

CIS 5500: Database and Information Systems

AWS RDS Handout

1. Introduction

This handout provides instructions for creating and using an RDS (Relational Database Service) instance on AWS and getting started with MySQL.

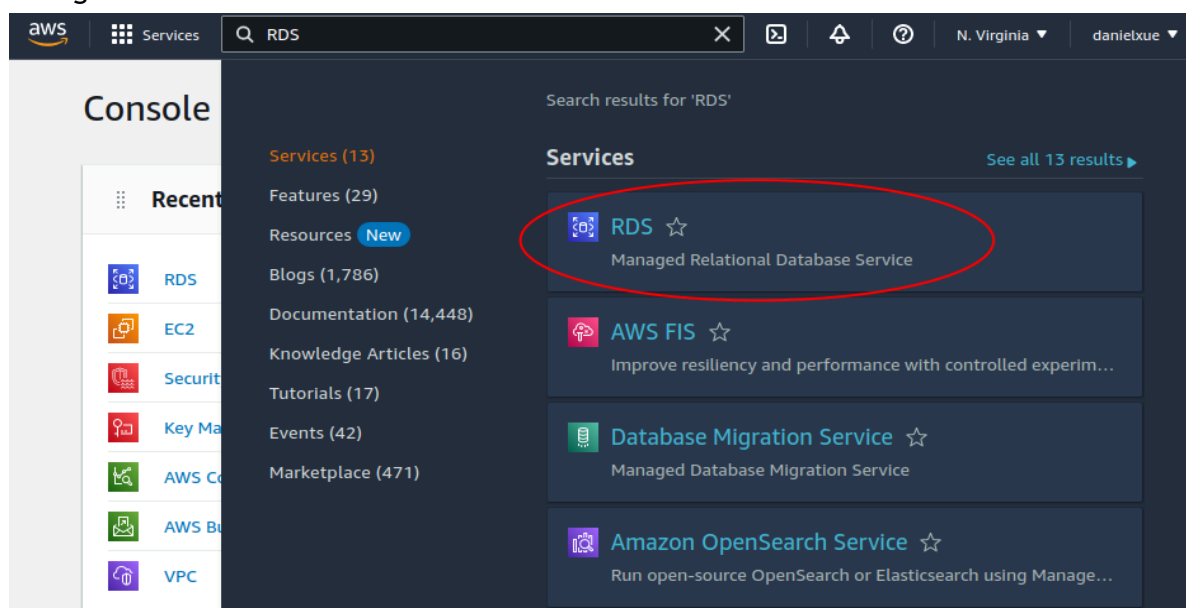
Note: This guide is meant to serve as a general reference - AWS often makes changes to the console interface, but the processes described in this document largely remain the same. You may ask a staff member for assistance in case online resources fail to help. This guide is meant for a root user AWS account, as opposed to an AWS Educate account. In case you have an AWS Educate account, you may choose to create a new account and follow this guide instead or ask a staff member for assistance.

In following this guide, you should remain within the [free tier](#) limits (you might be asked to provide payment information on AWS so you can be charged in case you exceed them), but please be sure to follow **all** setup instructions to ensure you do not exceed usage limits. You will ***not*** be reimbursed for any AWS expenses. Please track your usage in the Billing Dashboard and post on Ed if for some reason your usage is higher than expected so a TA can help resolve the error.

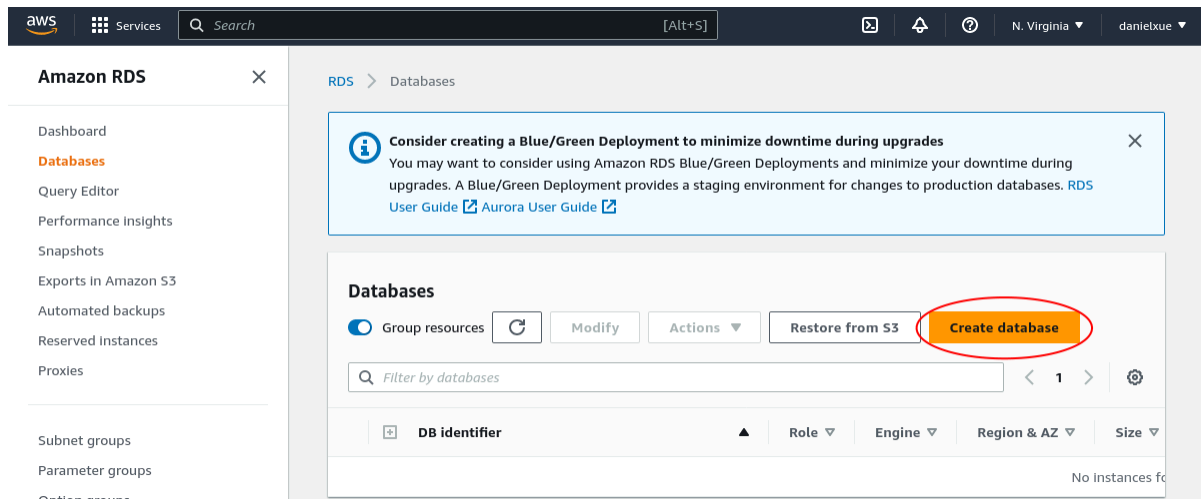
We recommend that you create **one** MySQL RDS instance for all of this semester's assignments and instead create separate databases within this single RDS instance as necessary. Creating multiple RDS instances can cause you to exceed free tier limits.

2. Creating an RDS Instance

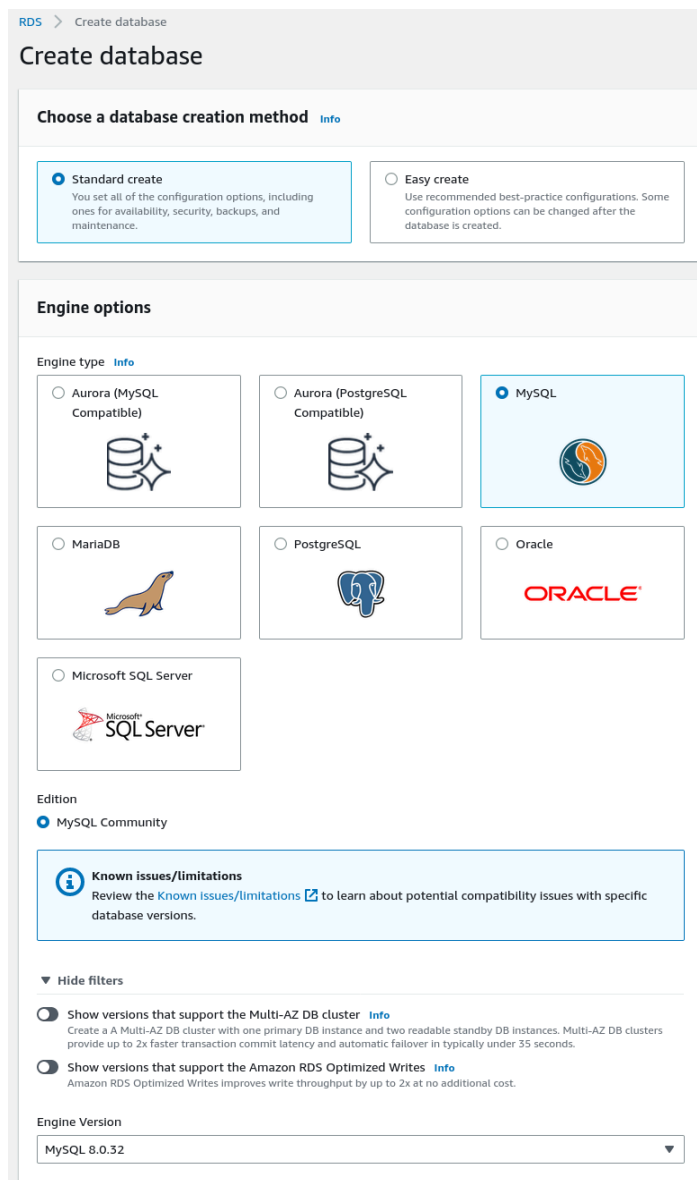
1. Create an AWS account [here](#) if you already do not have one. Then, log in and head over to the AWS [console](#). You can then go to the RDS console using [this link](#), but if that does not work, search for RDS on the top of the window and click on 'RDS - Managed Relational Database Service' as shown below:



2. From the left menu, select 'Databases' and click the 'Create Database' button:



3. This will bring up the create database wizard. For 'Choose a database creation method' use 'Standard Create'. For 'Engine options' select 'MySQL' and leave the Edition as 'MySQL Community' with the default version:



4. Next, under 'Templates', select the 'Free tier' option:

Templates
Choose a sample template to meet your use case.

- ☐ **Production**
Use defaults for high availability and fast, consistent performance.
- ☐ **Dev/Test**
This instance is intended for development use outside of a production environment.
- ☒ **Free tier**
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.
[Info](#)

5. Next, fill in the 'Settings' section - choose a master username and password that you will use to connect to the instance. Be sure to note down your admin credentials, choose a strong password, do not upload your admin credentials to a public repository, and do NOT store sensitive information in this instance. We have had student databases instances hacked in the past-don't let that be you!
6. Now, under the 'DB instance class' - use 'Burstable classes' - db.t2.micro or db.t3.micro (whatever the the default option is):

Instance configuration
The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class [Info](#)

- ☐ Standard classes (includes m classes)
- ☐ Memory optimized classes (includes r and x classes)
- ☒ Burstable classes (includes t classes)

db.t3.micro
2 vCPUs 1 GiB RAM Network: 2,085 Mbps

☐ Include previous generation classes

7. Uncheck 'Enable storage autoscaling' under 'Storage' and leave Storage type as General Purpose (SSD) with a 20 GiB allocated size. Double check you've done so! Missing this step is a common reason students exceed their free tier limits:

Storage

Storage type [Info](#)

General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage [Info](#)

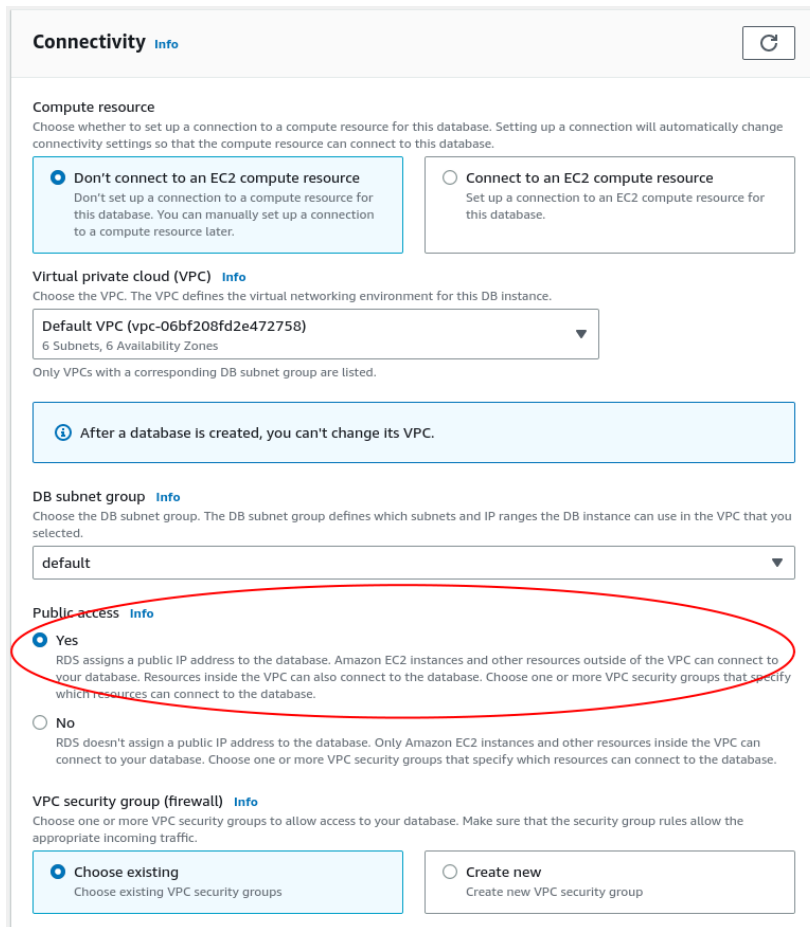
20 GiB

The minimum value is 20 GiB and the maximum value is 6,144 GiB

Storage autoscaling [Info](#)
Provides dynamic scaling support for your database's storage based on your application's needs.

☐ **Enable storage autoscaling**
Enabling this feature will allow the storage to increase after the specified threshold is exceeded.

8. Under connectivity, leave the 'VPC' and 'Subnet group' settings to their defaults. Set 'Public Access' to 'Yes' and leave the rest as is:



Connectivity [Info](#)

Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☒ **Don't connect to an EC2 compute resource**
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☐ **Connect to an EC2 compute resource**
Set up a connection to an EC2 compute resource for this database.

Virtual private cloud (VPC) [Info](#)
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-06bf208fd2e472758)
6 Subnets, 6 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

DB subnet group [Info](#)
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

default

Public access [Info](#)

☒ **Yes**
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

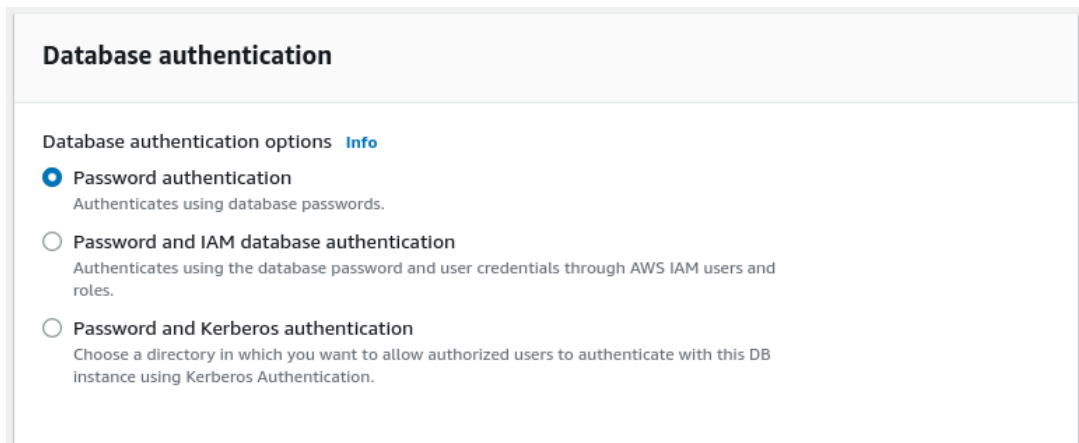
☐ **No**
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall) [Info](#)
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

☒ **Choose existing**
Choose existing VPC security groups

☐ **Create new**
Create new VPC security group

9. Lastly, leave 'Database authentication options' set to 'Password authentication':



Database authentication

Database authentication options [Info](#)

☒ **Password authentication**
Authenticates using database passwords.

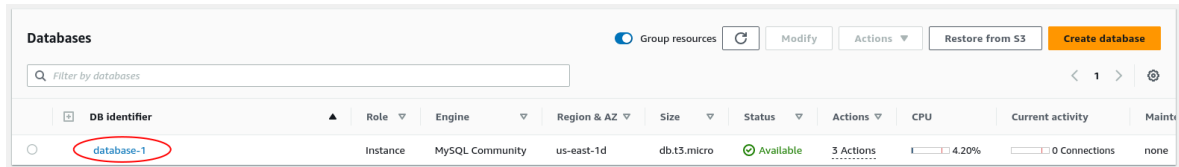
☐ **Password and IAM database authentication**
Authenticates using the database password and user credentials through AWS IAM users and roles.

☐ **Password and Kerberos authentication**
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

You may now encounter a section on estimated monthly costs. It should provide some information on free tier usage, but if it shows an estimated cost instead, check if correctly selected a free-tier instance template (as opposed to 'Production' or 'Dev/Test' templates) in step 4 of this section.

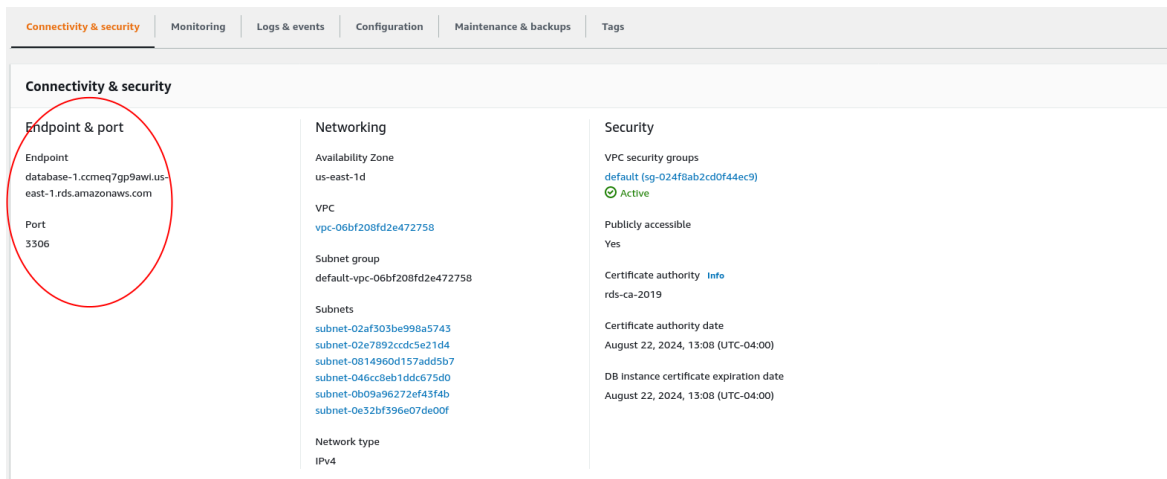
10. Click on 'Create database'. Upon redirection to the 'Databases' page, wait until the instance is created (the status parameter should read ✔ Available before you are able to connect to the instance). This process can take a few minutes!

11. Once the instance is created, you will be able to click on the corresponding entry for your instance in the DB identifier column:



DB identifier	Role	Engine	Region & AZ	Size	Status	Actions	CPU	Current activity	Maint
database-1	Instance	MySQL Community	us-east-1d	db.t3.micro	Available	3 Actions	4.20%	0 Connections	none

This will open the instance specific information page. Take a look at the 'Connectivity and security' pane and note down the endpoint (host) and port information. You will need this to connect to the instance as in section 4:

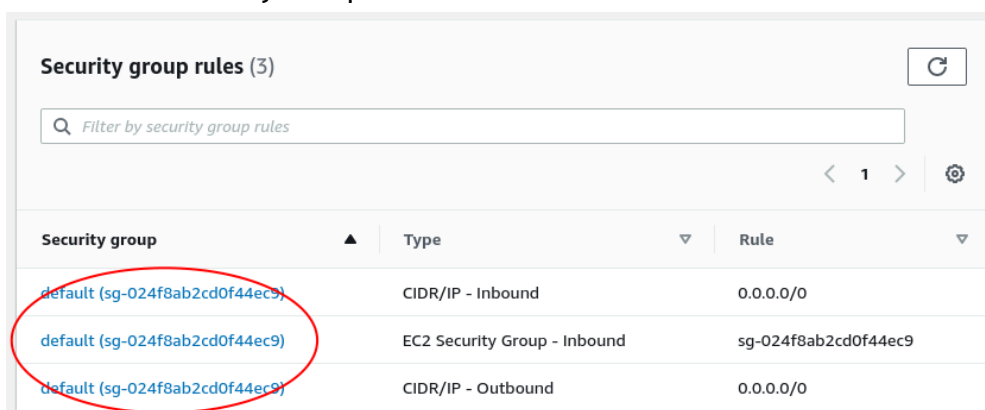


Endpoint & port	Networking	Security
Endpoint database-1.cmcq7gp9awl.us-east-1.rds.amazonaws.com Port 3306	Availability Zone us-east-1d VPC vpc-06bf208fd2e472758 Subnet group default-vpc-06bf208fd2e472758 Subnets subnet-02af303be996a5743 subnet-02e7892c0cd5e21d4 subnet-0814960d157add5b7 subnet-046cc9eb1ddc675d0 subnet-0b09a96272ef43f4b subnet-0e32bf396e07de00f Network type IPv4	VPC security groups default (sg-024f8ab2cd0f44ec9) Active Publicly accessible Yes Certificate authority Info rds-ca-2019 Certificate authority date August 22, 2024, 13:08 (UTC-04:00) DB instance certificate expiration date August 22, 2024, 13:08 (UTC-04:00)

3. Configuring Security Groups

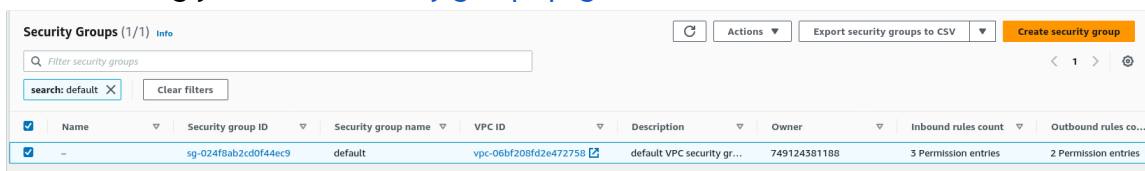
To allow for your client as well as the autograder to connect to your database, please allow access to all traffic from any IP in and out of the instance. Note this is not best practice for security in production systems, but will suffice for the purposes of this course. To do this:

1. From the database instance page, click on any one of the security group items form under 'Security Group Rules':



Security group	Type	Rule
default (sg-024f8ab2cd0f44ec9)	CIDR/IP - Inbound	0.0.0.0/0
default (sg-024f8ab2cd0f44ec9)	EC2 Security Group - Inbound	sg-024f8ab2cd0f44ec9
default (sg-024f8ab2cd0f44ec9)	CIDR/IP - Outbound	0.0.0.0/0

This will bring you to the [security groups page](#) as shown below:



Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules count	Outbound rules count
default	sg-024f8ab2cd0f44ec9	default	vpc-06bf208fd2e472758	default VPC security group	749124381188	3 Permission entries	2 Permission entries

- Click on the Security group ID (there should be only 1 since the link will filter the others to match only the one for the RDS instance). Under the 'Inbound tab, click on the 'Edit inbound rules'. First delete all existing rules and then add two rules with type 'All traffic' and sources 'Anywhere-IPv4' and 'Anywhere-IPv6':

The screenshot displays the 'Inbound rules' configuration page in the AWS Management Console. It shows two existing rules, 'Inbound rule 1' and 'Inbound rule 2', both set to 'All traffic'. Rule 1 has a source of '0.0.0.0/0' (IPv4) and Rule 2 has a source of ':::/0' (IPv6). The 'Add rule' button is located at the bottom left of the rule list. At the bottom of the console, there are buttons for 'Cancel', 'Preview changes', and 'Save rules'.

- Click on 'Save Rules'. Now go to the 'Outbound rules tab and click 'Edit Outbound Rules' and do the same-delete all existing rules and then add two rules with type 'All traffic' and sources 'Anywhere-IPv4' and 'Anywhere-IPv6':

Outbound rules [Info](#)

Outbound rule 1 [Delete](#)

Security group rule ID: -

Type: [Info](#) All traffic

Protocol: [Info](#) All

Port range: [Info](#) All

Destination type: [Info](#) Anywhere-IPv4

Destination: [Info](#) 0.0.0.0/0

Description - optional: [Info](#)

Outbound rule 2 [Delete](#)

Security group rule ID: -

Type: [Info](#) All traffic

Protocol: [Info](#) All

Port range: [Info](#) All

Destination type: [Info](#) Anywhere-IPv6

Destination: [Info](#) :::0

Description - optional: [Info](#)

[Add rule](#)

[Cancel](#) [Preview changes](#) [Save rules](#)

4. Connecting to the RDS Instance via DataGrip

This process is very similar to the one described in the **Section 5** of the DataGrip Handout. The key differences are:

1. In step 1, you should select the option for 'MySQL' instead of 'Oracle'
2. In step 2, once the data sources and driver connection prompt, you will use only the following connection parameters:
 - a. Host: From section 2 step 11
 - b. User: the username you selected in step 5 of section 2
 - c. Password: the password you selected in step 5 of section 2
 - d. Port: the port assigned for the MySQL service - from section 2 step 11
3. Again, hit 'Apply' and test the connection, and then 'OK' to open the corresponding editor console.

Note: MySQL connections also have a 'database' parameter. Since at the time of connection, you won't have a database set up, leave this blank and once the database is set up in section 5, use the 'USE <database-name>' command instead. MySQL instances default to running on port 3306 unless specified otherwise (please leave it as its default port to avoid issues).

5 Getting Started and Setting up a Database

Once connected to the instance, you may use the following commands to create and delete tables or databases. You should read through and understand these before proceeding with Homework 1 Part 2.

SHOW DATABASES; prints all current databases. For a MySQL instance, there will be a few system

information databases named ‘*information_schema*’, ‘*mysql*’, ‘*performance_schema*’, and ‘*sys*’.

To create a database named ‘*SCHOOL_DB*’, run the command: **CREATE DATABASE SCHOOL_DB;**

You must then select the database by running: **USE SCHOOL_DB;**

To create a table named *MY_TABLE* with the following schema (the key id is underlined):

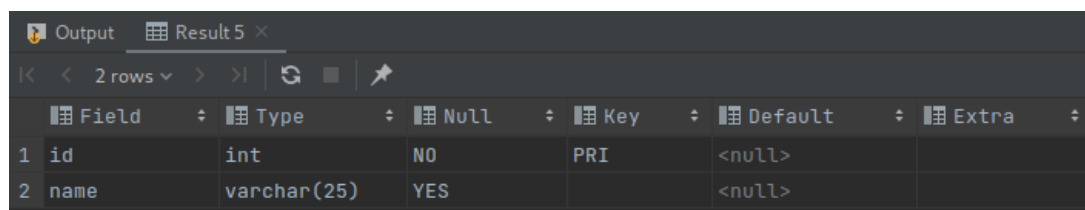
	Attribute	Type
MY_TABLE	<u>id</u>	int
	name	varchar(25)

Run the CREATE command:

CREATE TABLE STUDENTS (id int, name varchar(25), PRIMARY KEY(id));

To display the schema and constraints of *STUDENTS* use the describe command:

DESCRIBE STUDENTS;



Field	Type	Null	Key	Default	Extra
1 id	int	NO	PRI	<null>	
2 name	varchar(25)	YES		<null>	

Finally, use the DROP command delete the table: **DROP TABLE STUDENTS;**

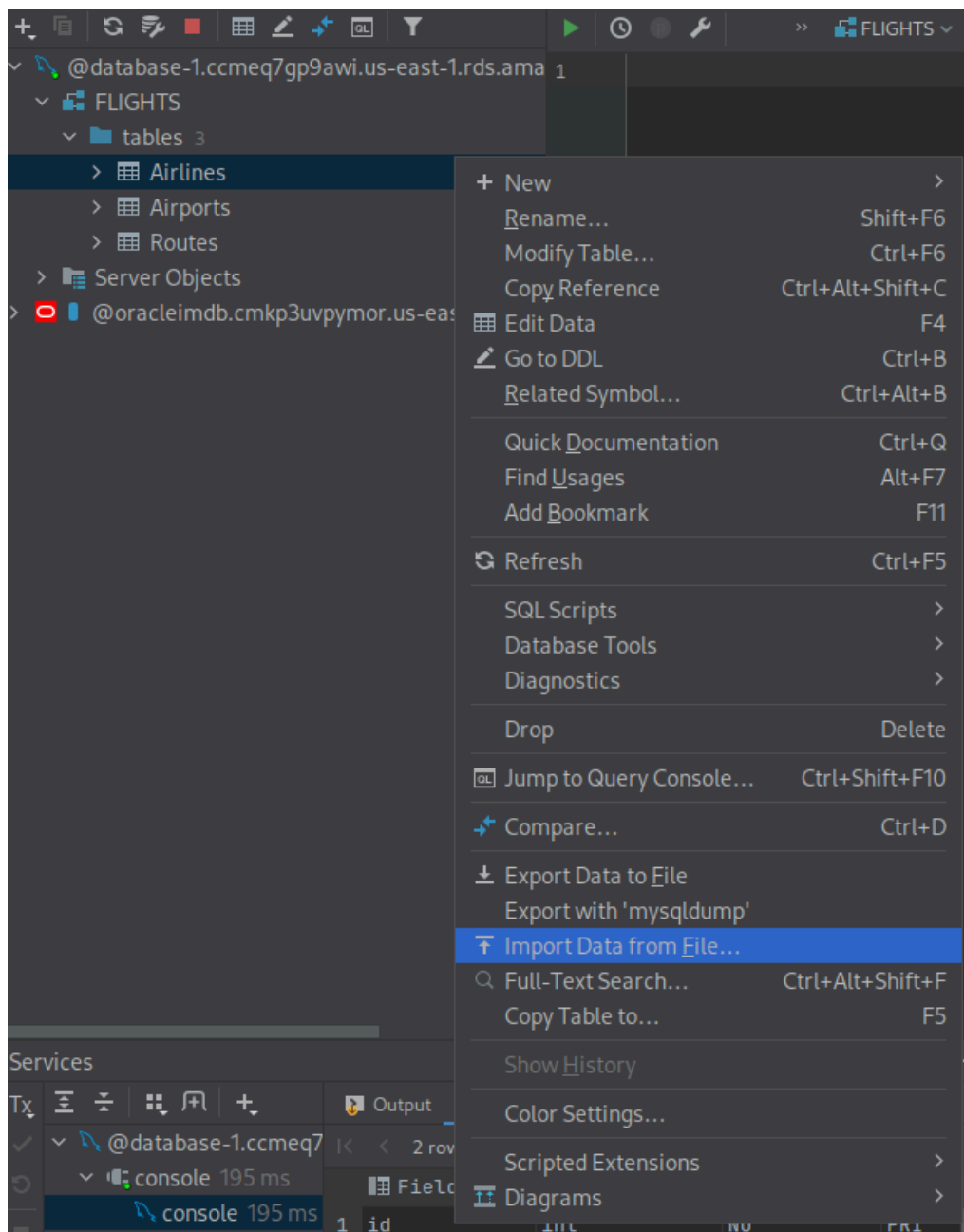
You could also use the DROP command to delete the entire *SCHOOL_DB* database:

DROP DATABASE SCHOOL_DB;

6. Uploading Data to Your Instance

This section contains instructions on uploading data to tables in an existing database using DataGrip’s import wizard. It is also possible to use the LOAD DATA statement in MySQL to do this as described [here](#). If you have not already created a database, refer to section 5 before proceeding. If you have not already made tables to upload data to, create all necessary tables. Take a look at Homework 1 Part 2 for instructions on creating tables needed for Parts 2 and 3 of the homework.

Once the database and tables are created, you will be able to see them on the left pane under ‘Database’. Look for the data source you’re currently working on (ex. *database-1*), and look for the database (ex. *FLIGHTS*) and tables (ex. *Airlines*, ...). To import data into a table, right click on it, and then select the ‘Import Data from File’ option:



Select the file you want to import (ex. *Airlines.csv*). This will bring up the import wizard, which will automatically detect the required formatting settings. Ex. for the flights dataset following settings are automatically detected.

Formats: CSV, TSV, Pipe-separated

Value separator: Comma

Row separator: Newline

Null value text: Empty string

Add row prefix/suffix

Quotation: " " Escape: duplicate, ' ' Escape: duplicate

Quote values: When needed

☐ Trim whitespaces

☒ First row is header

☐ First column is header

Target schema: @database-1.ccmeq7gp9awi.us-east-1.rds.am

Table: Airlines Existing

Comment:

Columns (8) Keys (1) Indexes Foreign Keys

id int -- part of primary key mapped to C1

name varchar(255) mapped to C2

alias varchar(255) mapped to C3

Data Preview DDL Preview

	C1	C2	C3	C4	C5
1	1	Private flight	\N	-	N/A
2	2	135 Airways	\N		GNL
3	3	1Time Airline	\N	1T	RNX
4	4	2 Sqn No 1 Elemen...	\N		WYT
5	5	213 Flight Unit	\N		TFU
6	6	223 Flight Unit S...	\N		CHD

Encoding: UTF-8

☒ Write errors to file: danielx/Airlines_2023-05-04_18_32_18.txt

☐ Insert inconvertible values as null

☐ Disable indexes and triggers, lock table (may be faster)

Import Cancel

Note the different options however. “null value text” may useful for handling how you want null values to be imported, “First row is header” may be useful for if the CSV has a header row (which is not the case for homework 1, but may be the case for other homeworks or datasets).

The wizard will also show you column mappings and a preview of the imported data. You may need to adjust Click on ‘Import’ once the import settings are satisfactory.

7. Modifying or Instance State

To avoid continuing to incur charges, you should be sure to **shut down your instance** from the instance page when you are done with the homework.