



nextwork.org

Visualise a Relational Database



Sanjana Tripathy

Result Grid									
empno	ename	job	manager	hiredate	salary	comm	deptno		
1	JOHNSON	ADMIN	6	1990-12-17 00:00:00	18000.00	NULL	4		
2	HARDING	MANAGER	9	1998-02-02 00:00:00	52000.00	300.00	3		
3	TAFT	SALES I	2	1999-01-02 00:00:00	25000.00	500.00	3		
4	HOLLOWAY	SALES I	2	1999-01-02 00:00:00	25000.00	500.00	3		
5	LINDBERG	TECH	6	1994-06-23 00:00:00	22500.00	1400.00	4		
6	GARFIELD	MANAGER	9	1993-05-01 00:00:00	54000.00	NULL	4		
7	POLK	TECH	6	1997-09-22 00:00:00	25000.00	NULL	4		
8	GRANT	ENGINEER	10	1997-03-30 00:00:00	32000.00	NULL	2		
9	JACKSON	CEO	NULL	1990-01-01 00:00:00	75000.00	NULL	4		
10	FILLMORE	MANAGER	9	1994-08-09 00:00:00	56000.00	NULL	2		
11	ADAMS	ENGINEER	10	1996-03-15 00:00:00	34000.00	NULL	2		
12	WASHBURN	ADMIN	6	1998-04-16 00:00:00	18000.00	NULL	4		
13	MONTROSE	ENGINEER	10	2000-12-03 00:00:00	30000.00	NULL	2		
14	ROOSEVELT	CPA	9	1995-10-12 00:00:00	35000.00	NULL	1		
15	REED	TECH	6	1995-01-01 00:00:00	22500.00	NULL	4		

A circular profile picture of a young woman with dark hair, wearing a pink top and blue pants, sitting on a blue chair.

Sanjana Tripathy
NextWork Student

nextwork.org

Introducing Today's Project!

What is Amazon RDS?

Amazon RDS is a managed relational database service that simplifies setup, operation, and scaling of databases in the cloud. It is useful because it automates backups, patching, and maintenance, allowing developers to focus on their applications.

How I used Amazon RDS in this project

In today's project, I used Amazon RDS to create a managed MySQL database, connect it securely to SQL Workbench for data entry, and integrate it with Amazon QuickSight for dynamic data visualization—all without managing database infrastructure.

One thing I didn't expect in this project was...

One thing I didn't expect in this project was how seamlessly Amazon QuickSight could connect with RDS once the security groups were configured—highlighting the importance of precise network settings in enabling smooth, secure data visualization.



Sanjana Tripathy
NextWork Student

nextwork.org

This project took me...

This project took me around 3 hours, including the time spent understanding new concepts, setting up resources from scratch, and documenting detailed hands-on notes for future reference.



In the first part of my project...

Creating a Relational Database

To create a relational DB in AWS, I used Amazon RDS. In the console, I chose "Easy Create," selected MySQL as the engine, Free tier instance, set the DB identifier to "dataquicksight", and configured credentials with a self-managed password.

The screenshot shows the 'Create database' wizard in the AWS RDS console. The 'Easy create' option is selected. The 'Engine type' section shows MySQL selected. The 'Edition' section shows 'MySQL Community'. The 'DB instance size' section shows 'Free tier' selected. The 'DB instance identifier' section shows 'dataquicksight'. The 'Master username' section shows 'admin'. The 'Credentials management' section shows 'Self managed'. At the bottom, there is a note about setting up an EC2 connection.

Create database

Choose a database creation method

Standard create You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy create Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Configuration

Engine type MySQL

Aurora (MySQL Compatible)

Aurora (PostgreSQL Compatible)

MySQL

PostgreSQL

MariaDB

Oracle

Microsoft SQL Server

Edition MySQL Community

DB instance size

Production db.t2.xlarge 2 vCPUs 12 GiB RAM 3000 IOPS 2.000 USD/hour

Dev/Test db.r3.large 2 vCPUs 8 GiB RAM 1600 IOPS 0.500 USD/hour

Free tier db.t2.micro 2 vCPUs 1 GiB RAM 200 IOPS 0.000 USD/hour

DB instance identifier
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
dataquicksight
The DB instance identifier is case-insensitive, but is stored as all lowercase (as in 'mydbinstance'). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Master username admin
1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager to manage your master user credentials.

Managed in AWS Secrets Manager - most secure AWS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

Auto generate password AWS generates a password for you, or you can specify your own password.

Master password Minimum characters: At least 8 printable ASCII characters. Can't contain any of the following symbols: / \ ^ \$

Password strength Minimum characters: At least 8 printable ASCII characters. Can't contain any of the following symbols: / \ ^ \$

Confirm master password

Set up EC2 connection - optional
You can also set up a connection to an EC2 instance after creating the database. Go to the database list page or the database details page, choose Actions, and then choose Set up to EC2 connection.

A circular profile picture of a young woman with dark hair, wearing a pink top and blue pants, sitting on a blue chair.

Sanjana Tripathy
NextWork Student

nextwork.org

Understanding Relational Databases

A relational database is a structured collection of data organized into tables with rows and columns. It uses relationships between tables to efficiently store, retrieve, and manage data using a language like SQL.

MySQL vs SQL

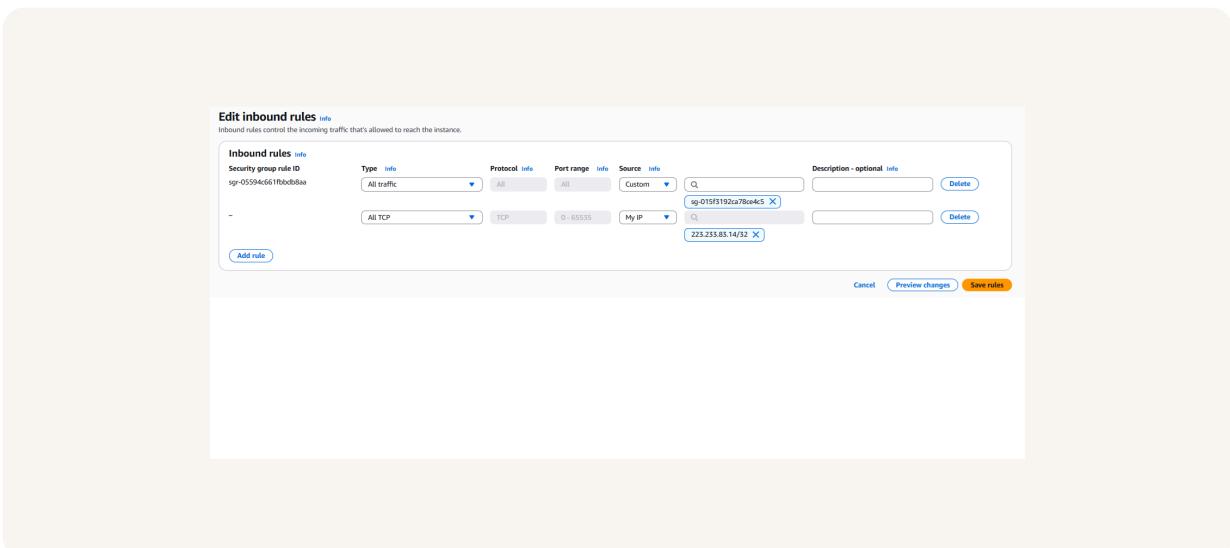
The difference is that SQL is a language used to query and manage data, while MySQL is a relational database management system (RDBMS) that uses SQL to interact with databases. SQL is the language; MySQL is the tool that implements it.



Populating my RDS instance

The first step I'll take is to make my RDS instance publicly accessible so it can be reached from outside the AWS network. This is essential for enabling a successful connection from MySQL Workbench on my local machine.

I had to update the default security group for my RDS schema to allow inbound traffic from my local machine's IP. This step ensures that my local system can securely connect to the RDS instance, even though it's already set to be publicly





Sanjana Tripathy
NextWork Student

nextwork.org

Using MySQL Workbench

empno	ename	job	manager	hiredate	salary	comm	department
1	JOHNSON	ADMIN	6	1990-12-17 00:00:00	18000.00	NULL	4
2	HARDING	MANAGER	9	1998-02-02 00:00:00	52000.00	300.00	3
3	TAFT	SALES I	2	1996-01-02 00:00:00	25000.00	500.00	3
4	HOOVER	SALES I	2	1990-04-02 00:00:00	27000.00	NULL	3
5	LINCOLN	TECH	6	1994-06-23 00:00:00	22500.00	1400.00	4
6	GARFIELD	MANAGER	9	1993-05-01 00:00:00	54000.00	NULL	4
7	POLK	TECH	6	1997-09-22 00:00:00	25000.00	NULL	4
8	GRANT	ENGINEER	10	1997-03-30 00:00:00	32000.00	NULL	2
9	JACKSON	CEO	NULL	1990-01-01 00:00:00	75000.00	NULL	4
10	FILLMORE	MANAGER	9	1994-08-09 00:00:00	56000.00	NULL	2
11	ADAMS	ENGINEER	10	1996-03-15 00:00:00	34000.00	NULL	2
12	WASHIN...	ADMIN	6	1998-04-16 00:00:00	18000.00	NULL	4
13	MONROE	ENGINEER	10	2000-12-03 00:00:00	30000.00	NULL	2
14	ROOSEVELT	CPA	9	1995-10-12 00:00:00	35000.00	NULL	1
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL

To populate my database, I first created a schema, then wrote SQL queries in MySQL Workbench to create tables and insert data. This provided the structured dataset needed for analysis and visualization in QuickSight.



Connecting QuickSight and RDS

To connect RDS to QuickSight, I opened Amazon QuickSight, and selected "New dataset." I chose RDS, filled in the connection details for my database, used the public network option, and validated the connection successfully.

This solution is risky as adding 0.0.0.0/0 to the inbound rule allows access from any IP address worldwide, exposing the RDS instance to potential unauthorized access and security threats. It's best to restrict access to trusted IPs only.

A better strategy

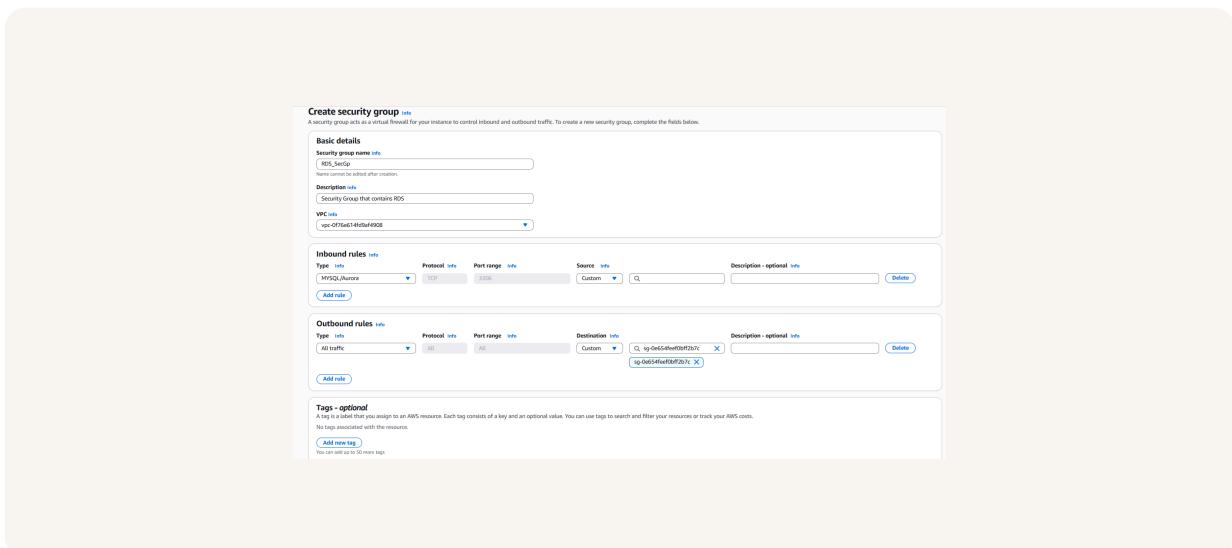
This new security group is created specifically for QuickSight to securely manage its network access. It allows me to authorize QuickSight as a trusted source, enabling safe and controlled connectivity to my RDS instance.

Next, I connected my new security group to QuickSight by navigating to Manage QuickSight, Manage VPC connections, then added a new VPC connection. I selected the correct database and assigned the execution role aws-quicksight-service-role-v0

Now to secure my RDS instance

To make my RDS instance secure, I created a new security group named RDS_SecGp in the default VPC, added an inbound rule for MySQL/Aurora, and allowed access only from my QuickSight_SecGp to ensure secure and restricted connectivity.

I made sure that my RDS instance could be accessed from QuickSight by modifying its settings, navigating to the Connectivity section, and attaching my newly created RDS_SecGp, removing any previously assigned security groups.





Adding RDS as a data source for QuickSight

New Amazon RDS data source

Data source name: RDS_VPC_Database

Instance ID: datasparksight

Connection type: RDS_VPC

Database name: QuickSightDatabase

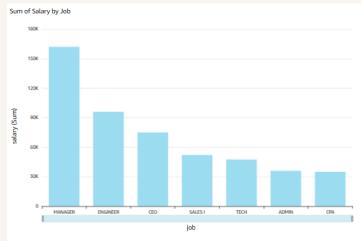
Username: admin

Password: [REDACTED]

Validated | SSL is enabled

Create data source

This data source is different from my initial one because it uses a more secure setup—both RDS and QuickSight are now in dedicated security groups, allowing controlled, private communication instead of relying on public access.





nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

