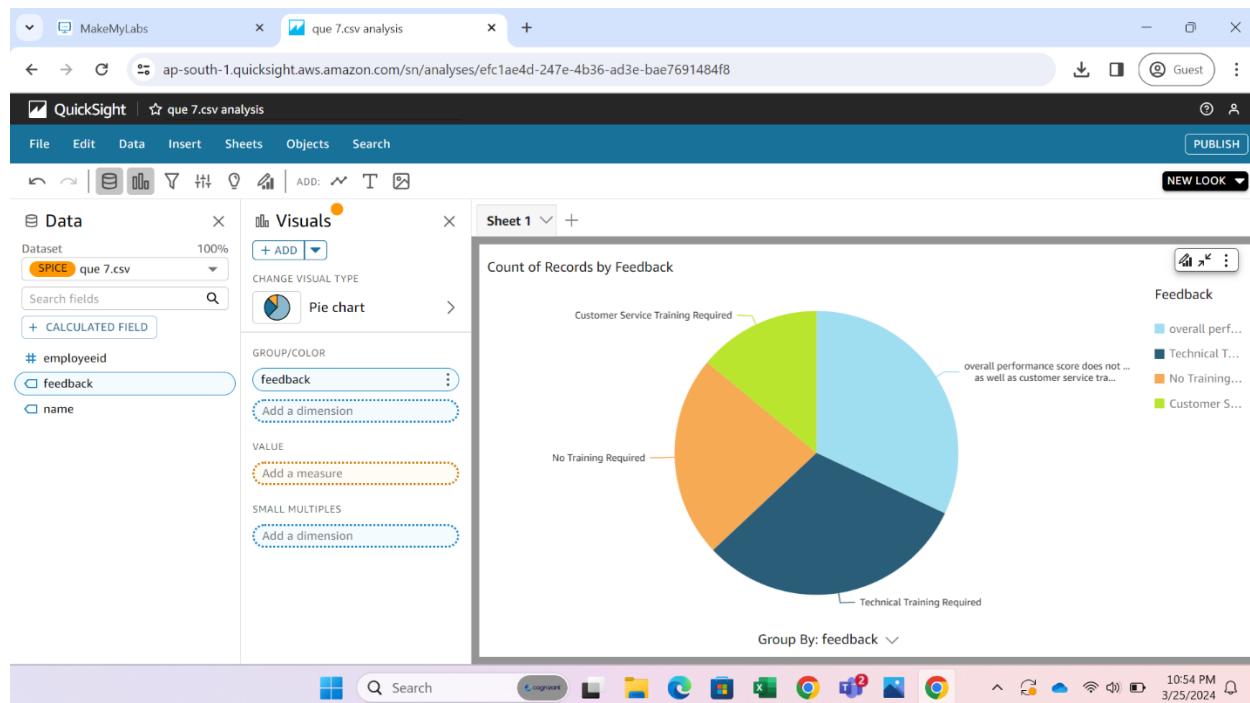
```

36 -- 7 Identify areas for improvement by analyzing performance gaps and
37 -- providing targeted training or support.
38 select employeeid , name ,
39 case
40     when OverallPerformanceScore>4.5 and CustomerSatisfactionScore>4 then 'No Training Required'
41     when OverallPerformanceScore<4.5 and CustomerSatisfactionScore<4 then 'Customer Service Training Required'
42     when OverallPerformanceScore<4.5 and CustomerSatisfactionScore>4 then 'Technical Training Required'
43     else 'overall performance score does not meet the required criteria hence, technical
44         as well as customer service training required'
45 end as Feedback from empdataset
46 where department = 'Sales'
47 order by EmployeeID;
48
49

```

The screenshot shows the AWS Redshift query editor interface. The code in the editor is a SQL script designed to identify performance gaps and provide targeted training based on overall performance and customer satisfaction scores. The results are displayed in a table titled 'Result 1 (100)' with columns: employeeid, name, department, jobtitle. The elapsed time for the query is 10 ms, and it processed 100 rows.

b. Visualization



CONCLUSION

In conclusion, the project exemplifies the seamless integration and powerful capabilities of AWS services in constructing a robust data infrastructure. By harnessing Amazon S3 as a scalable storage solution, we ensured the efficient and secure handling of our datasets. The coupling of S3 with Amazon Redshift allowed for the establishment of a high-performance data warehousing environment, facilitating quick access and analysis of our structured data. Moreover, the utilization of Amazon EC2 instances enabled us to execute complex SQL operations, ensuring the manipulation and transformation of data according to our analytical needs.

Furthermore, the deployment of AWS Quicksight for visualization added a layer of insight to our data exploration process. With Quicksight's intuitive interface and comprehensive visualization options, we were able to craft informative reports and dashboards that provided stakeholders with clear and actionable insights. This not only enhanced our understanding of the underlying data but also empowered decision-makers to make informed choices based on the analyzed information.

In summary, the successful execution of this project underscores the value proposition of AWS as a leading cloud provider for data analytics initiatives. The combination of S3, Redshift, EC2, and Quicksight offered a cohesive and scalable solution that streamlined the entire data processing pipeline. As organizations continue to grapple with increasing volumes of data, leveraging AWS services provides a reliable and efficient means to extract actionable insights and drive business outcomes. Moving forward, the lessons learned from this project will serve as a foundation for future endeavors in harnessing the full potential of cloud-based data analytics solutions.