



nextwork.org

Visualise a Relational Database



ampahben3@gmail.com

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

	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

Introducing Today's Project!

What is Amazon RDS?

Amazon RDS is a managed database service that simplifies the setup, scaling, and operation of databases in the cloud. It automates backups, patching, and performance optimization while enhancing security and ensuring high availability.

How I used Amazon RDS in this project

I used Amazon RDS to store and manage the data for visualization in QuickSight. I secured the RDS instance, connected it to QuickSight, and set up a dataset to query tables for creating charts. This ensured reliable, scalable, and secure data.

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

One thing I didn't expect was the challenges with setting up QuickSight and deleting resources. The process kept blocking me, making progress difficult at times. But overcoming those obstacles made the setup more solid.

AM

ampahben3@gmail.com

NextWork Student

nextwork.org

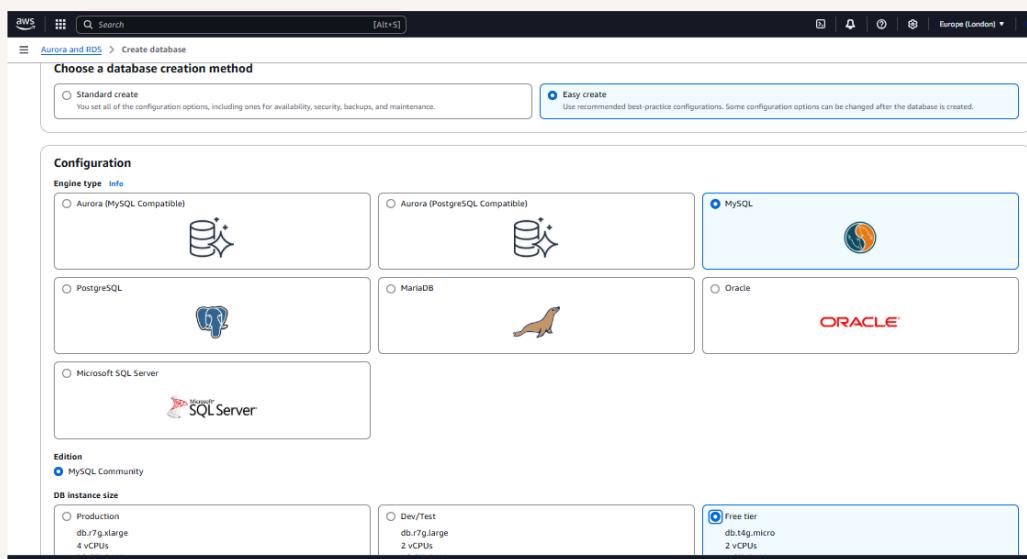
This project took me...

This project took me an hour and 45 minutes to complete because I needed to pay careful attention to each step. Ensuring proper security, setup, and troubleshooting took time, but the effort made the final result more reliable and efficient.

In the first part of my project...

Creating a Relational Database

I created my relational database by searching RDS in AWS, choosing Easy Create, selecting MySQL as the engine, setting the instance size to Free Tier, naming it QuickSight Database, defining my master username and password, and selecting self-managed



Understanding Relational Databases

A relational database is a structured collection of data organized into tables with defined relationships, using SQL for efficient querying and data management while maintaining integrity through keys and constraints.

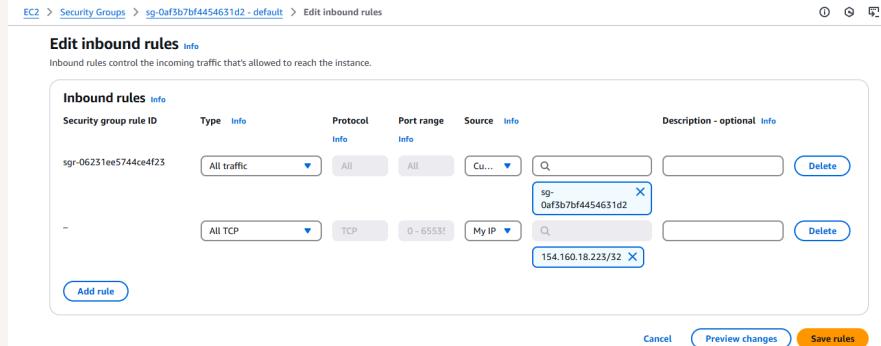
MySQL vs SQL

The difference between MySQL and SQL is that MySQL is a relational database management system that uses SQL to interact with databases, while SQL is a standard language for managing and querying data in various database systems.

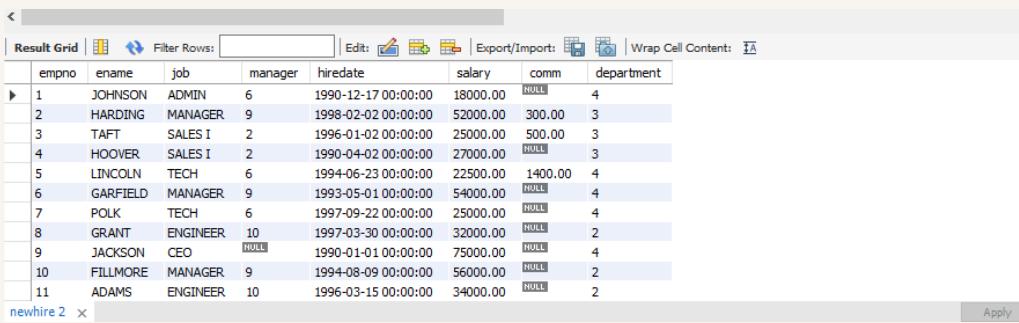
Populating my RDS instance

The first thing I did was make my RDS instance public because I needed it to be accessible from external networks, like my local machine. This allows connections from outside AWS but requires strict security settings to prevent unauthorized access.

I had to update the default security group for my RDS schema because I wanted my local MySQL Workbench to connect with AWS RDS. By modifying inbound rules, I allowed my workstation's IP to access the database, enabling secure remote management.



Using MySQL Workbench



The screenshot shows a MySQL Workbench interface with a result grid titled 'Result Grid'. The grid displays 11 rows of employee data from a table named 'newhire'. The columns are labeled: empno, ename, job, manager, hiredate, salary, comm, and department. The data includes various employees like Johnson, Harding, and Adams, with their respective details such as hire date (e.g., 1990-12-17) and salary (e.g., 18000.00). The 'department' column shows values like 4, 3, and 2.

	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

To populate my database, I used the `INSERT INTO` statement with `VALUES` to add structured employee records efficiently. This method ensured accurate data entry for easy retrieval and management.

Connecting QuickSight and RDS

To connect my RDS instance to QuickSight, I updated inbound rules, set up QuickSight, selected RDS, entered details, and successfully tested the connection. Now, I can visualize my data seamlessly.

This solution is risky because making the RDS instance public allows unrestricted access, exposing sensitive data to potential security threats. Unauthorized users could exploit vulnerabilities, modify or steal data, and impact system performance.

A better strategy

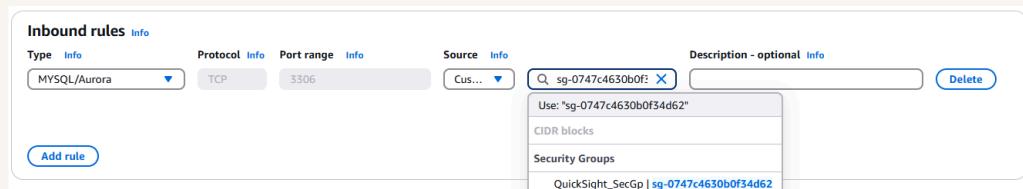
First, I created a new security group so EC2 instances could securely connect to QuickSight. This group was explicitly designed for QuickSight, ensuring controlled access and preventing unnecessary exposure while maintaining efficient data flow.

Next, I connected my new security group to QuickSight by creating a custom security group. Then, I followed the process to attach QuickSight to this group, ensuring controlled access to my RDS instance while maintaining security and preventing.

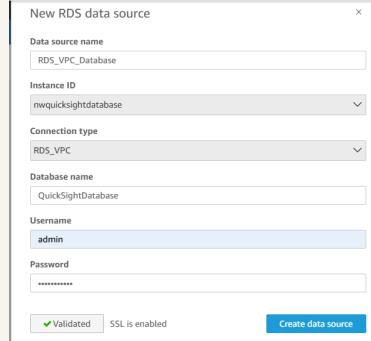
Now to secure my RDS instance

To secure my RDS instance, I created a security group and added an inbound rule for MySQL/Aurora. Then, I attached the security group I created for QuickSight, allowing controlled access while keeping my RDS instance private and protected.

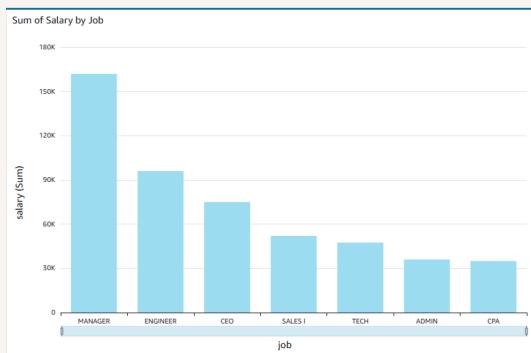
I made sure my RDS instance could be accessed from QuickSight by selecting it and modifying its security group settings. I set it to the RDS security group I created, allowing secure QuickSight access while keeping RDS private.



Adding RDS as a data source for QuickSight



This data source differs from my initial one because the previous connection used a Public Network, exposing my RDS instance. Now, it connects through the new security group created for RDS, ensuring restricted access and improved security.





nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

