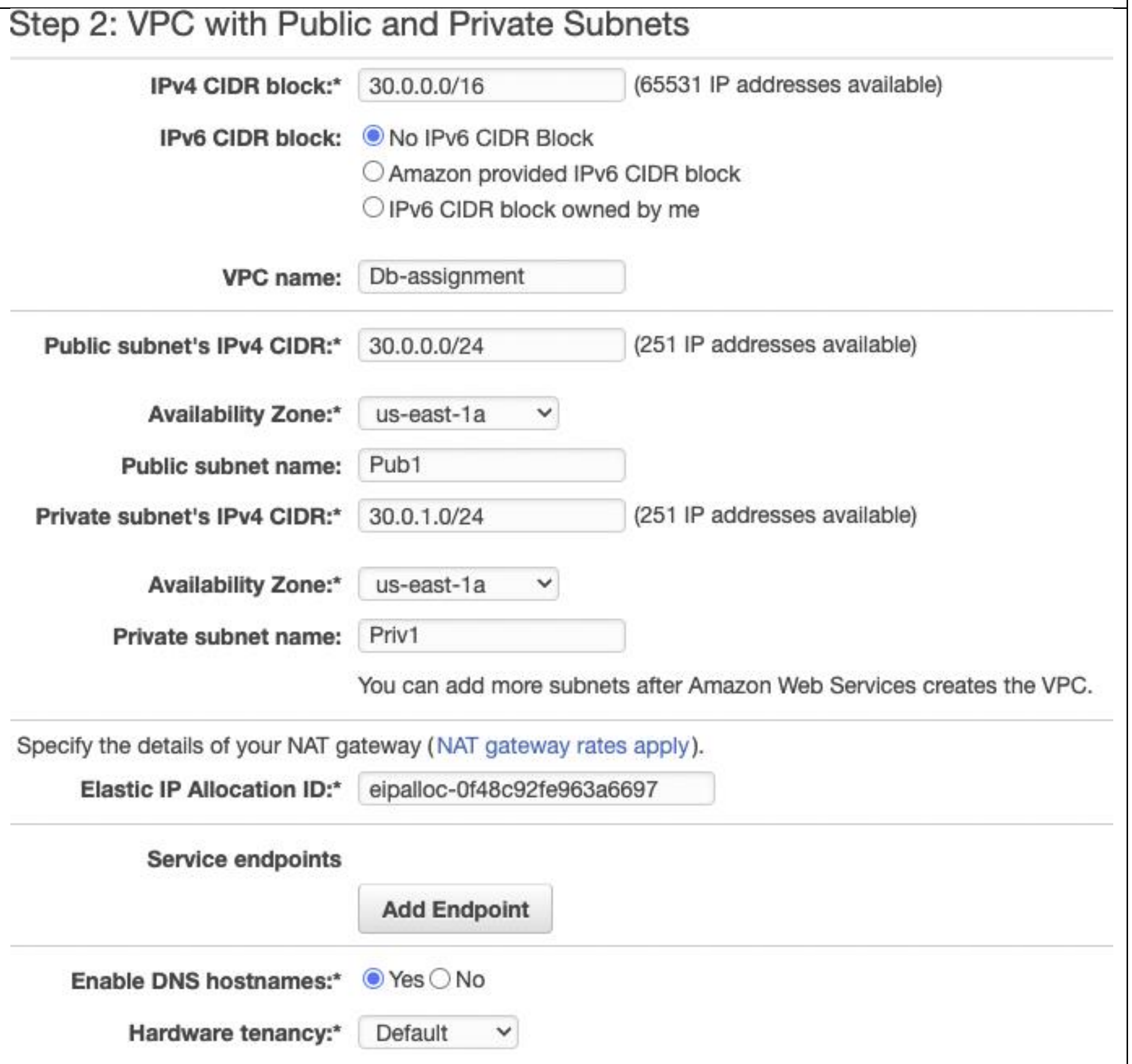
Name	Sisi Zhang, Harish Ram
Action	result

Create a VPC with Private and Public Subnets

Use the following procedure to create a VPC with both public and private subnets.

To create a VPC and subnets

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the top-right corner of the AWS Management Console, choose the region to create your VPC in. This example uses the Northern Virginia region.
3. In the upper-left corner, choose **VPC Dashboard**. To begin creating a VPC, choose **Launch VPC Wizard**.
4. On the **Step 1: Select a VPC Configuration** page, choose **VPC with Public and Private Subnets**, and then choose **Select**.
5. On the **Step 2: VPC with Public and Private Subnets** page, set these values:
   - **IPv4 CIDR block:** 30.0.0.0/16 (MAKE SURE YOU DO NOT HAVE ANOTHER PVC WITH similar CIDR Block)
   - **IPv6 CIDR block:** No IPv6 CIDR Block
   - **VPC name:** `tutorial-vpc`
   - **Public subnet's IPv4 CIDR:** `30.0.0.0/24`
   - **Availability Zone:** `us-east-2a`
   - **Public subnet name:** `TutPubSubNet`



Step 2: VPC with Public and Private Subnets

IPv4 CIDR block:* 30.0.0.0/16 (65531 IP addresses available)

IPv6 CIDR block: ● No IPv6 CIDR Block ○ Amazon provided IPv6 CIDR block ○ IPv6 CIDR block owned by me

VPC name: Db-assignment

Public subnet's IPv4 CIDR:* 30.0.0.0/24 (251 IP addresses available)

Availability Zone:* us-east-1a

Public subnet name: Pub1

Private subnet's IPv4 CIDR:* 30.0.1.0/24 (251 IP addresses available)

Availability Zone:* us-east-1a

Private subnet name: Priv1

You can add more subnets after Amazon Web Services creates the VPC.

Specify the details of your NAT gateway (NAT gateway rates apply).

Elastic IP Allocation ID:* eipalloc-0f48c92fe963a6697

Service endpoints

Enable DNS hostnames:* ● Yes ○ No

Hardware tenancy:* Default

- **Private subnet's IPv4 CIDR:** `30.0.1.0/24`
- **Availability Zone:** `us-east-2a`
- **Private subnet name:** `TutPriSutNet-1`
- **Instance type:** `t2.micro`

  Important

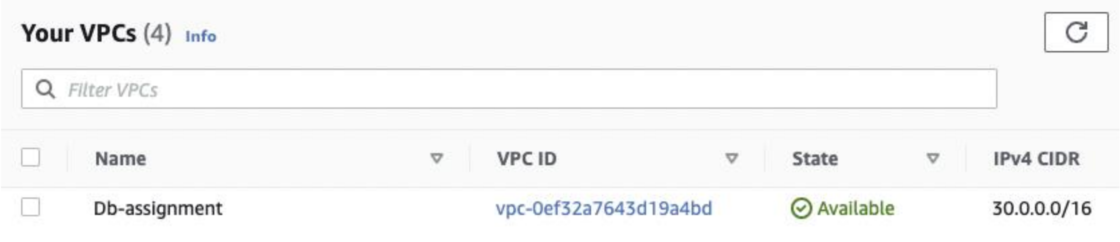
  If you do not see the **Instance type** box in the console, click **Use a NAT instance instead**. This link is on the right.

  Note

  If the t2.micro instance type is not listed, you can select a different instance type.
- **Key pair name:** `use your key pair that you have previously downloaded`
- **Service endpoints:** Skip this field.
- **Enable DNS hostnames:** `Yes`
- **Hardware tenancy:** `Default`

6. When you're finished, choose **Create VPC**

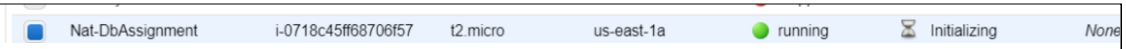
A new VPC will be created



Your VPCs (4) Info

Name	VPC ID	State	IPv4 CIDR
Db-assignment	vpc-0ef32a7643d19a4bd	Available	30.0.0.0/16

. In Addition, a new EC2 instance will also be created. This instance will be your NAT. Given a name to this new instance, e.g., Nat-DbAssignment



Nat-DbAssignment i-0718c45ff68706f57 t2.micro us-east-1a ● running ⌛ Initializing *None*

This is not happening

Create Additional Subnets

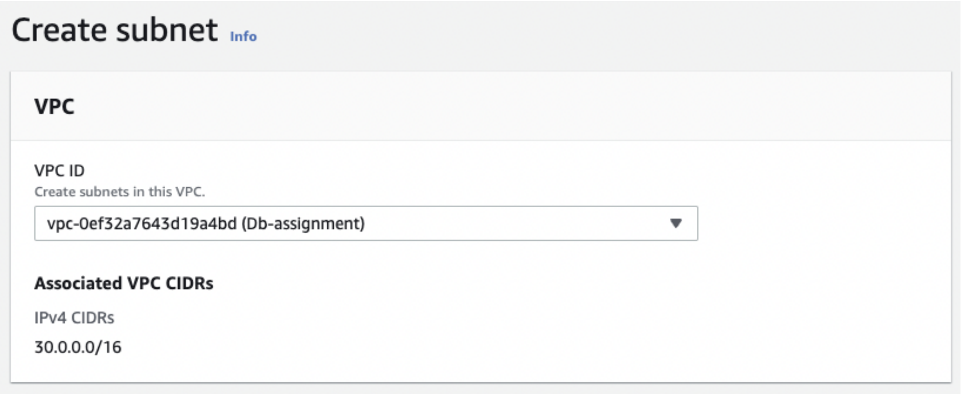
You must have either two private subnets or two public subnets available to create an Amazon RDS DB subnet group for an RDS DB instance to use in a VPC. Because the RDS DB instance for this tutorial is private, add a second private subnet to the VPC.

To create an additional subnet

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. To add the second private subnet to your VPC, choose **VPC Dashboard**, choose **Subnets**, and then choose **Create subnet**.
3. On the **Create subnet** page, set these values:
   - **Name tag:** `TutPriSubNet-2`
   - **VPC:** Choose the VPC that you created in the previous step, for example: `vpc-`*`identifier`*` (30.0.0.0/16) | tutorial-vpc`
   - **Availability Zone:** `us-east-1b`

     Note

     Choose an Availability Zone that is different from the one that you chose for the first private subnet.



Create subnet

VPC

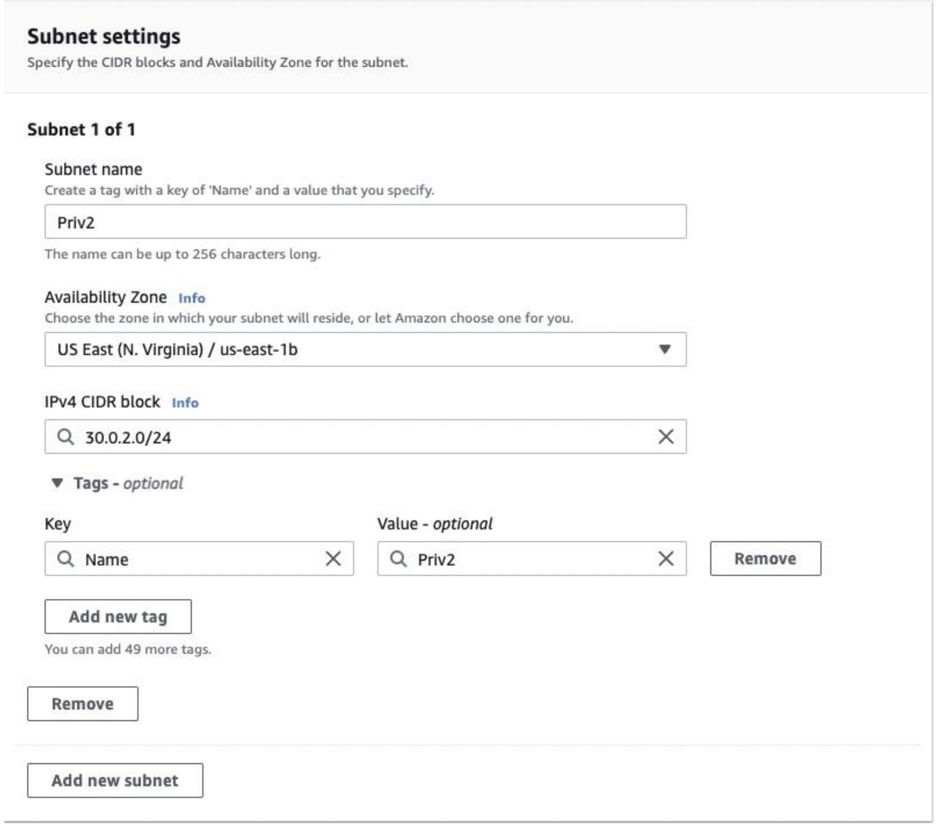
VPC ID
Create subnets in this VPC.
vpc-0ef32a7643d19a4bd (Db-assignment) ▼

Associated VPC CIDRs

IPv4 CIDRs
30.0.0.0/16

- **IPv4 CIDR block: 30.0.2.0/24** `(Be careful! Use a CIDR block compatible with your VPC and different from the previous subnets`

4. When you're finished, choose **Create**. Next, choose **Close** on the confirmation page.
5.



Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name

Create a tag with a key of 'Name' and a value that you specify.

Priv2

The name can be up to 256 characters long.

Availability Zone Info

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

US East (N. Virginia) / us-east-1b ▼

IPv4 CIDR block Info

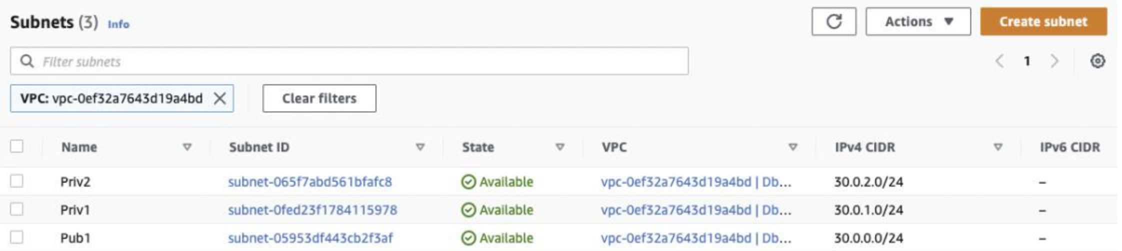
30.0.2.0/24

▼ Tags - *optional*

Key	Value - *optional*	
Name	Priv2	Remove

You can add 49 more tags.

In your dashboard you will see the new subnet

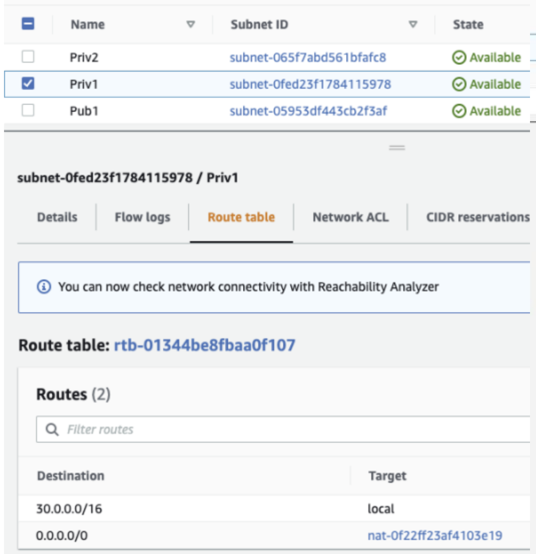


Subnets (3) Info

VPC: vpc-0ef32a7643d19a4bd

□	Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
□	Priv2	subnet-065f7abd561bfafc8	✓ Available	vpc-0ef32a7643d19a4bd | Db...	30.0.2.0/24	–
□	Priv1	subnet-0fed23f1784115978	✓ Available	vpc-0ef32a7643d19a4bd | Db...	30.0.1.0/24	–
□	Pub1	subnet-05953df443cb2f3af	✓ Available	vpc-0ef32a7643d19a4bd | Db...	30.0.0.0/24	–

1. To ensure that the second private subnet that you created uses the same route table as the first private subnet, choose **VPC Dashboard**, choose **Subnets**, and then choose the first private subnet that you created for the VPC, TutPrivSubNet-1. Below the list of subnets, choose the **Route Table** tab, and note the value for **Route Table**—for example: rtb-00d84cf5ff0e6f409.
2. In the list of subnets, deselect the first private subnet.
3. In the list of subnets, choose the second private subnet **TutPriSubNet-2,** and choose the **Route Table** tab.
4. If the current route table is not the same as the route table for the first private subnet, choose **Edit route table association**. For **Route Table ID**, choose the route table that you noted earlier—for example: rtb-`00d84cf5ff0e6f409. Next, to save your selection, choose **Save**.



	Name	Subnet ID	State
☐	Priv2	subnet-065f7abd561bfafc8	Available
☑	Priv1	subnet-0fed23f1784115978	Available
☐	Pub1	subnet-05953df443cb2f3af	Available

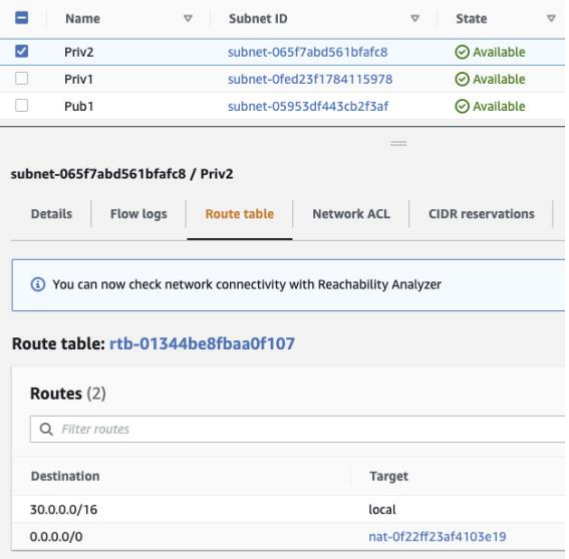
subnet-0fed23f1784115978 / Priv1

You can now check network connectivity with Reachability Analyzer

Route table: rtb-01344be8fbaa0f107

Routes (2)

Destination	Target
30.0.0.0/16	local
0.0.0.0/0	nat-0f22ff23af4103e19



	Name	Subnet ID	State
☑	Priv2	subnet-065f7abd561bfafc8	Available
☐	Priv1	subnet-0fed23f1784115978	Available
☐	Pub1	subnet-05953df443cb2f3af	Available

subnet-065f7abd561bfafc8 / Priv2

You can now check network connectivity with Reachability Analyzer

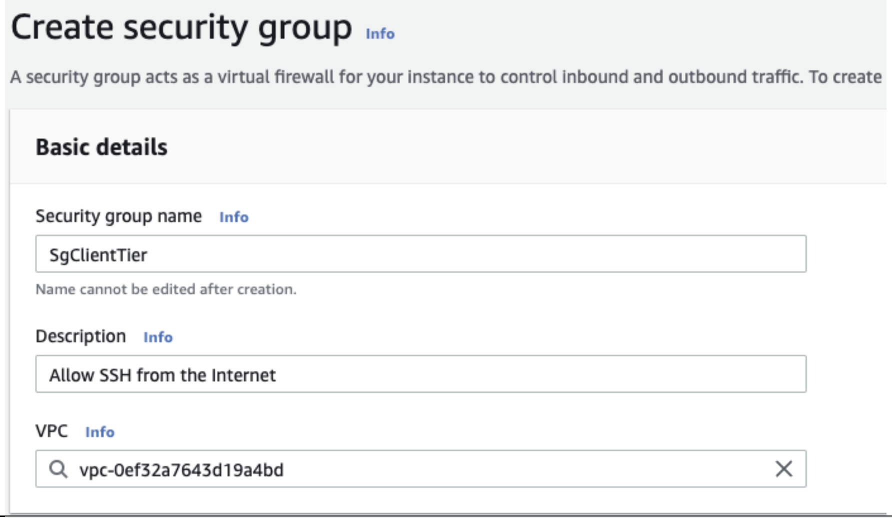
Route table: rtb-01344be8fbaa0f107

Routes (2)

Destination	Target
30.0.0.0/16	local
0.0.0.0/0	nat-0f22ff23af4103e19

Create a VPC Security Group for a Public Web Server

Next you create a security group for public access. To connect to public instances in your VPC, you add inbound rules to your VPC security group that allow traffic to connect from the internet.



Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create

Basic details

Security group name Info

SgClientTier

Name cannot be edited after creation.

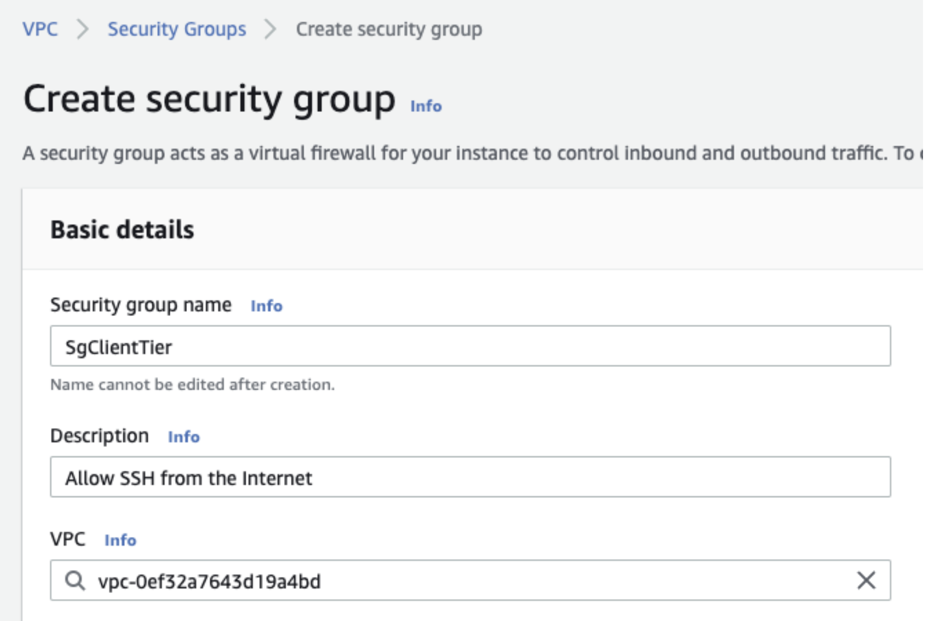
Description Info

Allow SSH from the Internet

VPC Info

vpc-0ef32a7643d19a4bd

Create a Security Group that will be used by the servers launched in the public subnet, i.e., that will be accessible via internet



VPC > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To

Basic details

Security group name Info

SgClientTier

Name cannot be edited after creation.

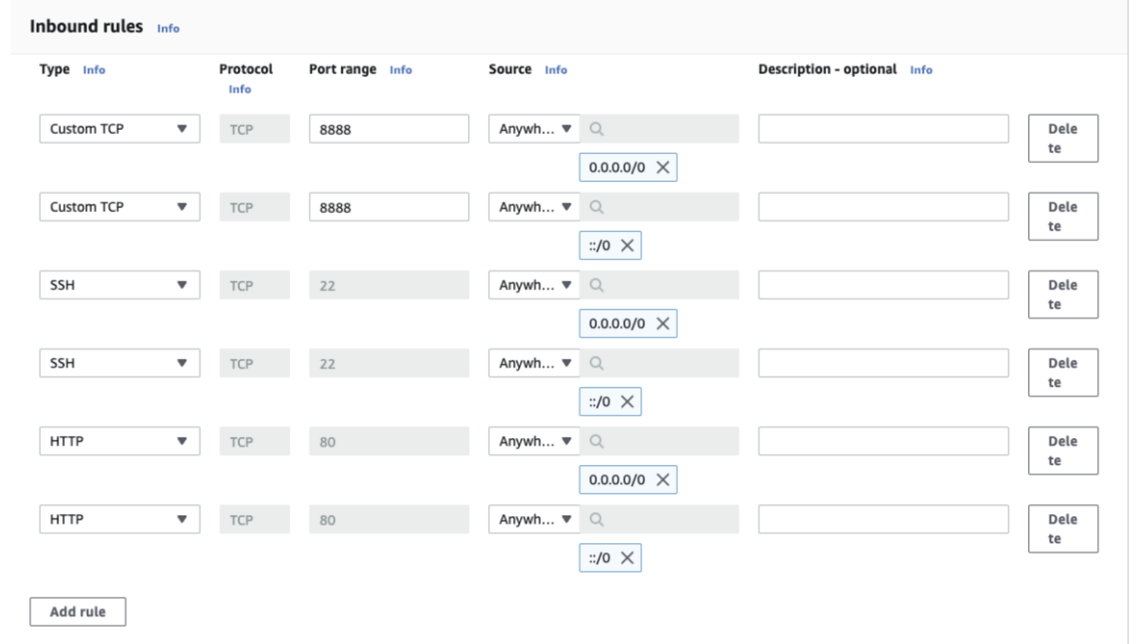
Description Info

Allow SSH from the Internet

VPC Info

vpc-0ef32a7643d19a4bd

Add inbound rules to the security group. Make sure ports 22 and 8888 are open from anywhere

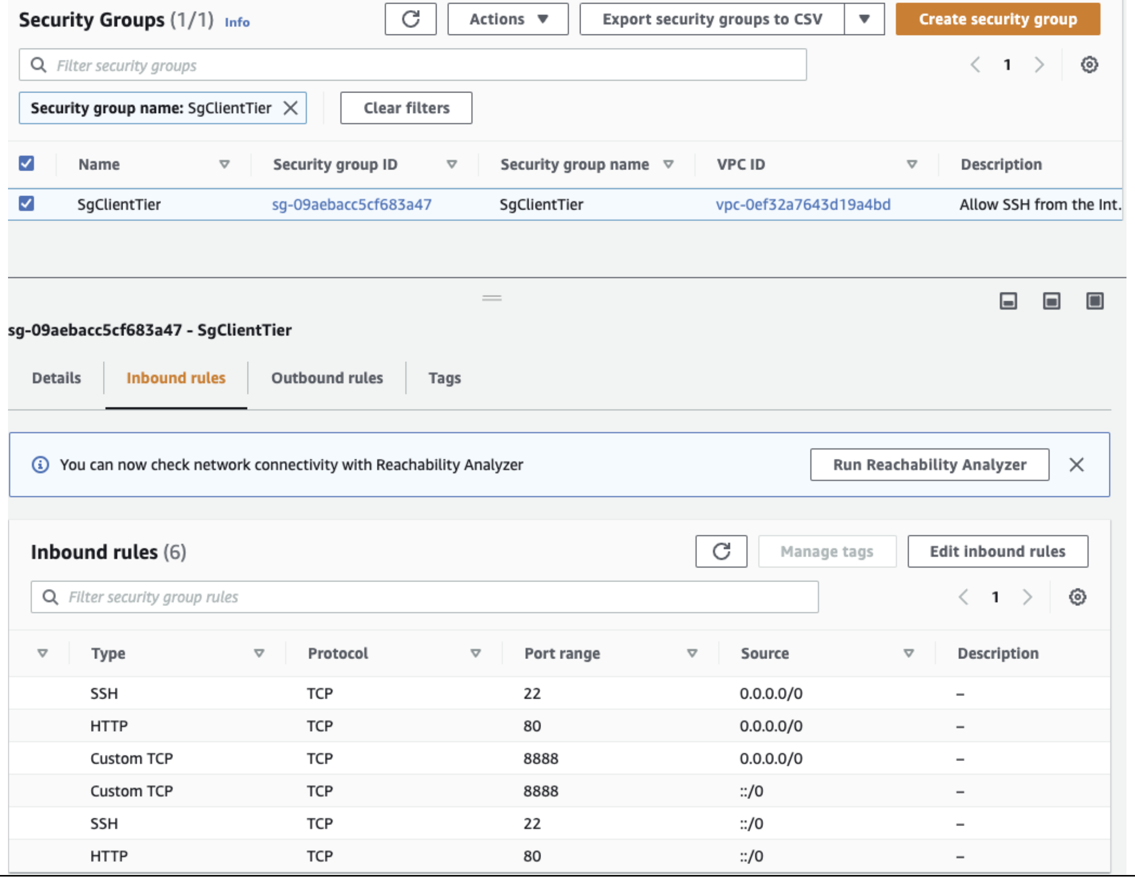


Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
Custom TCP ▼	TCP	8888	Anywh... ▼	0.0.0.0/0 ✕		Delete
Custom TCP ▼	TCP	8888	Anywh... ▼	::/0 ✕		Delete
SSH ▼	TCP	22	Anywh... ▼	0.0.0.0/0 ✕		Delete
SSH ▼	TCP	22	Anywh... ▼	::/0 ✕		Delete
HTTP ▼	TCP	80	Anywh... ▼	0.0.0.0/0 ✕		Delete
HTTP ▼	TCP	80	Anywh... ▼	::/0 ✕		Delete

Add rule

The new SG will be linked to the VPC you created previously (Note: Make sure the VPC ID matches the VPC ID of the VPC you created in this tutorial)



Security Groups (1/1)

Security group name: SgClientTier

☑	Name	Security group ID	Security group name	VPC ID	Description
☑	SgClientTier	sg-09aebacc5cf683a47	SgClientTier	vpc-0ef32a7643d19a4bd	Allow SSH from the Int.

sg-09aebacc5cf683a47 - SgClientTier

You can now check network connectivity with Reachability Analyzer

Inbound rules (6)

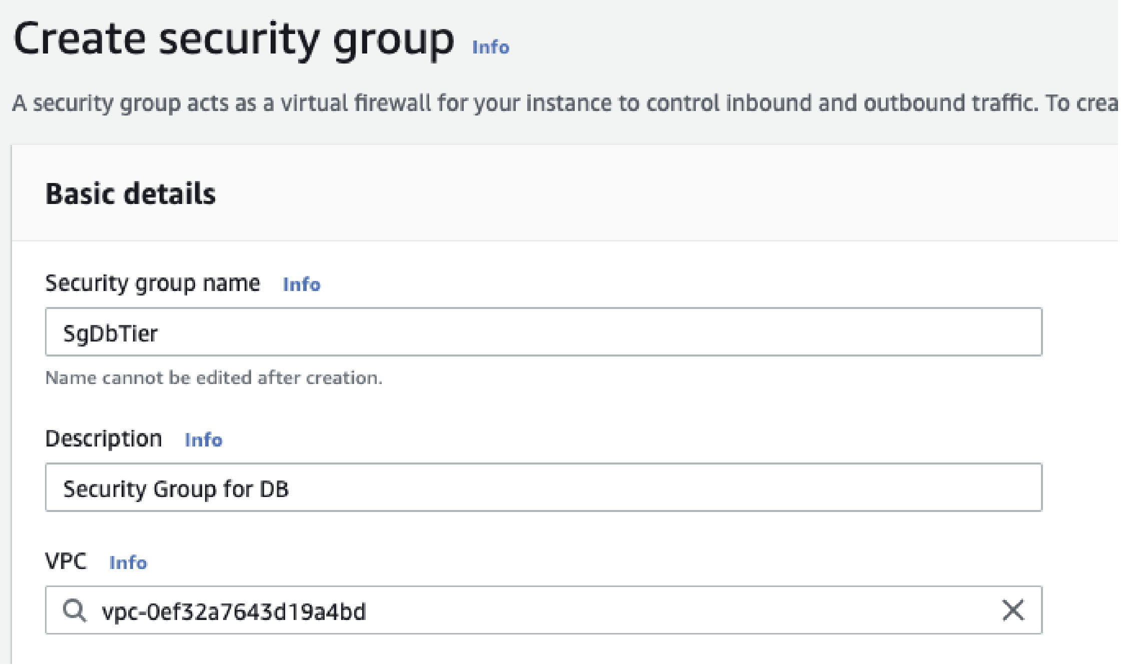
Type	Protocol	Port range	Source	Description
SSH	TCP	22	0.0.0.0/0	–
HTTP	TCP	80	0.0.0.0/0	–
Custom TCP	TCP	8888	0.0.0.0/0	–
Custom TCP	TCP	8888	::/0	–
SSH	TCP	22	::/0	–
HTTP	TCP	80	::/0	–

Create a VPC Security Group for a Private Amazon RDS DB Instance

To keep your Amazon RDS DB instance private, create a second security group for private access. To connect to private instances in your VPC, you add inbound rules to your VPC security group that allow traffic from your web server only.

To create a VPC security group

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. Choose **VPC Dashboard**, choose **Security Groups**, and then choose **Create security group**.
3. On the **Create security group** page, set these values:
    - **Security group name:** DbAssignmentDatabaseSG
    - **Description:** `Tutorial DB Instance Security Group`
    - **VPC:** Choose the VPC that you created earlier, for example: Db-Assignment - vpc-021595493537934b1 - 30.0.0.0/16
4. To create the security group, choose **Create**. Next, choose **Close** on the confirmation page.



Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To crea

Basic details

Security group name Info

SgDbTier

Name cannot be edited after creation.

Description Info

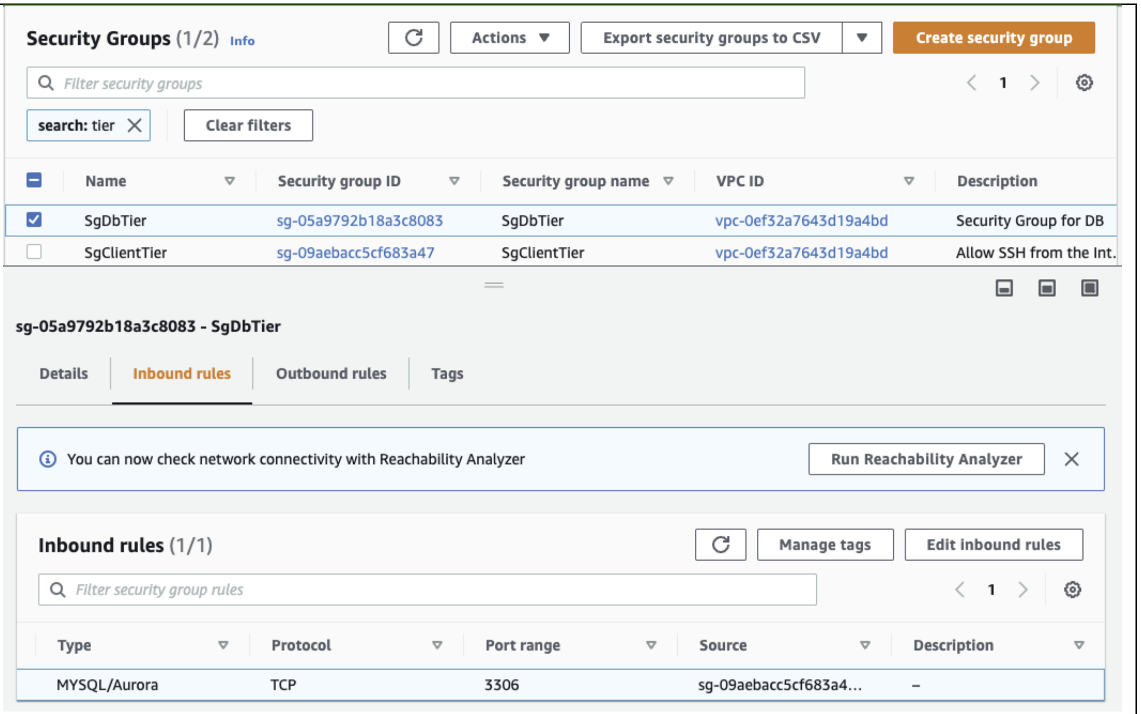
Security Group for DB

VPC Info

vpc-0ef32a7643d19a4bd

To add inbound rules to the security group

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. Choose **VPC Dashboard**, choose **Security Groups**, and then choose the DbAssignmentDatabaseSG security group that you created in the previous procedure.
3. Under the list of security groups, choose the **Inbound Rules** tab, and then choose **Edit rules**.
4. On the **Edit inbound rules** page, choose **Add Rule**.
5. Set the following values for your new inbound rule to allow MySQL traffic on port 3306 from your EC2 instance. If you do this, you can connect from your web server to your DB instance to store and retrieve data from your web application to your database.
   - **Type:** `MySQL/Aurora`
   - **Source:** The identifier of the `tutorial-securitygroup` security group that you created previously in this tutorial, for example: `sg-082febcae29151850`
   - `( DbAssignmentClientSG /`
   - `DbAssignmentClientServerSG`
   - `vpc-021595493537934b1 )`
6. To save your settings, choose **Save rules**. Next, choose **Close** on the confirmation page



☐	Name	Security group ID	Security group name	VPC ID	Description
☑	SgDbTier	sg-05a9792b18a3c8083	SgDbTier	vpc-0ef32a7643d19a4bd	Security Group for DB
☐	SgClientTier	sg-09aebacc5cf683a47	SgClientTier	vpc-0ef32a7643d19a4bd	Allow SSH from the Int.

sg-05a9792b18a3c8083 - SgDbTier

You can now check network connectivity with Reachability Analyzer

Inbound rules (1/1)

Type	Protocol	Port range	Source	Description
MYSQL/Aurora	TCP	3306	sg-09aebacc5cf683a4...	–

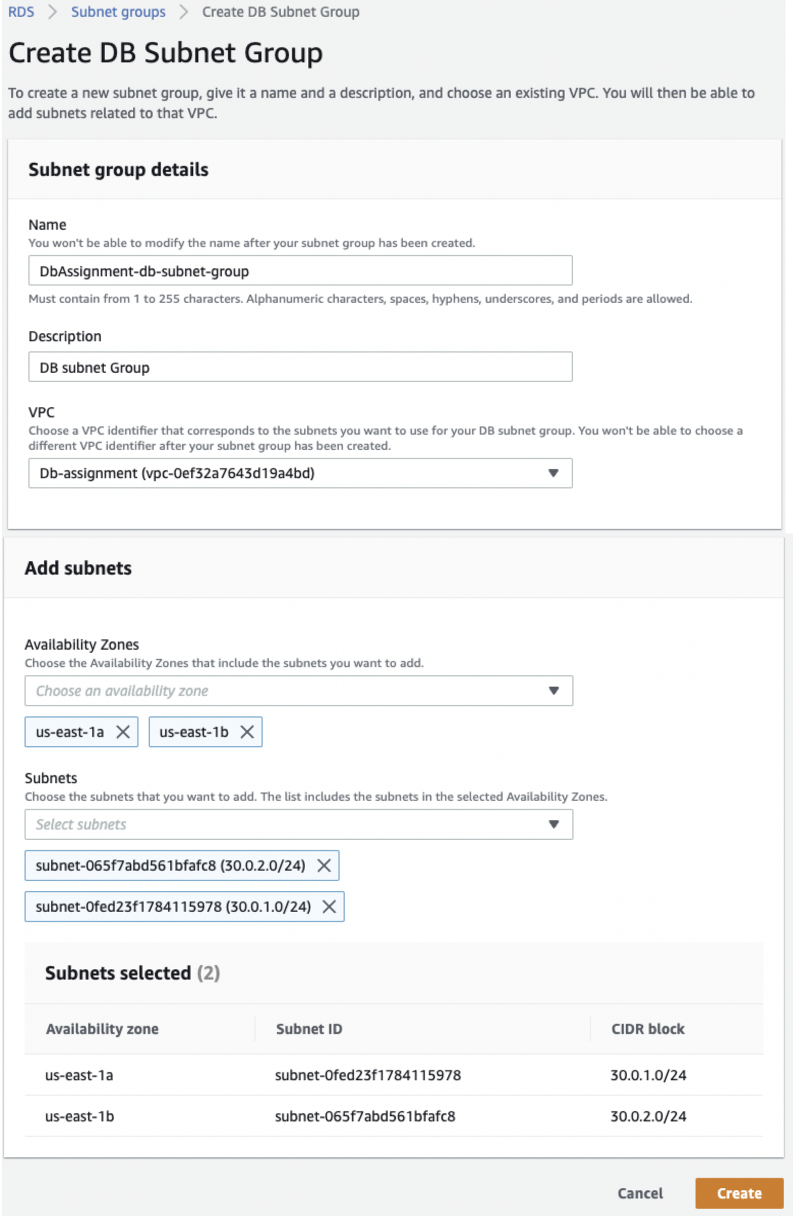
Create a DB Subnet Group

A DB subnet group is a collection of subnets that you create in a VPC and that you then designate for your DB instances. A DB subnet group allows you to specify a particular VPC when creating DB instances.

To create a DB subnet group

1. Open the Amazon RDS console at https://console.aws.amazon.com/rds/.
2. In the navigation pane, choose **Subnet groups**.
3. Choose **Create DB Subnet Group**.
4. On the **Create DB subnet group** page, set these values in **Subnet group details**:
   - **Name:** `DbAssignment-db-subnet-group`
   - **Description:** `Tutorial DB Subnet Group`
   - **VPC:** `tutorial-vpc (vpc-`*`identifier`*`)`
5. In the **Add subnets** section, choose **Add all the subnets related to this VPC**.
6. Remove the public subnet from the list (only keep the 2 private subnets you created in this tutorial)
7. Choose **Create**.

   Your new DB subnet group appears in the DB subnet groups list on the RDS console. You can click the DB subnet group to see details, including all of the



Create DB Subnet Group

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

Subnet group details

Name
You won't be able to modify the name after your subnet group has been created.
DbAssignment-db-subnet-group
Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description
DB subnet Group

VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.
Db-assignment (vpc-0ef32a7643d19a4bd) ▼

Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone ▼
us-east-1a ✕ us-east-1b ✕

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets ▼
subnet-065f7abd561bfafc8 (30.0.2.0/24) ✕
subnet-0fed23f1784115978 (30.0.1.0/24) ✕

Subnets selected (2)

Availability zone	Subnet ID	CIDR block
us-east-1a	subnet-0fed23f1784115978	30.0.1.0/24
us-east-1b	subnet-065f7abd561bfafc8	30.0.2.0/24

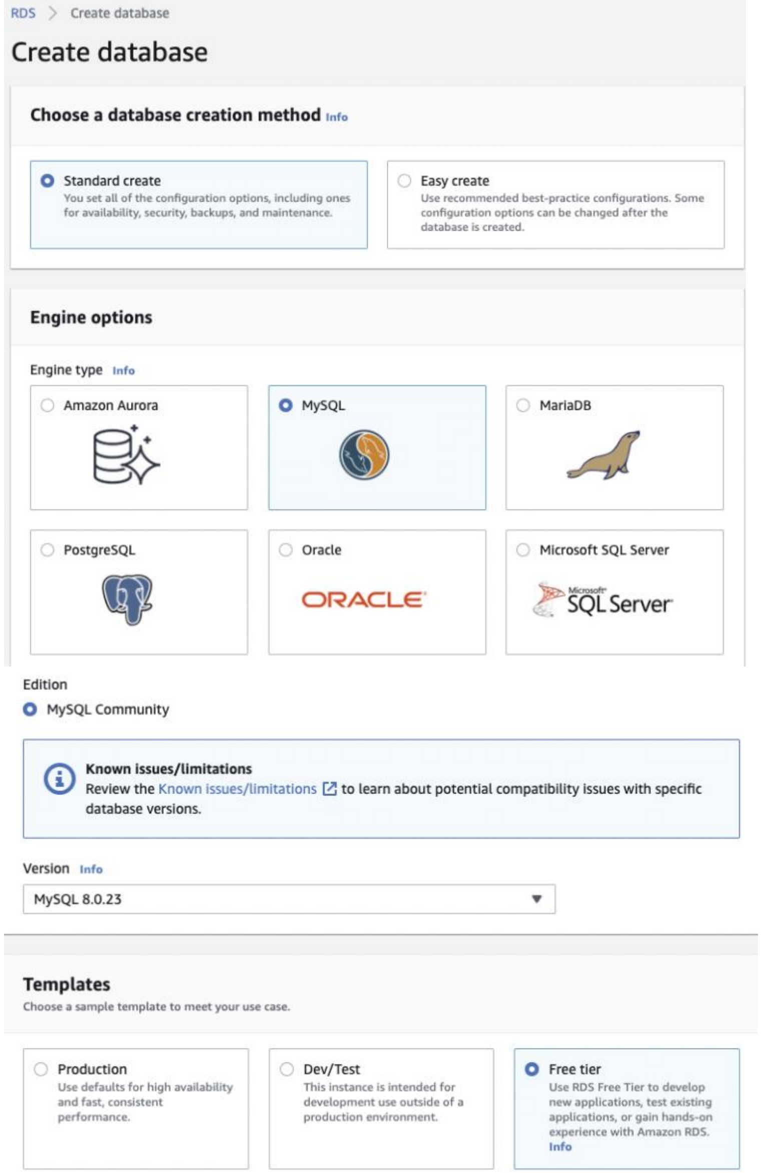
subnets associated with the group, in the details pane at the bottom of the window.	

To launch a MySQL DB instance

1. Sign in to the AWS Management Console and open the Amazon RDS console at https://console.aws.amazon.com/rds/.
2. In the top right corner of the AWS Management Console, choose the AWS Region in which you want to create the DB instance.
3. In the navigation pane, choose **Databases**.

   If the navigation pane is closed, choose the menu icon at the top left to open it.

4. Choose **Create database** to open the **Select engine** page.



RDS > 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.

Engine options

Engine type Info

○ Amazon Aurora
● MySQL
○ MariaDB
○ PostgreSQL
○ Oracle
○ Microsoft SQL Server

Edition
● MySQL Community

Known issues/limitations
Review the Known issues/limitations to learn about potential compatibility issues with specific database versions.

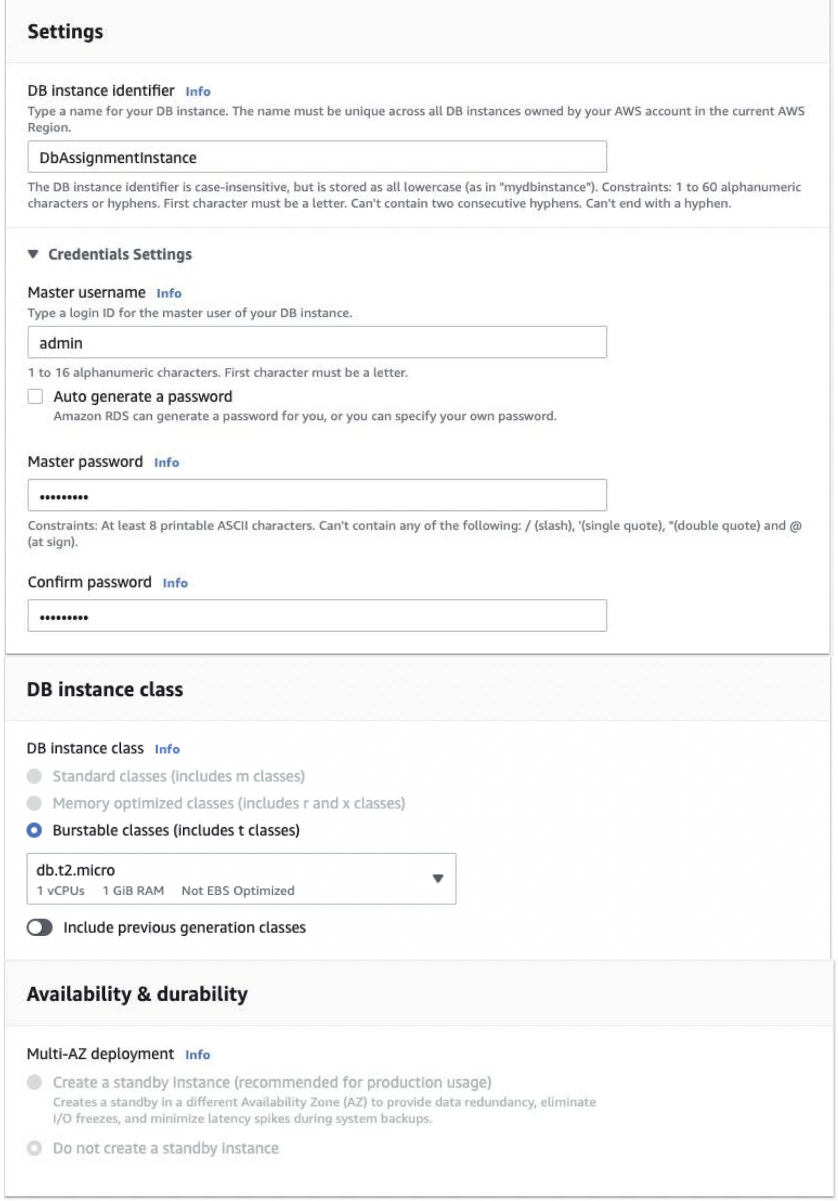
Version Info
MySQL 8.0.23

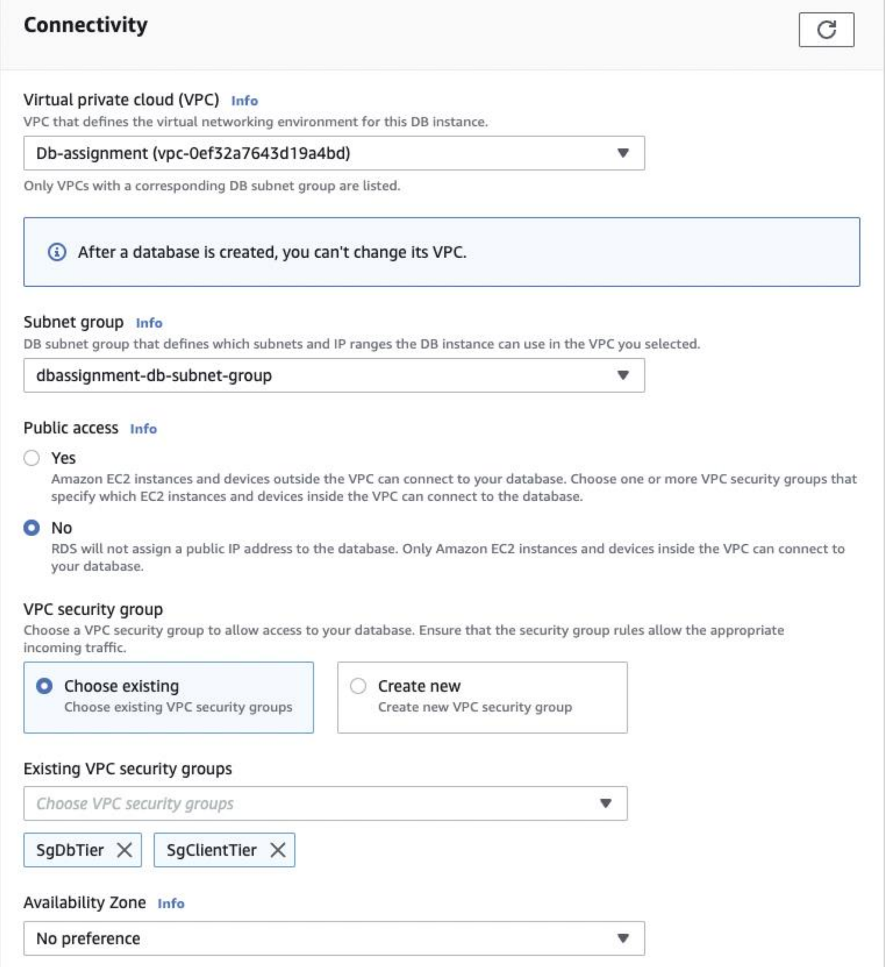
Templates
Choose a sample template to meet your use case.

○ 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

Accept the default parameters. Provide the name of the dbinstance, e.g., DbAssignmentInstance, username and password of the DB administrator. This is a new username and password that do not necessary match the users/passwords of the users defined in IAM	 Settings DB instance identifier Info 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. DbAssignmentInstance The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen. ▼ Credentials Settings Master username Info Type a login ID for the master user of your DB instance. admin 1 to 16 alphanumeric characters. First character must be a letter. □ Auto generate a password Amazon RDS can generate a password for you, or you can specify your own password. Master password Info ••••••••• Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign). Confirm password Info ••••••••• DB instance class DB instance class Info ○ Standard classes (includes m classes) ○ Memory optimized classes (includes r and x classes) ● Burstable classes (includes t classes) db.t2.micro 1 vCPUs 1 GiB RAM Not EBS Optimized Include previous generation classes Availability & durability Multi-AZ deployment Info ○ Create a standby instance (recommended for production usage) Creates a standby in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups. ○ Do not create a standby instance

Select the VPC you created in this tutorial Select the subnet group you created in this tutorial Make sure the DB is NOT public accessible Select an AZ Select the VPC Security Group you created in this tutorial, ie, DbAssignmentDatabaseSG	

Connectivity

Virtual private cloud (VPC) Info

VPC that defines the virtual networking environment for this DB instance.

Db-assignment (vpc-0ef32a7643d19a4bd) ▼

Only VPCs with a corresponding DB subnet group are listed.

ⓘ After a database is created, you can't change its VPC.

Subnet group Info

DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

dbassignment-db-subnet-group ▼

Public access Info

○ Yes

Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

● No

RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group

Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

● Choose existing
Choose existing VPC security groups

○ Create new
Create new VPC security group

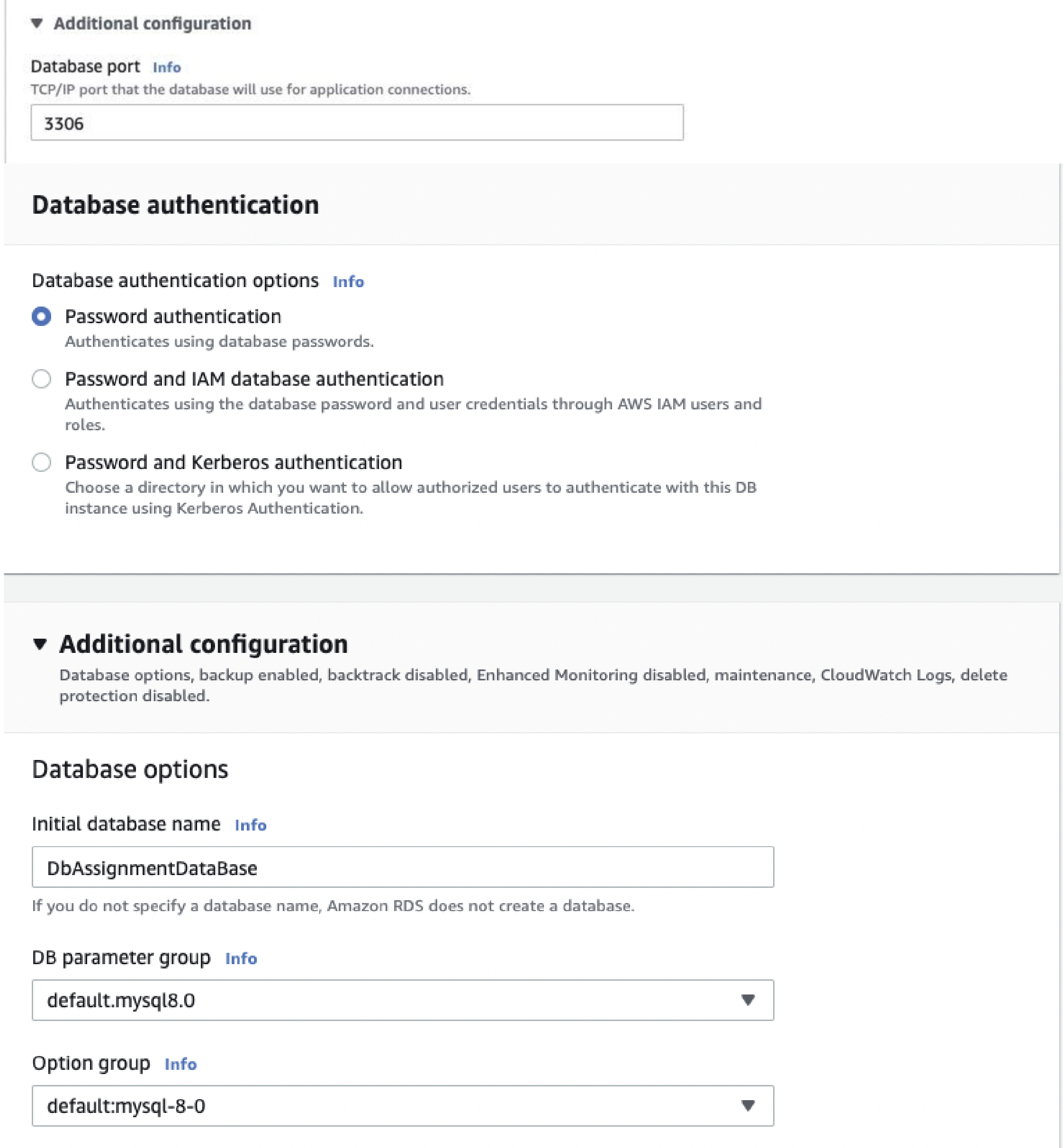
Existing VPC security groups

Choose VPC security groups ▼

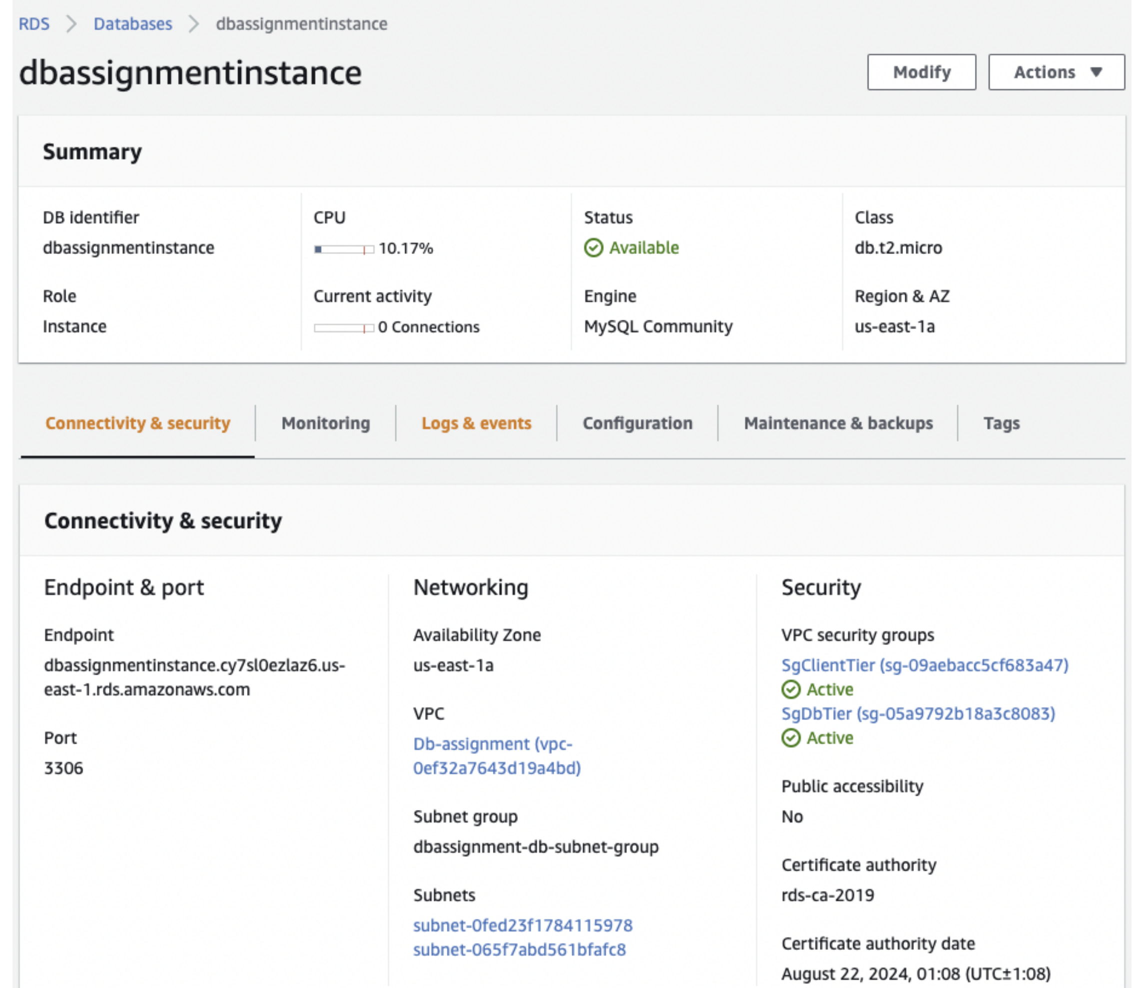
SgDbTier ✕ SgClientTier ✕

Availability Zone Info

No preference ▼

Provide a DB name. Use the default port, 3306 In this tutorial disable IAM DB authentication Access the default options to the other fields Create the DB	 Database port TCP/IP port that the database will use for application connections. 3306 **Database authentication** Database authentication options ● 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. Database options, backup enabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete protection disabled. Database options Initial database name DbAssignmentDataBase If you do not specify a database name, Amazon RDS does not create a database. DB parameter group default.mysql8.0 ▼ Option group default:mysql-8-0 ▼

You DB is created



dbassignmentinstance

Summary

DB identifier	CPU	Status	Class
dbassignmentinstance	10.17%	✓ Available	db.t2.micro
Role	**Current activity**	**Engine**	**Region & AZ**
Instance	0 Connections	MySQL Community	us-east-1a

Connectivity & security

Endpoint & port

Endpoint
dbassignmentinstance.cy7sl0ezlaz6.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1a

VPC
Db-assignment (vpc-0ef32a7643d19a4bd)

Subnet group
dbassignment-db-subnet-group

Subnets
subnet-0fed23f1784115978
subnet-065f7abd561bfafc8

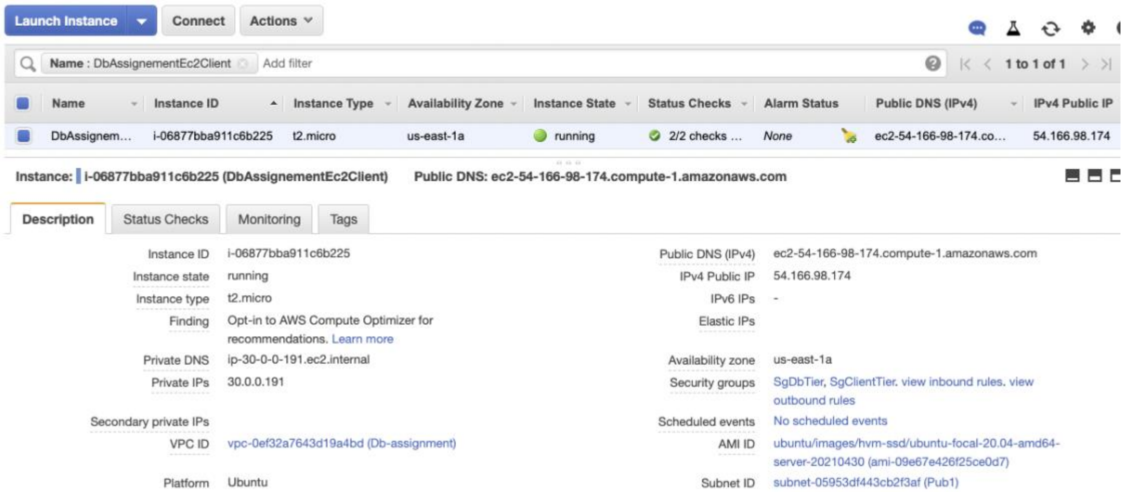
Security

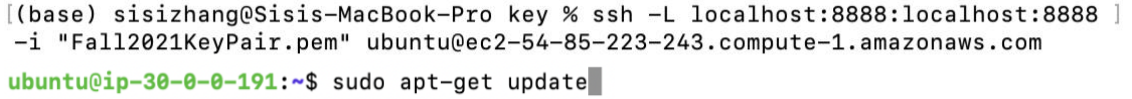
VPC security groups
SgClientTier (sg-09aebacc5cf683a47)
✓ Active
SgDbTier (sg-05a9792b18a3c8083)
✓ Active

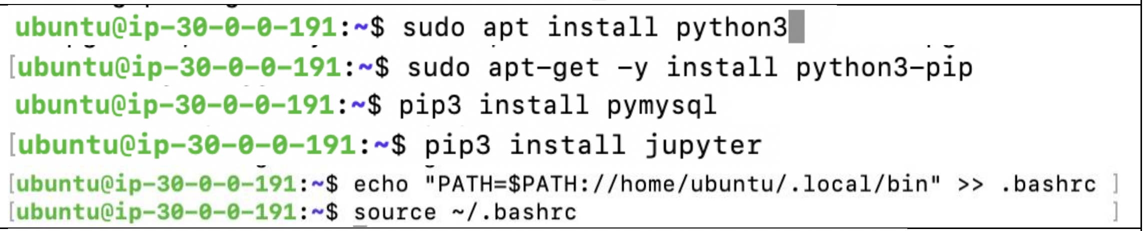
Public accessibility
No

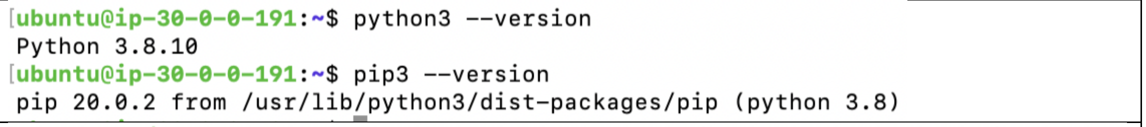
Certificate authority
rds-ca-2019

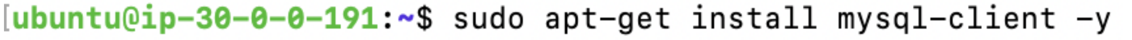
Certificate authority date
August 22, 2024, 01:08 (UTC±1:08)

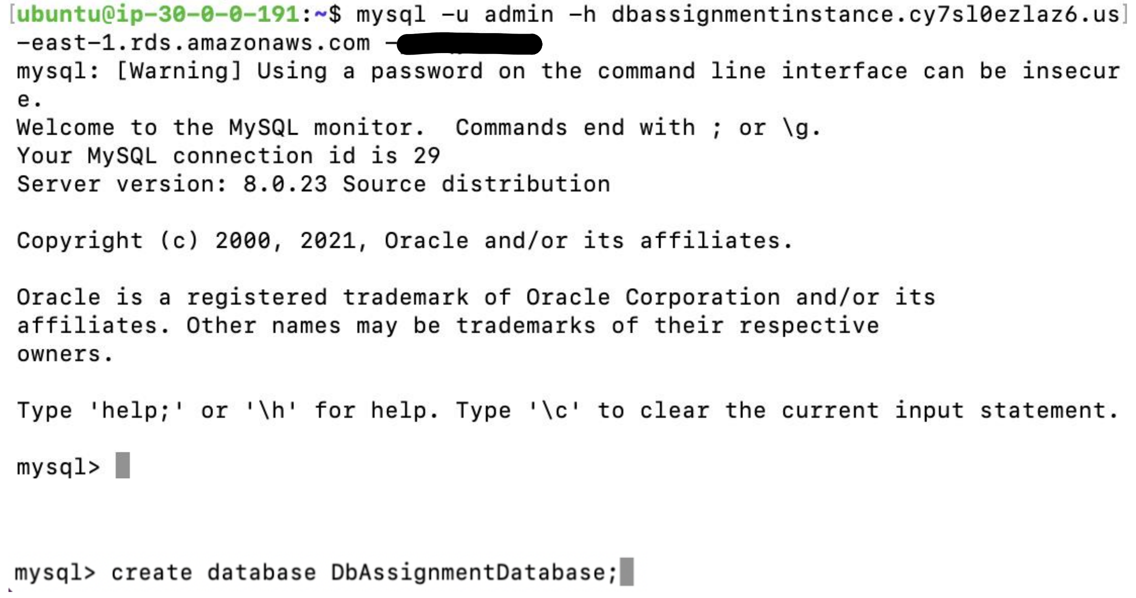
Launch an Ubuntu EC2 inside the public subnet-1 in the same VPC you launched the MySql DB. Use the DbAssignmentClientServerSG as security group	 Launch Instance · Connect · Actions Name : DbAssignementEc2Client · Add filter · 1 to 1 of 1 Name · Instance ID · Instance Type · Availability Zone · Instance State · Status Checks · Alarm Status · Public DNS (IPv4) · IPv4 Public IP DbAssignem... · i-06877bba911c6b225 · t2.micro · us-east-1a · running · 2/2 checks ... · None · ec2-54-166-98-174.co... · 54.166.98.174 Instance: i-06877bba911c6b225 (DbAssignementEc2Client) Public DNS: ec2-54-166-98-174.compute-1.amazonaws.com Description · Status Checks · Monitoring · Tags Instance ID: i-06877bba911c6b225 Instance state: running Instance type: t2.micro Finding: Opt-in to AWS Compute Optimizer for recommendations. Learn more Private DNS: ip-30-0-0-191.ec2.internal Private IPs: 30.0.0.191 Secondary private IPs: VPC ID: vpc-0ef32a7643d19a4bd (Db-assignment) Platform: Ubuntu Public DNS (IPv4): ec2-54-166-98-174.compute-1.amazonaws.com IPv4 Public IP: 54.166.98.174 IPv6 IPs: - Elastic IPs: Availability zone: us-east-1a Security groups: SgDbTier, SgClientTier. view inbound rules. view outbound rules Scheduled events: No scheduled events AMI ID: ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20210430 (ami-09e67e426f25ce0d7) Subnet ID: subnet-05953df443cb2f3af (Pub1)

SSH to the client EC2 instance
Update ubuntu packages
	
	`[(base) sisizhang@Sisis-MacBook-Pro key % ssh -L localhost:8888:localhost:8888 ]`
	`-i "Fall2021KeyPair.pem" ubuntu@ec2-54-85-223-243.compute-1.amazonaws.com`
	`ubuntu@ip-30-0-0-191:~$ sudo apt-get update`

Install python3 , pip3, pymysql, and jupyter	
	`ubuntu@ip-30-0-0-191:~$ sudo apt install python3`
	`[ubuntu@ip-30-0-0-191:~$ sudo apt-get -y install python3-pip`
	`ubuntu@ip-30-0-0-191:~$ pip3 install pymysql`
	`[ubuntu@ip-30-0-0-191:~$ pip3 install jupyter`
	`[ubuntu@ip-30-0-0-191:~$ echo "PATH=$PATH://home/ubuntu/.local/bin" >> .bashrc ]`
	`[ubuntu@ip-30-0-0-191:~$ source ~/.bashrc ]`

Make sure you have version 3 installed	
	`[ubuntu@ip-30-0-0-191:~$ python3 --version`
	`Python 3.8.10`
	`[ubuntu@ip-30-0-0-191:~$ pip3 --version`
	`pip 20.0.2 from /usr/lib/python3/dist-packages/pip (python 3.8)`

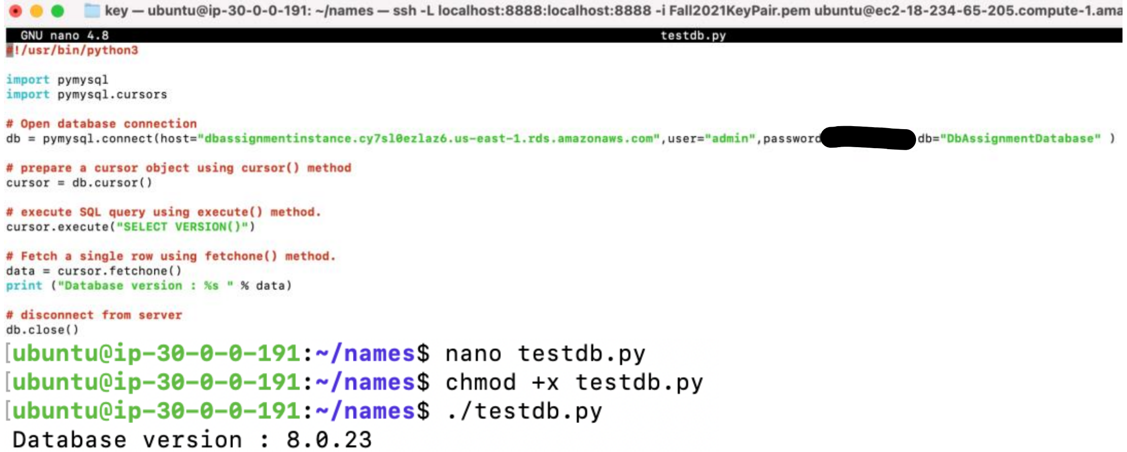
Now, let us test if you connect to the mysql database using a mysql client. To do so: a) Install mysql-client b) Connect to the mysql-server c) Create a database	 `[ubuntu@ip-30-0-0-191:~$ sudo apt-get install mysql-client -y`



```
[ubuntu@ip-30-0-0-191:~$ mysql -u admin -h dbassignmentinstance.cy7sl0ezlaz6.us]
-east-1.rds.amazonaws.com -[redacted]
mysql: [Warning] Using a password on the command line interface can be insecur
e.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 8.0.23 Source distribution

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 


mysql> create database DbAssignmentDatabase;
```

Test your DB connection using a python program deployed in your client EC2 instance
Python3 testdb.py
You can create a Python program using nano or vi. You can also use your laptop to create the program and after transfer it to the EC2 using a secure ftp. You can also install Jupyter and use Jupyter to edit the program (easier one). Or you can use a text editor that allows remote editing from your latptop, e.g., Notepad++.

Your connect string must contain the public ip address of your DB. User name and password should be the one your created previously.

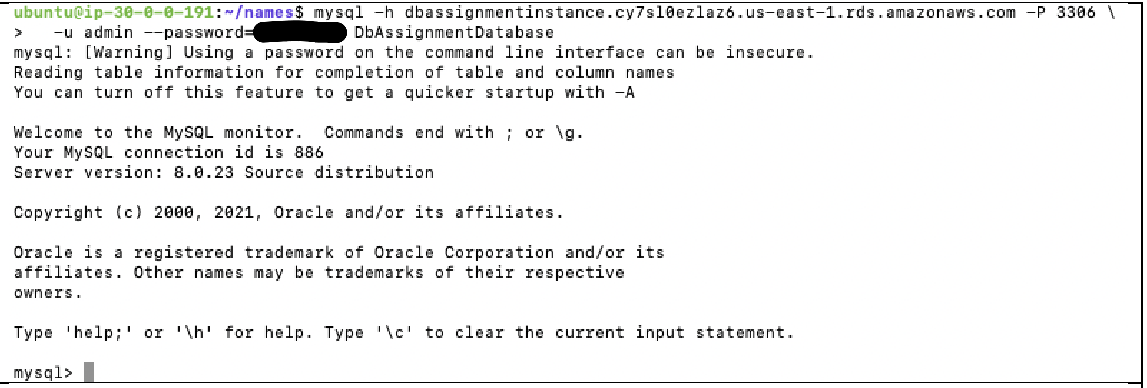


key — ubuntu@ip-30-0-0-191: ~/names — ssh -L localhost:8888:localhost:8888 -i Fall2021KeyPair.pem ubuntu@ec2-18-234-65-205.compute-1.ama
GNU nano 4.8
testdb.py

```
#!/usr/bin/python3

import pymysql
import pymysql.cursors

# Open database connection
db = pymysql.connect(host="dbassignmentinstance.cy7sl0ezlaz6.us-east-1.rds.amazonaws.com",user="admin",password[redacted]db="DbAssignmentDatabase" )

# prepare a cursor object using cursor() method
cursor = db.cursor()

# execute SQL query using execute() method.
cursor.execute("SELECT VERSION()")

# Fetch a single row using fetchone() method.
data = cursor.fetchone()
print ("Database version : %s " % data)

# disconnect from server
db.close()
```

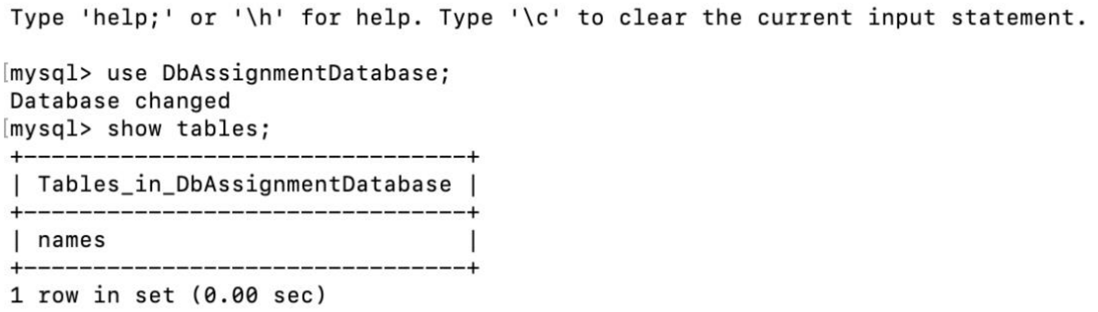
```
[ubuntu@ip-30-0-0-191:~/names$ nano testdb.py
[ubuntu@ip-30-0-0-191:~/names$ chmod +x testdb.py
[ubuntu@ip-30-0-0-191:~/names$ ./testdb.py
Database version : 8.0.23
```

- Your database screen shoot AWS console



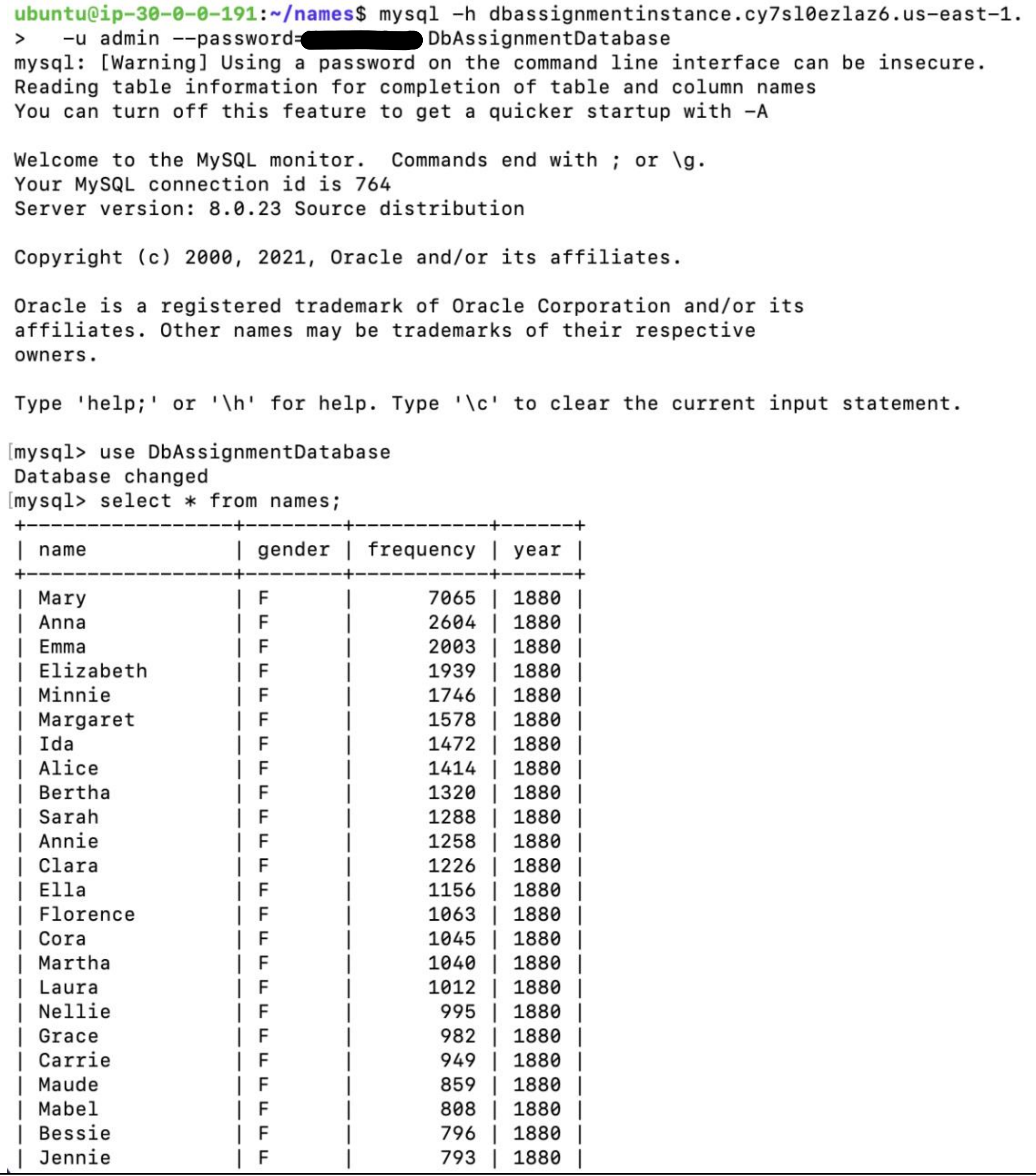
```
ubuntu@ip-30-0-0-191:~/names$ mysql -h dbassignmentinstance.cy7sl0ezlaz6.us-east-1.rds.amazonaws.com -P 3306 \
>   -u admin --password=[redacted] DbAssignmentDatabase
mysql: [Warning] Using a password on the command line interface can be insecure.
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 886
Server version: 8.0.23 Source distribution

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

- A screen shot showing the tables of your database



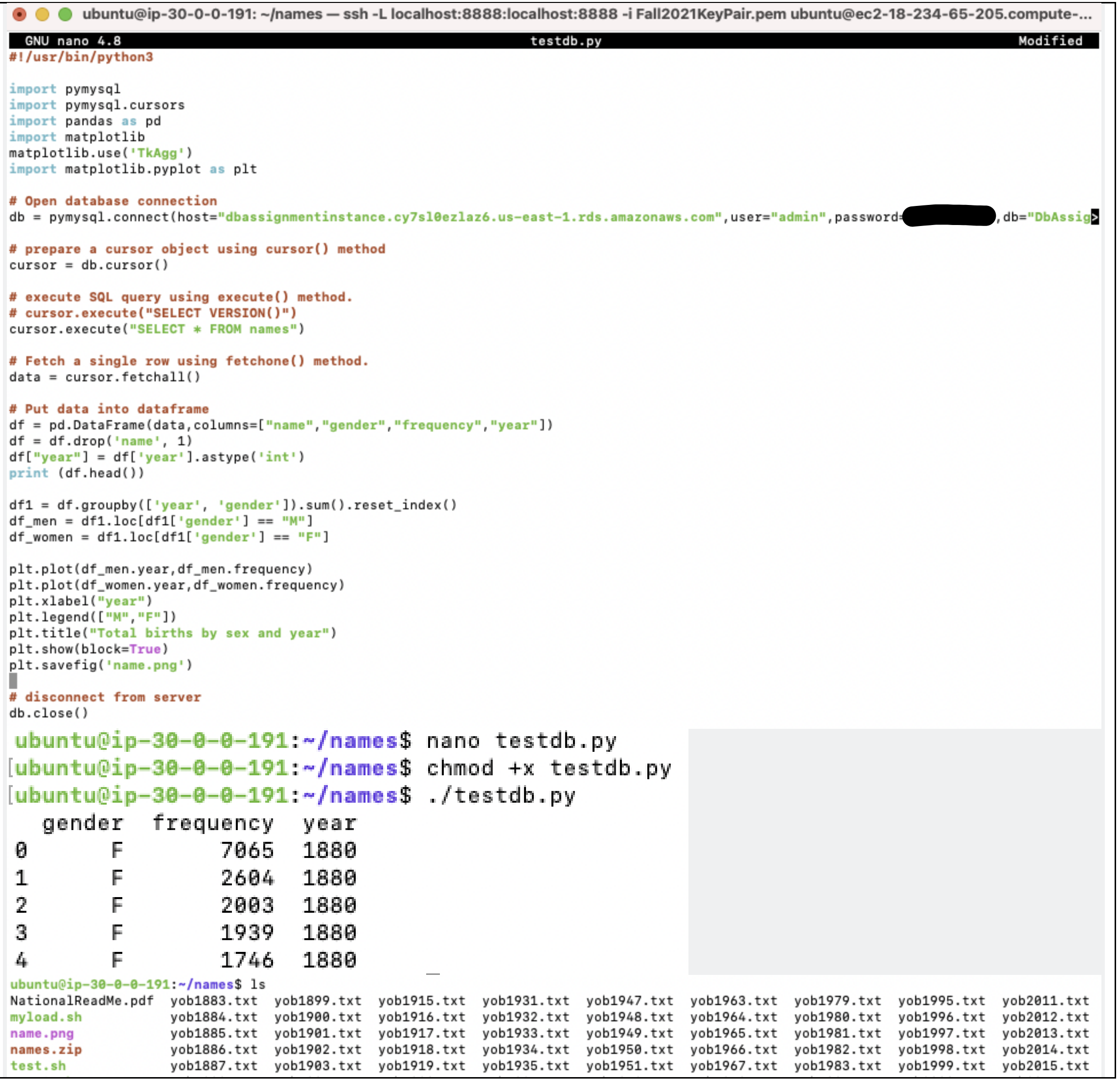
```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use DbAssignmentDatabase;
Database changed
mysql> show tables;
+--------------------------------+
| Tables_in_DbAssignmentDatabase |
+--------------------------------+
| names                          |
+--------------------------------+
1 row in set (0.00 sec)
```

- A screen shot showing the data populated in your database



```
ubuntu@ip-30-0-0-191:~/names$ mysql -h dbassignmentinstance.cy7sl0ezlaz6.us-east-1.
>   -u admin --password=[redacted] DbAssignmentDatabase
mysql: [Warning] Using a password on the command line interface can be insecure.
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 764
Server version: 8.0.23 Source distribution

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use DbAssignmentDatabase
Database changed
mysql> select * from names;
+-----------------+--------+-----------+------+
| name            | gender | frequency | year |
+-----------------+--------+-----------+------+
| Mary            | F      |      7065 | 1880 |
| Anna            | F      |      2604 | 1880 |
| Emma            | F      |      2003 | 1880 |
| Elizabeth       | F      |      1939 | 1880 |
| Minnie          | F      |      1746 | 1880 |
| Margaret        | F      |      1578 | 1880 |
| Ida             | F      |      1472 | 1880 |
| Alice           | F      |      1414 | 1880 |
| Bertha          | F      |      1320 | 1880 |
| Sarah           | F      |      1288 | 1880 |
| Annie           | F      |      1258 | 1880 |
| Clara           | F      |      1226 | 1880 |
| Ella            | F      |      1156 | 1880 |
| Florence        | F      |      1063 | 1880 |
| Cora            | F      |      1045 | 1880 |
| Martha          | F      |      1040 | 1880 |
| Laura           | F      |      1012 | 1880 |
| Nellie          | F      |       995 | 1880 |
| Grace           | F      |       982 | 1880 |
| Carrie          | F      |       949 | 1880 |
| Maude           | F      |       859 | 1880 |
| Mabel           | F      |       808 | 1880 |
| Bessie          | F      |       796 | 1880 |
| Jennie          | F      |       793 | 1880 |
```

- And finally how you generate the required graph.

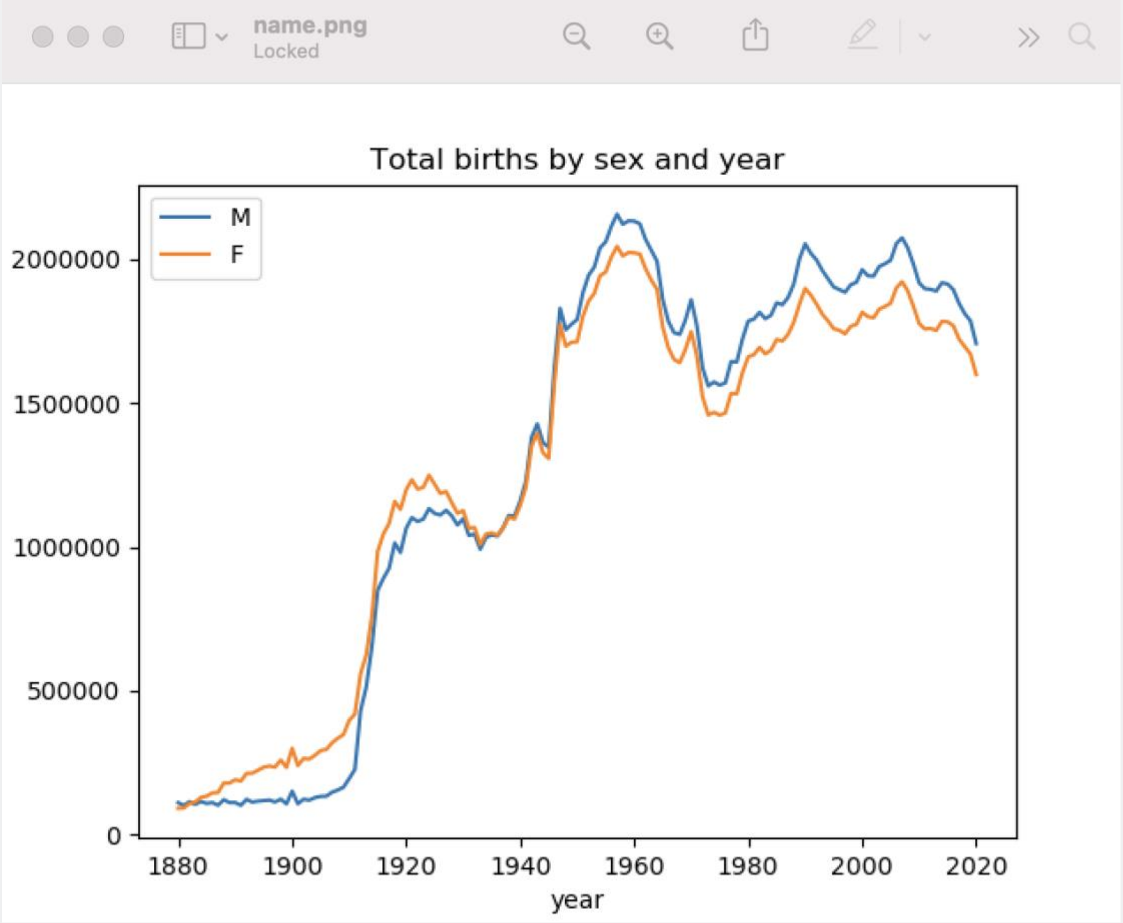


ubuntu@ip-30-0-0-191: ~/names — ssh -L localhost:8888:localhost:8888 -i Fall2021KeyPair.pem ubuntu@ec2-18-234-65-205.compute-...

```
  GNU nano 4.8                              testdb.py                              Modified
#!/usr/bin/python3

import pymysql
import pymysql.cursors
import pandas as pd
import matplotlib
matplotlib.use('TkAgg')
import matplotlib.pyplot as plt

# Open database connection
db = pymysql.connect(host="dbassignmentinstance.cy7sl0ezlaz6.us-east-1.rds.amazonaws.com",user="admin",password=[redacted],db="DbAssig>

# prepare a cursor object using cursor() method
cursor = db.cursor()

# execute SQL query using execute() method.
# cursor.execute("SELECT VERSION()")
cursor.execute("SELECT * FROM names")

# Fetch a single row using fetchone() method.
data = cursor.fetchall()

# Put data into dataframe
df = pd.DataFrame(data,columns=["name","gender","frequency","year"])
df = df.drop('name', 1)
df["year"] = df['year'].astype('int')
print (df.head())

df1 = df.groupby(['year', 'gender']).sum().reset_index()
df_men = df1.loc[df1['gender'] == "M"]
df_women = df1.loc[df1['gender'] == "F"]

plt.plot(df_men.year,df_men.frequency)
plt.plot(df_women.year,df_women.frequency)
plt.xlabel("year")
plt.legend(["M","F"])
plt.title("Total births by sex and year")
plt.show(block=True)
plt.savefig('name.png')


# disconnect from server
db.close()
```

```
ubuntu@ip-30-0-0-191:~/names$ nano testdb.py
[ubuntu@ip-30-0-0-191:~/names$ chmod +x testdb.py
[ubuntu@ip-30-0-0-191:~/names$ ./testdb.py
  gender  frequency  year
0      F       7065  1880
1      F       2604  1880
2      F       2003  1880
3      F       1939  1880
4      F       1746  1880
```

```
ubuntu@ip-30-0-0-191:~/names$ ls
NationalReadMe.pdf  yob1883.txt  yob1899.txt  yob1915.txt  yob1931.txt  yob1947.txt  yob1963.txt  yob1979.txt  yob1995.txt  yob2011.txt
myload.sh           yob1884.txt  yob1900.txt  yob1916.txt  yob1932.txt  yob1948.txt  yob1964.txt  yob1980.txt  yob1996.txt  yob2012.txt
name.png            yob1885.txt  yob1901.txt  yob1917.txt  yob1933.txt  yob1949.txt  yob1965.txt  yob1981.txt  yob1997.txt  yob2013.txt
names.zip           yob1886.txt  yob1902.txt  yob1918.txt  yob1934.txt  yob1950.txt  yob1966.txt  yob1982.txt  yob1998.txt  yob2014.txt
test.sh             yob1887.txt  yob1903.txt  yob1919.txt  yob1935.txt  yob1951.txt  yob1967.txt  yob1983.txt  yob1999.txt  yob2015.txt
```


name.png
Locked
Total births by sex and year
M
F
2000000
1500000
1000000
500000
0
1880
1900
1920
1940
1960
1980
2000
2020
year