

## Visualize a Relational Database

- Login with your IAM user.
- IAM admin user with custom password and **AdministratorAccess**.
- Choose **Download .csv file**.
- Login with the **Console sign-in URL**.

In this step, you're going to:

1. Create a relational database from scratch

 **What's a relational database? How is it different from a normal database?**

A relational database is a type of database that organizes data into tables, which are collections of rows and columns. We call it "relational" because the rows *relate* to the columns and vice-versa. When a database is relational we can query it using a special language called **SQL** (Structured Query Language).

- Head to your **RDS console** - search for rds in search bar at the top of the screen.
- In the left navigation bar, select **Databases**.
- In the Database section, select **Create Database**.
- On the Create database page, choose **Easy Create**.
- In the Configuration section, make the following changes:
  - For **Engine type**, choose **MySQL**.
  - For **DB instance size**, choose **Free tier**.

 **Whoa what is MySQL? How is it different to SQL?**

**SQL** (Structured Query Language) is a standard programming language used for managing and manipulating relational databases.

**MySQL** is a relational database management system (RDBMS) that uses SQL as the language for database interaction. If databases were libraries, think of RDBMS as the librarians who know exactly where each book is located, how to organize new books that arrive, and how to maintain the library's order. MySQL is software used to create, manage, and interact with relational databases but is only one of many options for using SQL to manage relational data. Other options include PostgreSQL, MariaDB, or SQLite.

- For **DB instance identifier**, type QuickSightDatabase.
- For **Master username**, enter admin.
- In the **Credentials management** section, select **Self managed**
- For **Master password**, type a unique password, and confirm password.
- Make sure you save your database login details somewhere safe! You'll need them later on.
- Leave the rest as the default settings and select choose **Create database**.

## Create database

### Choose a database creation method [Info](#)

#### 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 [Info](#)

##### Aurora (MySQL Compatible)



##### Aurora (PostgreSQL Compatible)



##### MySQL



##### MariaDB



##### PostgreSQL



##### Oracle



##### Microsoft SQL Server



#### Edition

##### MySQL Community

#### DB Instance size

##### Production

db.r6g.xlarge  
4 vCPUs

##### Dev/Test

db.r6g.large  
2 vCPUs

##### Free tier

db.t3.micro  
2 vCPUs

### This is what we created in this step:

Connect RDS to MySQL Workbench

Now that we've got our relational database shell in AWS (this is what we call our 'RDS instance'), we need to create a few tables and enter in some data.

To do this we'll be using one of the most popular database tools in the world, that isn't part of AWS (gasp!);

- **MySQL Workbench** from Oracle.

### In this step, you're going to:

1. Download MySQL Workbench

2. Make your RDS instance public, to allow connections from outside the AWS network (ie. from our local machine using MySQL Workbench)
3. Modify the security group attached to your RDS instance so that your local machine can access your RDS instance.
4. Connect MySQL Workbench to your RDS.

 **Why are we using MySQL Workbench and not something native to AWS?**

AWS RDS can actually import data from S3 using tools like AWS Database Migration Service or SQL commands, but these methods can require more setup and configuration.

If you're doing a large-scale data import then you might consider more AWS native workflows and CI/CD pipelines...but for this project MySQL Workbench suits us perfectly.

#### Download MySQL Workbench

- Open the [MySQL Workbench download page](#), select your operating system, and then download the top option.
- While that's downloading, let's make our RDS public so that we can modify it from MySQL Workbench.

#### Make your AWS database public so we can connect it to MySQL Workbench

- Open the Amazon RDS console, in the left-hand navigation pane, choose **Databases**.
- Then, choose **QuickSightDatabase**.
- On the **QuickSightDatabase** page, choose **Modify**.
- Scroll down to the Connectivity section, choose **Additional Configuration**.
- Then, choose **Publicly accessible**.
- Choose **Continue**.
- Choose **Apply immediately**.
- Select **Modify DB instance**.

This can take some time, so don't be concerned if it takes five or so minutes for the modifications to come through! It's a good time for another cup of tea.

#### Modify the security group attached to your RDS instance

Next we want to modify the security group attached to your RDS instance so that your local machine can access your RDS instance.

- In Amazon RDS in your AWS console, go to the left-hand navigation, and select **Databases**.
- Then, choose **quicksightdatabase**.

- On the **quicksightdatabase** page, in the Connectivity & security section, choose the **VPC security groups** link.
- On the **Security groups** page, choose the **Security group ID**.

 **What is this? Where are we?**

This is the default security group set up by AWS for our RDS instance. We didn't have to do anything special for it to exist, just create our usual RDS!

A **security group** is like a wall around your AWS resources that controls the inbound and outbound traffic to and from a resource. Not every AWS resource has one by default, but almost all that are inside a VPC do.

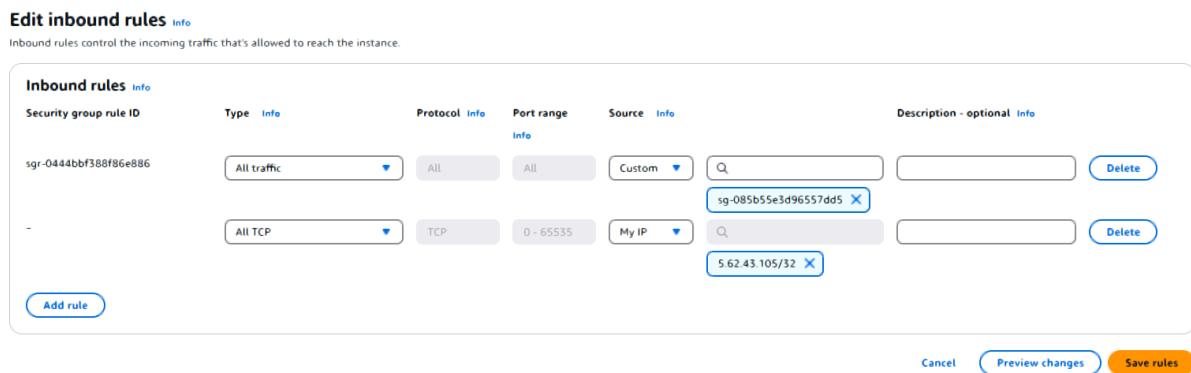
- On the sg-default page, in the Inbound rules section, choose **Edit inbound rules**.
- On the **Edit inbound rules** page, in the Inbound rules section, choose **Add rule**, and make the following changes:
  - For **Type**, choose **All TCP** from the drop-down list.
  - For **Source**, choose **My IP**.
- Then, choose **Save rules**.

**Edit inbound rules** Info

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-0444bbf388f86e886	All traffic	All	All	Custom	sg-085b55e3d96557dd5 <input type="button" value="Delete"/>
-	All TCP	TCP	0 - 65535	My IP	5.62.43.105/32 <input type="button" value="Delete"/>

**Add rule**



 **What did we just change? And why?**

Every RDS instance we create has a default security group attached to it. This security group controls the traffic going into and out of the database.

We need MySQL Workbench to access our database, so we need to change the inbound traffic rules in the security group.

By saying that we only allow traffic from our current IP address, we are only allowing our machine to connect to the database - this will enable MySQL Workbench to connect when we use it from our local computer, which makes it good for security!

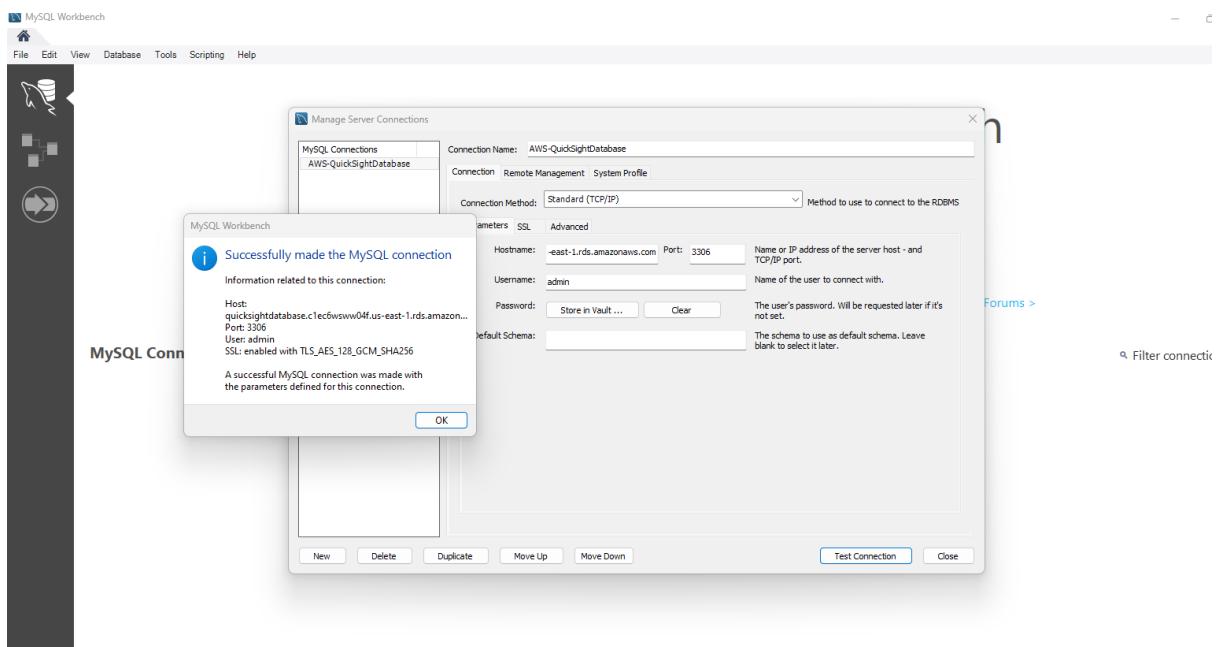
### Login to your Database from MySQL Workbench

- Verify that MySQL Workbench has downloaded successfully. Then, install and open the software.
- Select **Databases** in the top menu bar, then **Manage Connections...**

- Select **New** in the bottom left.
- Enter the following details:
  - For **Connection Name**, paste AWS-QuickSightDatabase
  - For **Hostname**, return to your RDS database page for your QuickSightDatabase.
  - Look under **Connectivity and security** to find the **Endpoint**. Example: QuickSightDatabase.abc.us-east-1.rds.amazonaws.com,1433
  - Copy this into **Hostname** in your MySQL Workbench.
  - Copy the **Port** from the same place in AWS to the **Port** field in MySQL Workbench.

**💡 Top tip:** To find the endpoint, open the Amazon RDS console, and choose **QuickSightDatabase**. On the **QuickSightDatabase** page, in the Connectivity & Security section, copy the **Endpoint** and **Port**.

- For **Username**, type the username you entered when creating the **QuickSightDatabase** (probably admin unless you changed it).
- For **Password**, select **Store in Keychain ...** then enter in your database password.
- Then, choose **Test Connection**.
- You should get a pop-up that says **Successfully made the MySQL connection**.



## Create Database Tables and Load Data

Now that we've got our MySQL Workbench connected with our RDS instance, we can actually start to add our own tables and data.

### In this step, you're going to:

1. Create a new schema using MySQL Workbench
2. Create two new tables in your schema

### 3. Populate those tables with data using SQL

This is going to help us create the data we want to pull into QuickSight to show off in some cool charts.

Let's load some data into our relational database!

- Close any pop-ups that are still open in MySQL Workbench.
- Select your newly created connection to open the MySQL Workbench Query Editor.
- Select **Schemas** as the tab in the top left, next to **Administration**.

#### What does 'Schema' mean?

A database **schema** is like a blueprint or structure of how the data is organized in a database. It sets up how the database is structured, including the tables, what fields should go in those tables, the relationships between tables, and other advanced components of a database.

Think of a schema as your design plan for how your data is stored and organized!

- Right click on the blank space under the **Schemas** menu.
- Select **Create Schema**.
- Name your schema **QuickSightDatabase**
- Leave everything else as is, and select **Apply**.
- Review the SQL Script pop-up; notice that it's showing you how the command looks in SQL!
- Select **Apply**.
- Once it's finished running, select **Close**.
- Select the tab marked  **Query 1** near the top of the screen.

#### Can't find the 'Query 1' tab? No worries!

Try selecting the first file icon in the top tool bar. When you hover over the icon it says "Create a new SQL tab for executing queries". Click this icon to open a new query script!

- In your new Query script, copy and paste the following SQL query.

```
CREATE TABLE newhire( empno INT PRIMARY KEY, ename VARCHAR(10), job VARCHAR(9), manager  
INT NULL, hiredate DATETIME, salary NUMERIC(7,2), comm NUMERIC(7,2) NULL, department INT)
```

- Run your Query script by selecting the lightning button above your script.

#### Did you get an error?

If you get an error it might be because you haven't selected your schema in the right menu. Make sure that you **double-click** your schema '**QuickSightDatabase**'. Then run the query again!

- To see the results from our query, delete the current query and replace it with the following:

```
SELECT * FROM newhire;
```

- Run this new query to see the empty table you just created!
- Now let's populate our new table by running another query.

- Delete the current contents of your query script and paste in the following:

```
INSERT INTO newhire (empno, ename, job, manager, hiredate, salary, comm, department) VALUES (1, 'JOHNSON', 'ADMIN', 6, '1990-12-17', 18000, NULL, 4), (2, 'HARDING', 'MANAGER', 9, '1998-02-02', 52000, 300, 3), (3, 'TAFT', 'SALES I', 2, '1996-01-02', 25000, 500, 3), (4, 'HOOVER', 'SALES I', 2, '1990-04-02', 27000, NULL, 3), (5, 'LINCOLN', 'TECH', 6, '1994-06-23', 22500, 1400, 4), (6, 'GARFIELD', 'MANAGER', 9, '1993-05-01', 54000, NULL, 4), (7, 'POLK', 'TECH', 6, '1997-09-22', 25000, NULL, 4), (8, 'GRANT', 'ENGINEER', 10, '1997-03-30', 32000, NULL, 2), (9, 'JACKSON', 'CEO', NULL, '1990-01-01', 75000, NULL, 4), (10, 'FILLMORE', 'MANAGER', 9, '1994-08-09', 56000, NULL, 2), (11, 'ADAMS', 'ENGINEER', 10, '1996-03-15', 34000, NULL, 2), (12, 'WASHINGTON', 'ADMIN', 6, '1998-04-16', 18000, NULL, 4), (13, 'MONROE', 'ENGINEER', 10, '2000-12-03', 30000, NULL, 2), (14, 'ROOSEVELT', 'CPA', 9, '1995-10-12', 35000, NULL, 1);
```

- Run the script by select the ' ⚡ ' button again!
- Once it's run, remove the query and run the following to see your results:

```
SELECT * FROM newhire;
```

- Remove the current query and run the following to create and populate a second table:

```
CREATE TABLE department( deptno INT NOT NULL, dname VARCHAR(14), loc VARCHAR(13)); INSERT INTO department (deptno, dname, loc) VALUES (1, 'ACCOUNTING', 'ST LOUIS'), (2, 'RESEARCH', 'NEW YORK'), (3, 'SALES', 'ATLANTA'), (4, 'OPERATIONS', 'SEATTLE');
```

- Once it's run, remove the query and run the following to see your results:

```
SELECT * FROM department;
```

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** Migration, AlViS-QuickSightDatabase.
- Toolbar:** Standard MySQL Workbench toolbar.
- Navigator:** Shows the schema structure under 'SCHEMAS' for 'QuickSightDatabase'. It includes 'Tables', 'Views', 'Stored Procedures', and 'Functions'.
- Query Editor:** Titled 'Query 1 - QuickSightDatabase - Schema'. It contains the SQL command:
 

```
1 • SELECT * FROM newhire;
```
- Result Grid:** Displays the data from the 'newhire' table. The columns are empno, ename, job, manager, hiredate, salary, comm, and department. The data is as follows:
 

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

#	Action	Message
3	CREATE TABLE newhire(empno INT PRIMARY KEY, ename VARCHAR(10), job VARCHAR(9), manager INT NULL...)	Error Code: 1046. No database selected Select the default DB to be used by double-click
4	CREATE TABLE newhire(empno INT PRIMARY KEY, ename VARCHAR(10), job VARCHAR(9), manager INT NULL...)	Error Code: 1046. No database selected Select the default DB to be used by double-click
5	CREATE TABLE newhire(empno INT PRIMARY KEY, ename VARCHAR(10), job VARCHAR(9), manager INT NULL...)	0 row(s) affected
6	SELECT * FROM newhire LIMIT 0, 1000	0 row(s) returned
7	INSERT INTO newhire (empno, ename, job, manager, hiredate, salary, comm, department) VALUES (1, 'JOHNSON', 'ADMIN', 6, '1990-12-17 00:00:00', 18000.00, NULL, '')	14 row(s) affected Records: 14 Duplicates: 0 Warnings: 0
8	SELECT * FROM newhire LIMIT 0, 1000	14 row(s) returned

The screenshot shows the AWS QuickSight Data Editor interface. At the top, there's a toolbar with various icons for database management. Below the toolbar, a tab bar shows "Query 1" and "QuickSightDatabase - Schema". A SQL editor window contains the following code:

```
1 •  SELECT * FROM department;
```

Below the code, the results are displayed in a "Result Grid" table:

	deptno	dname	loc
▶	1	ACCOUNTING	ST LOUIS
	2	RESEARCH	NEW YORK
	3	SALES	ATLANTA
	4	OPERATIONS	SEATTLE

The interface also includes a sidebar with options like "Result Grid", "Form Editor", "Field Types", and "Query Stats". At the bottom, there's a status bar showing "department 3" and "Read Only".

### Connect RDS to QuickSight

Now let's connect it up to QuickSight so we can start seeing some visual charts of our data.

#### In this step, you're going to:

1. Adjust the security group attached to our RDS instance to allow inbound requests from QuickSight.
2. Add your RDS instance as a data source in QuickSight.
  - Navigate back into your RDS instance from the **RDS console** in AWS.
  - Open your RDS instance.
  - Under **Connectivity & security**, select the link in **VPC security groups** to open the related security group.
  - Open the security group by selecting the **Security group ID**
  - Select **Edit inbound rules** to add a new rule with the following details:

- Type: **All Traffic**
- Source: **Custom**, then 0.0.0.0/0 in the next box

 **What does this inbound rule do?**

This inbound rule lets any external service access our RDS instance. **0.0.0.0/0** means "all addresses" so applies to anyone trying to access our RDS instance.

- Select **Save rules**
- Navigate to QuickSight by searching Amazon QuickSight in the search bar at the top of your AWS console.
- If this is your first time using QuickSight, follow the sign-up flow;
  - PLEASE make sure to **untick** the offer to upgrade with the optional add-on **Add Paginated Reports**. No getting charged today!
  - Make sure you select the same Region as the one you've been doing this project in.

 **How do I check which Region I'm in?**

Take a look at the top right of your AWS Console. You should see a drop-down with all the Regions available to you.

- Once you're in QuickSight, select **Datasets** from the left menu.
- In the top right of the screen, select **New dataset**
- WOW! Look at all those options! So many things we could build in the future.
- For now, let's stick to the mission; select **RDS**
- Fill out the following values:
  - **Data source name:** RDS\_Public\_Database
  - **Instance ID:** select your database from the drop-down
  - **Connection type:** Public network
  - **Database name:** QuickSightDatabase
  - **Username:** admin (or the username you created when you set up your RDS instance)
  - **Password:** enter in your RDS instance password

 It's critical that your Database name matches **exactly** to the name of your database. If things go wrong, this is a good place to double check your work!

- Select **Validate connection**

Manage QuickSight / Manage VPC connections / Add VPC Connection

### Add VPC Connection

Securely connect your data to QuickSight using a Virtual Private Cloud (VPC) connection. [Learn more](#)

AWS console links

- [VPC](#)
- [Subnet](#)
- [Security group](#)
- [DNS resolvers](#)
- [IAM console](#)

VPC connection name: RDS\_VPC

VPC ID: vpc-045084a379fb9e1c

Execution role: aws-quicksight-service-role-v0

Subnets (Select at least two)

Availability Zone	Subnet ID
us-east-1a	subnet-020cee657c7f6a041
us-east-1b	subnet-015e54eb6cbe472c0
us-east-1c	subnet-095c35fc6e4abd8dc
us-east-1d	subnet-03a7831803870795c
us-east-1e	subnet-08e6dd0e384bf39d
us-east-1f	subnet-054a89db72baf282a

Security Group IDs: sg-08a5b3362bc5494b6

DNS resolver endpoints (optional)  
One endpoint per line

**ADD** **CANCEL**

The RDS instance is **publicly available**. This is bad because it means anyone can access it, making us vulnerable from hackers and malicious people trying to get our data.

We definitely need to make this more secure. Ideally, we can adjust it so only QuickSight can access our RDS instance.

### Secure QuickSight

We can put QuickSight in a Security Group and our RDS in a Security Group, then let our RDS Security accept requests from the QuickSight security group only.

#### In this step, you're going to:

1. Create a new security group for QuickSight
2. Attach QuickSight to our new security group

This is going to help us to connect QuickSight to our RDS instance securely, without RDS being publicly accessible.

#### Create security group for QuickSight

- Close your current QuickSight data source pop up and return to your AWS console.
- Search for security groups in the search bar in your AWS console
- Select **Create security group**
- For **Security group name** enter QuickSight\_SecGp
- For **Description** enter Security Group that contains QuickSight
- Select the **default VPC** as your VPC option. We haven't created our own VPC so the default one is what our RDS and QuickSight will be living in.
- Leave the inbound and outbound rules as they are.

- Select **Create security group**
- Take note of your new **QuickSight\_SecGp ID**; take a screenshot or copy and paste it somewhere safe. You'll need it so we can attach our new security group to QuickSight.

We have an empty QuickSight Security Group living inside the same VPC as our RDS instance.

### Attach Security Group to QuickSight

Now that we've created our Security Group we need to actually associate it with to QuickSight. Off we go!

- Navigate to QuickSight using the search bar.
- Select the profile icon in the top right and select **Manage QuickSight** from the dropdown.
- Select **Manage VPC connections**
- Select the **Add VPC connection** button
- For **VPC connection name**, enter **RDS\_VPC**
- Select the VPC from the dropdown that matches the one you added to your QuickSight security group. If you only see one VPC in the dropdown, that'll be it!
- For **Execution role**, select **aws-quicksight-service-role-v0**.

#### What is the Execution role?

The Execution role is the IAM role which will give us the permissions to create this VPC connection.

- Select the default dropdown options for the **Subnet ID** fields
- For **Security Group IDs** select the same ID as your **QuickSight\_SecGp** which you saved earlier.
- Select **Add** to confirm your VPC
- "The role provided is unauthorized to perform the required actions".
- The **aws-quicksight-service-role-v0** role we selected as the **Execution role**. We need to edit it to have the proper permissions to execute this task.

### Update role

- Open a new tab to your AWS console and search for IAM
- In the left menu, select **Roles**
- In the Roles search bar, search for our role; **aws-quicksight-service-role-v0**
- Click into the **aws-quicksight-service-role-v0**
- In the **Permissions policies** section, select **Add permissions**
- From the dropdown, select **Create inline policy**
- Select the **JSON** option as a policy editor
- Paste in the following IAM policy:

```
{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "ec2:DescribeVpcs", "ec2:DescribeSubnets", "ec2:DescribeSecurityGroups", "ec2:DescribeNetworkInterfaces", "ec2>CreateNetworkInterface", "ec2>DeleteNetworkInterface", "ec2:ModifyNetworkInterfaceAttribute", "iam:PassRole" ], "Resource": "*" }, { "Effect": "Allow", "Action": [ "iam:PassRole" ], "Resource": "*" } ] }
```

 **Hold on a second! How did you know what policies to put in here?**

Great question! One of the fastest ways to find out which policies to add is to use [ChatGPT](#) and explain the situation.

A prompt along the lines of,

" I'm creating a VPC Connection in QuickSight and my Execution Role doesn't have the right permissions to perform this task. Please create the necessary JSON policy to attach to my Execution Role in IAM "

should do the trick!

- Select **Next**
- For **Policy name**, enter QuickSightAllowVPC
- Select **Create policy**

### Let's try again!

- Return to your QuickSight VPC connection.
- Select **Add** one more time.
- We now have attached QuickSight to our new security group.

### Secure RDS

Alright - so our QuickSight service is now safely in our own security group. Let's do the same for our RDS instance and stop using the default security group.

#### In this step, you're going to:

1. Make our RDS instance private instead of publicly accessible.
2. Create a new security group specially for our RDS instance.
3. Give our QuickSight security group access to our RDS security group so they can talk to each other.

### Make your Database Not Publicly Accessible

The database no longer needs to be publicly accessible; that's way too sketchy. We're going to do a much better job with security groups.

- Open the Amazon RDS console, in the left-hand navigation, select **Databases**.

- Then, choose **QuickSightDatabase**
- On the **QuickSightDatabase** page, choose **Modify**.
- On the **ModifyDB instance: QuickSightDatabase** page, in the Connectivity section, choose **Additional Configuration**.
- Select **Not publicly accessible**, and choose **Continue**.
- Select **Apply immediately**.
- Select **Modify DB instance**.

#### Create a security group for RDS

- Search for security groups in the search bar in your AWS console
- Select **Create security group**
- For **Security group name** enter **RDS\_SecGp**
- For **Description** enter Security Group that contains RDS
- Select the **default VPC** as your VPC option. Our RDS security group will live in the same VPC as our QuickSight security group. Perfect!
- Add inbound rules to allow QuickSight to query our RDS instance;
  - Under **Inbound rules**, select **Add rule**
  - For **Type** select **MySQL/Aurora**
  - For **Source** select **Custom** and then search for the security group ID of your **QuickSight\_SecGp**
- Select **Create security group**

Nice! Now let's attach it to our RDS instance.

- Return to your RDS instance and select **Modify**
- Under the **Connectivity** section, look for **Security group**
- Select your newly created **RDS\_SecGp** and remove any existing one.
- Select **Continue**
- Select **Apply immediately**
- Select **Modify DB instance**

Epic. We've created our own RDS security group, added inbound rules to allow QuickSight in, and attached it to our RDS instance.

## Reconnect RDS with QuickSight

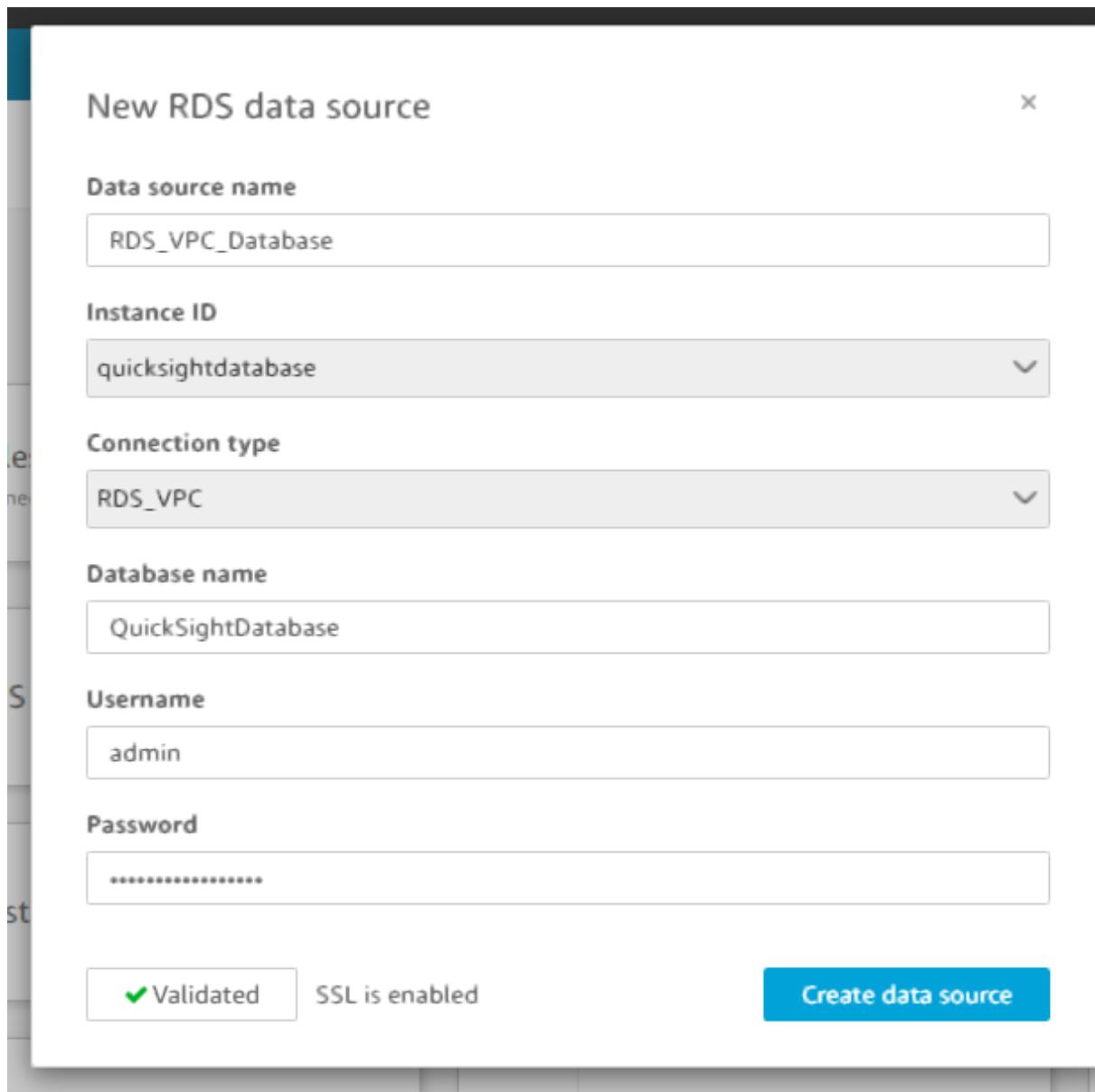
Now that everything is wired up, secure, and ready to go, we can jump back into QuickSight and add our new secure data source.

### In this step, you're going to:

1. Create a dataset in QuickSight to connect with our new security group
2. Choose the table we want to query to create our charts
  - Return to the **QuickSight** console (you may need to click the QuickSight logo in the top left to leave the QuickSight VPC settings).
  - Select **Datasets**
  - Select **New dataset**
  - Select **RDS**
  - Fill out the following values:
    - **Data source name:** RDS\_VPC\_Database
    - **Instance ID:** select your database from the drop-down
    - **Connection type:** RDS\_VPC (not 'Public network' - yay!)
    - **Database name:** QuickSightDatabase
    - **Username:** admin (or the username you created when you set up your RDS instance)
    - **Password:** enter in your RDS instance password

 Remember; it's critical that your Database name matches **exactly** to the name of your database. If things go wrong, this is a good place to double check your work!

- Select **Validate connection**



- Select **Create data source**
- Select **newhire** as the table to visualize.
- Click **Select**
- Select **Directly query your data** and then **Visualize**.

Make some charts!

Follow these steps to create a security group for Amazon QuickSight to access the RDS database in a VPC.

- Cancel any pop-up that shows and select the **Vertical Bar Chart** from the left hand **Visuals** section.
- Drag **jobs** into the **x-axis**.
- Drag **salary** into the **Value** measure.

- Continue adding any other charts you feel like!
- When you're ready, select **Publish** in the top right
- Name your dashboard RDS New Hire Data
- Select **Publish Dashboard**

