

RDS, PostgreSQL, and pgAdmin

- Open up the pgAdmin UI. Explain the following to students:
 - pgAdmin will connect to a cloud-based database, such as AWS, as well as local databases.
 - pgAdmin offers a visual interface for managing data.
- Log in to the AWS console and navigate to “RDS” under “Database.”

▼ All services



Compute

EC2
Lightsail [↗](#)
Elastic Container Service
EKS
Lambda
Batch
Elastic Beanstalk



Management Tools

CloudWatch
AWS Auto Scaling
CloudFormation
CloudTrail
Config
OpsWorks
Service Catalog
Systems Manager
Trusted Advisor
Managed Services



Storage

S3
EFS
Glacier
Storage Gateway




Media Services

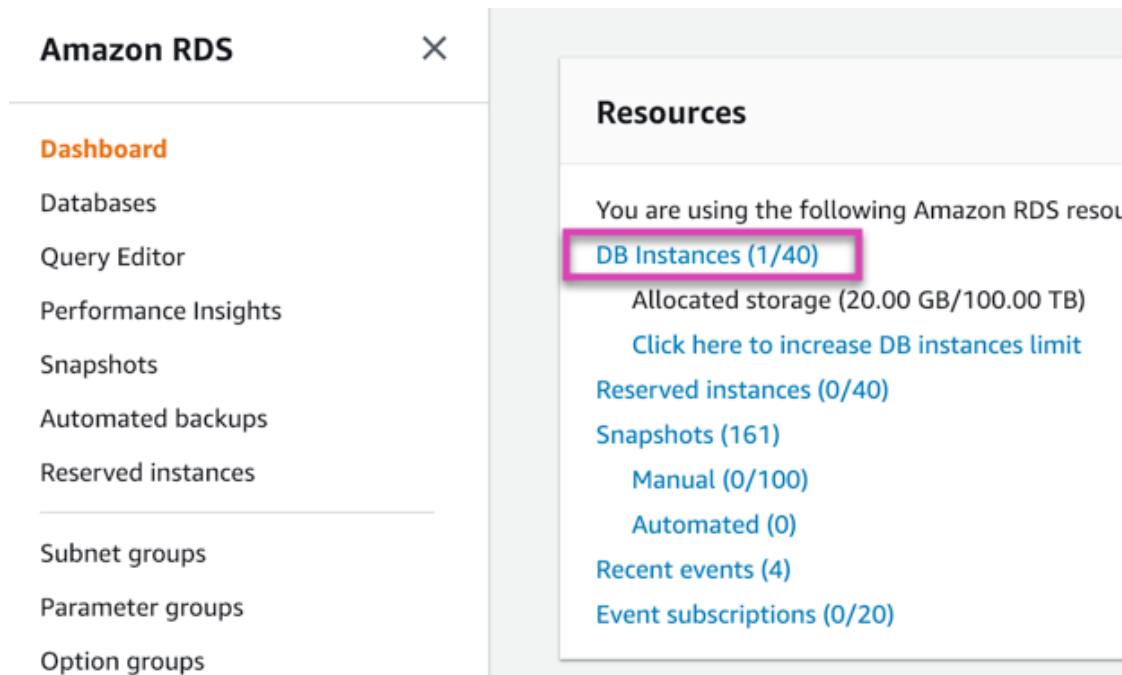
Elastic Transcoder
Kinesis Video Streams
MediaConvert
MediaLive
MediaPackage
MediaStore
MediaTailor



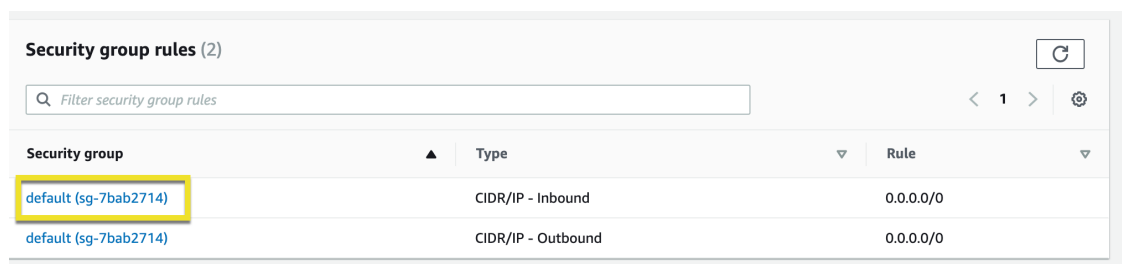
Database

RDS 
DynamoDB
ElastiCache
Neptune
Amazon Redshift

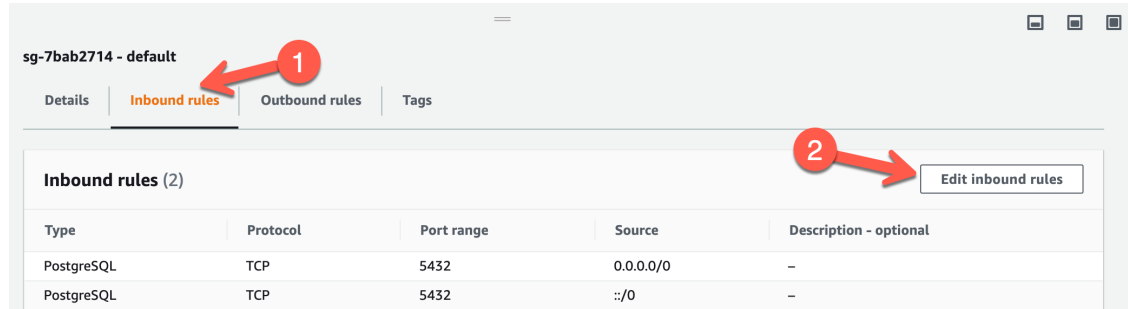
- Click on the “Services” dropdown arrow on the top left, and then select “RDS”. In the “Resources” section to the right, select “DB Instances”.



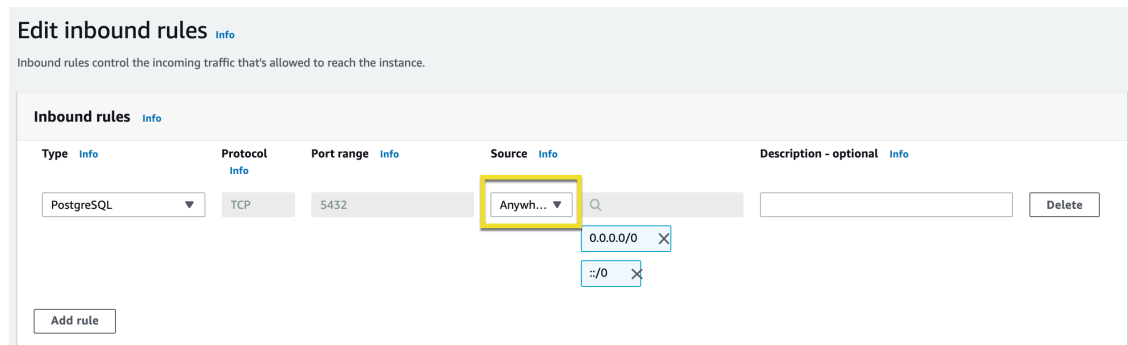
- Go to the database created earlier, mypostgresdb.
- Navigate to the “Security Group” rules section on the right, and explain the following:
 - These security groups tell the RDS instance what traffic is allowed in and out of the database.
 - The security settings can range from restrictive, such as a specific URL, to open.
- The database will be open to all traffic.
- Click the security group for type “CIDR/IP - Inbound.”



- You will be redirected to a new page. Follow these steps to give the database access to all inbound traffic:
 - Click on the “Inbound rules” tab, then click **Edit Inbound Rules**. This will bring up a menu to set rules for the security group.



- Change the “Source” to “Anywhere” and click **Save**. The RDS instance will now accept a connection from anywhere. This isn’t completely open to the world because the endpoint, username, and password are still needed to connect.



- Navigate back to the instance console, and have the class find the endpoint in the “Connectivity” tab.

Connectivity & security

Endpoint & port

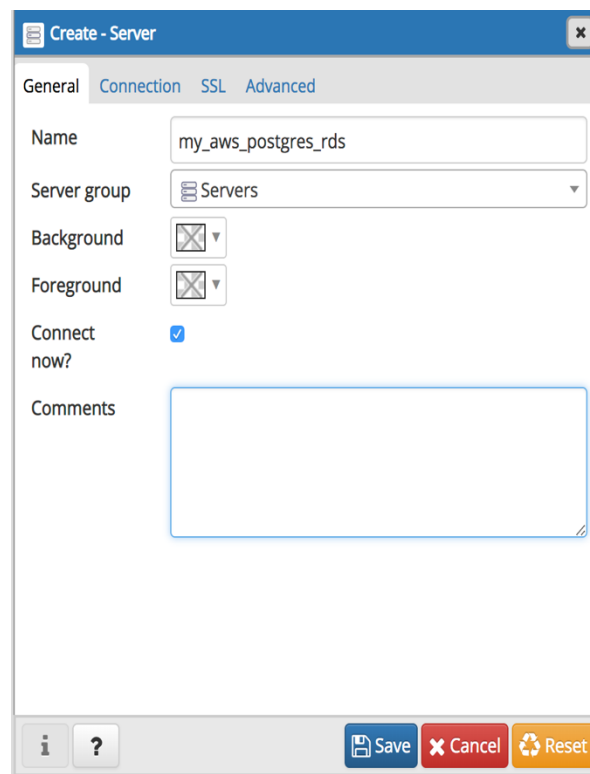
Endpoint

mypostgresdb.czu4ofportww.us-east-2.rds.amazonaws.com

Port

5432

- Open up pgAdmin, right-click **Servers**, and then go to “Create - Server”. Then, walk through the following steps to create a connection to the AWS RDS instance:
 - Under the “General” tab, enter the server name as my_aws_postgres_rds.



The screenshot shows the 'Create - Server' dialog box in pgAdmin. The 'General' tab is selected, and the 'Name' field is filled with 'my_aws_postgres_rds'. The 'Server group' is set to 'Servers'. The 'Background' and 'Foreground' checkboxes are both checked. The 'Connect now?' checkbox is also checked. The 'Comments' field is empty. At the bottom, there are buttons for 'Save', 'Cancel', and 'Reset'.

- Under the “Connection” tab, do the following:
 - Enter the endpoint in the “Hostname/address” field. This is unique to the instance.
 - Enter ‘postgres’ in the “Maintenance database” field. This is the default for all postgres RDS instances.
 - Enter the username in the “Username” field—in this case, root.
 - Enter the password that was created for your RDS instance.
 - Check the box next to “Save Password.”
- Click **Save**. If all information is entered correctly, this will set up the connection and not return an error.

The screenshot shows a dialog box titled "Create - Server" with a close button (X) in the top right corner. The dialog has five tabs: "General", "Connection", "SSL", "SSH Tunnel", and "Advanced". The "Connection" tab is selected. The form contains the following fields and controls:

- Host name/address:** A text field containing the value "mypostgresdb.cae1r8ifpdhe.us-east-1.rds.amazonaws".
- Port:** A text field containing the value "5432".
- Maintenance database:** A text field containing the value "postgres".
- Username:** A text field containing the value "root".
- Password:** A text field with masked characters (dots).
- Save password?:** A checkbox that is checked (indicated by a blue checkmark).
- Role:** An empty text field.
- Service:** An empty text field.


At the bottom of the dialog, there are three buttons: "Save" (blue), "Cancel" (red), and "Reset" (yellow). To the left of these buttons are two small icons: an information icon (i) and a question mark icon (?).

- To delete a database click on [Deleting a DB instance](#) and follow the instructions.

Deleting a DB instance

After you have connected to the sample DB instance that you created, you should delete the DB instance so you are no longer charged for it.

To delete a DB instance with no final DB snapshot

1. Sign in to the AWS Management Console and open the Amazon RDS console at <https://console.aws.amazon.com/rds/> .
2. In the navigation pane, choose **Databases**.
3. Choose the DB instance that you want to delete.
4. For **Actions**, choose **Delete**.
5. For **Create final snapshot?**, choose **No**, and select the acknowledgment.
6. Choose **Delete**.