Database Migration Service: Heterogeneous Migration between On-premise Oracle to MySQL

Step 1: Create an EC2 instance (Target DB) with MYSQL Server 5.7 installed on it

```
sudo wget http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm
sudo rpm -ivh mysql-community-release-el7-5.noarch.rpm
sudo yum install mysql-server -y
sudo service mysqld start
sudo systemctl enable mysqld
```

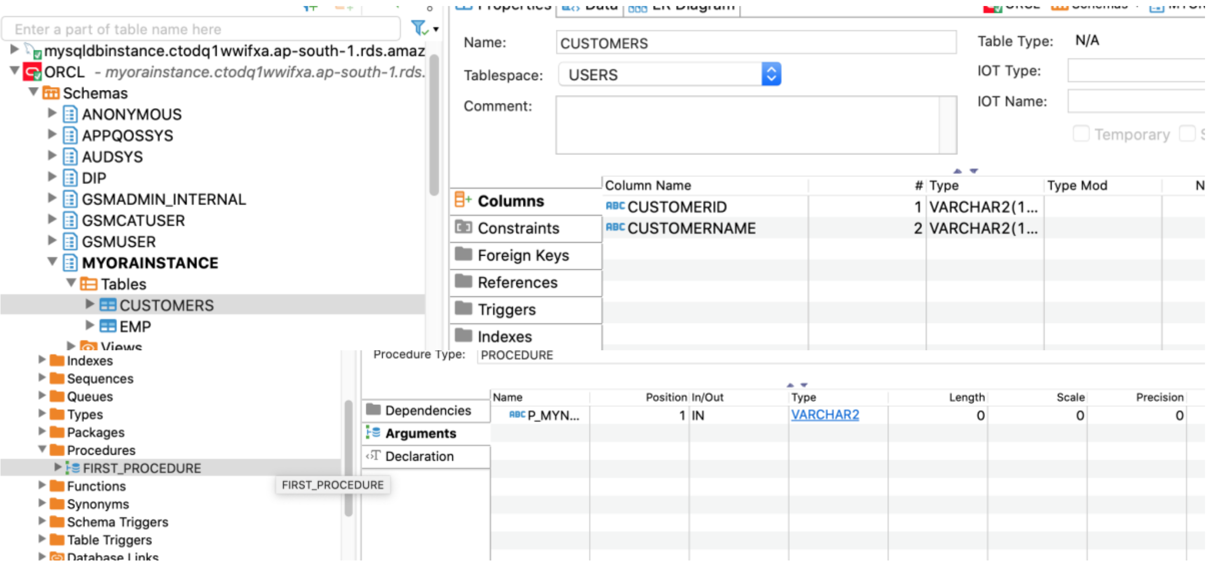
Step 2: Login to mysql and create a database.

```
mysql -uroot
mysql> create database myonpremdb;
mysql> CREATE USER 'onpremuser'@'localhost' IDENTIFIED BY
'onpremuser';

mysql> GRANT ALL PRIVILEGES ON *.* TO 'onpremuser'@'localhost' WITH
GRANT OPTION;

mysql> CREATE USER 'onpremuser'@'%' IDENTIFIED BY 'onpremuser';
mysql> GRANT ALL PRIVILEGES ON *.* TO 'onpremuser'@'%' WITH GRANT
OPTION;
mysql> exit;
```

Login as onprem user

```
$ mysql -u onpremuser -ponpremuser
```

Step 3: Create a new DB in RDS with Oracle 12 as the engine with old interface. Note down the endpoint once created. Make sure the security group created by RDS has “everyone” in the source. In the existing schema, create a few tables with data and also some stored procedures.



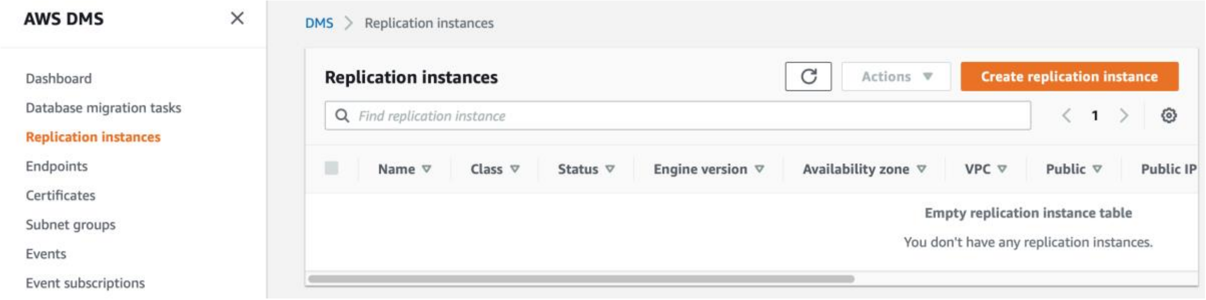
Name: CUSTOMERS
Tablespace: USERS
Comment:
Table Type: N/A
IOT Type:
IOT Name:

Column Name	#	Type	Type Mod
CUSTOMERID	1	VARCHAR2(1...	
CUSTOMERNAME	2	VARCHAR2(1...	

Procedure Type: PROCEDURE

Name	Position	In/Out	Type	Length	Scale	Precision
P_MYN...	1	IN	VARCHAR2	0	0	0

Step 4: Open DMS in AWS console and click on Replication Instances.



Replication instances

Name	Class	Status	Engine version	Availability zone	VPC	Public	Public IP

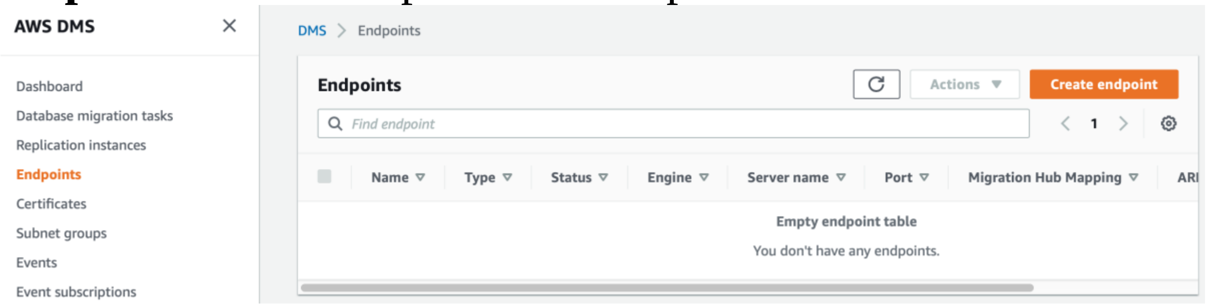
Empty replication instance table

You don't have any replication instances.

Step 5: Click on Create Replication instance and choose the following options
Name : ReplicationInstance
Instance class : T2.micro
VPC: Default VPC
Publicly Accessible: Checked
Multi – AZ: Single AZ
Click Create

Create end point connection for your source and Destination databases.

Step 6: Click on endpoints on left panel



Endpoints

□	Name	Type	Status	Engine	Server name	Port	Migration Hub Mapping	ARI

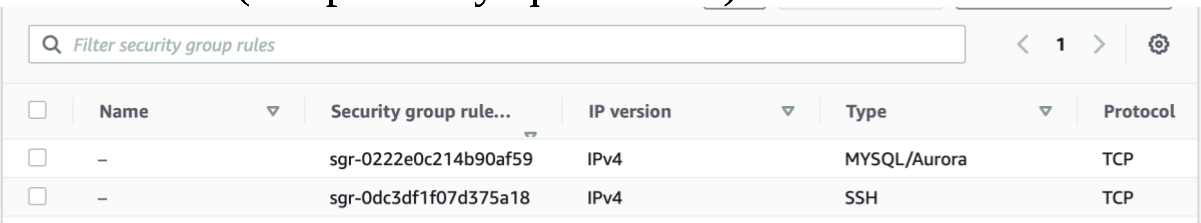
Empty endpoint table

You don't have any endpoints.

Step 7: Click create endpoint. Choose Source endpoint and enter the following details

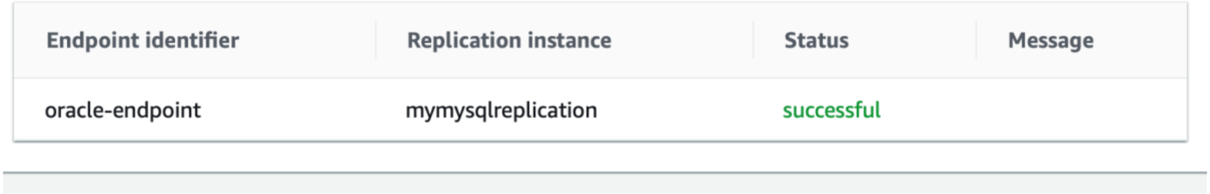
Select RDS db instance: Check and select the Oracle RDS database

Endpoint identifier: Oracle-endpoint
Source Engine: MySQL
Access to endpoint database: Manual
Server name: Public IP of your MySQL instance(on prem)
User name:mysqlinstance
Password: mysqlinstance
Port: 1521
Click Test Endpoint and choose Default VPC. Click Run Test
Note: Make sure you have mysql protocol allowed in your ec2 instance(On-prem mysql instance) inbound rules



□	Name	Security group rule...	IP version	Type	Protocol
□	–	sgr-0222e0c214b90af59	IPv4	MYSQL/Aurora	TCP
□	–	sgr-0dc3df1f07d375a18	IPv4	SSH	TCP

Once the test is successful following screen is displayed



Endpoint identifier	Replication instance	Status	Message
oracle-endpoint	mymysqlreplication	successful	

Click Create Endpoint.

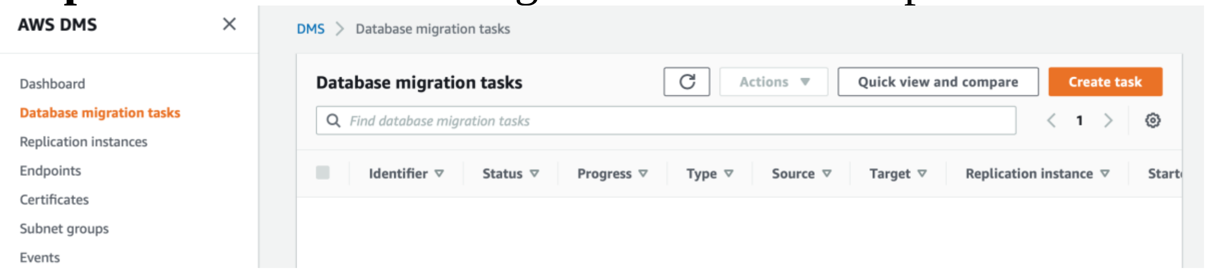
Step 8: Click create endpoint. Choose Target endpoint and enter the following details

Endpoint identifier: Onprem-Mysql-endpoint
Source Engine: MySQL
Access to endpoint database: Manual
Server name: mysql endpoint
User name:onpremuser
Password: onpremuser
Port: 3306
Click Test Endpoint and choose Default VPC. Click Run Test

Wait till the test is successful

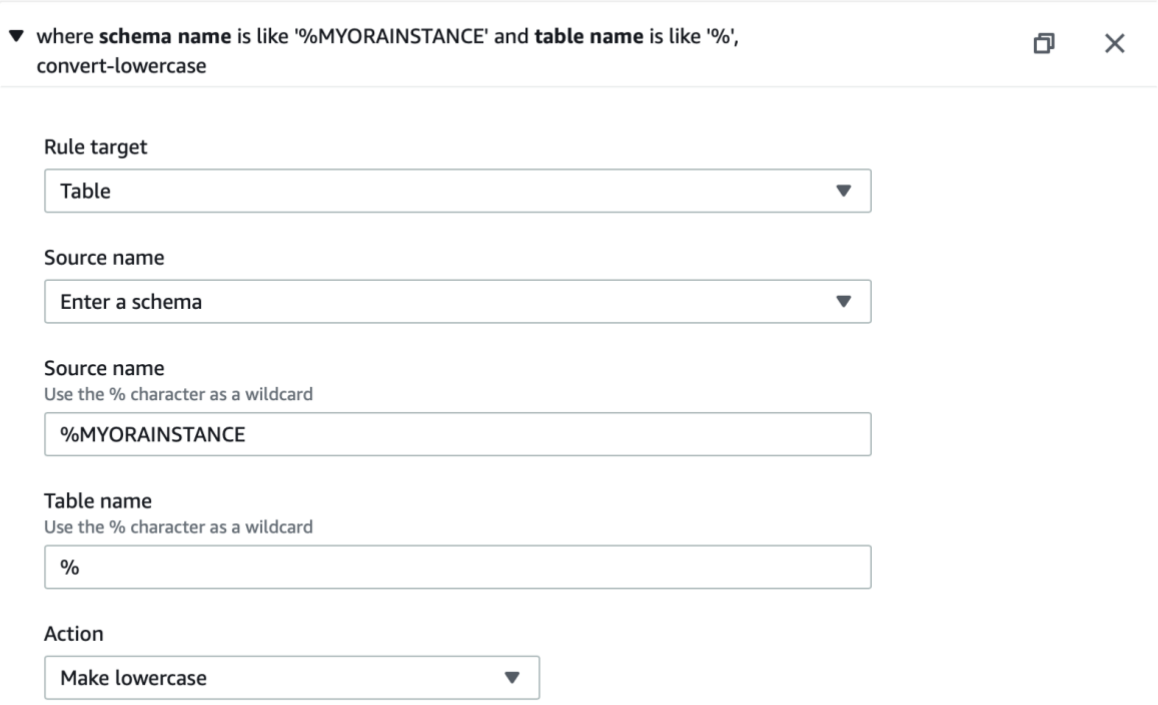
Click Create Endpoint.

Step 9: Click Database Migration Task on left panel



Step 10: Click Create task and enter following details
Task Identifier: MyHeteroDMSTask
Replication instance: Select
Source DB Endpoint: Select your Oracle endpoint
Target DB Endpoint: Select your ec2 mysql endpoint
Table prep mode: Do Nothing
Under table Mappings do the following
Click Add New selection rule
Schema: Enter a schema
Source name: %MYORAINSTANCE
Table name: %

Click Transformation rules and click Add 2 Transformation Rules as below



▼ where **schema name** is like '%MYORAINSTANCE' and **table name** is like '%', convert-lowercase

Rule target

Table ▼

Source name

Enter a schema ▼

Source name
Use the % character as a wildcard

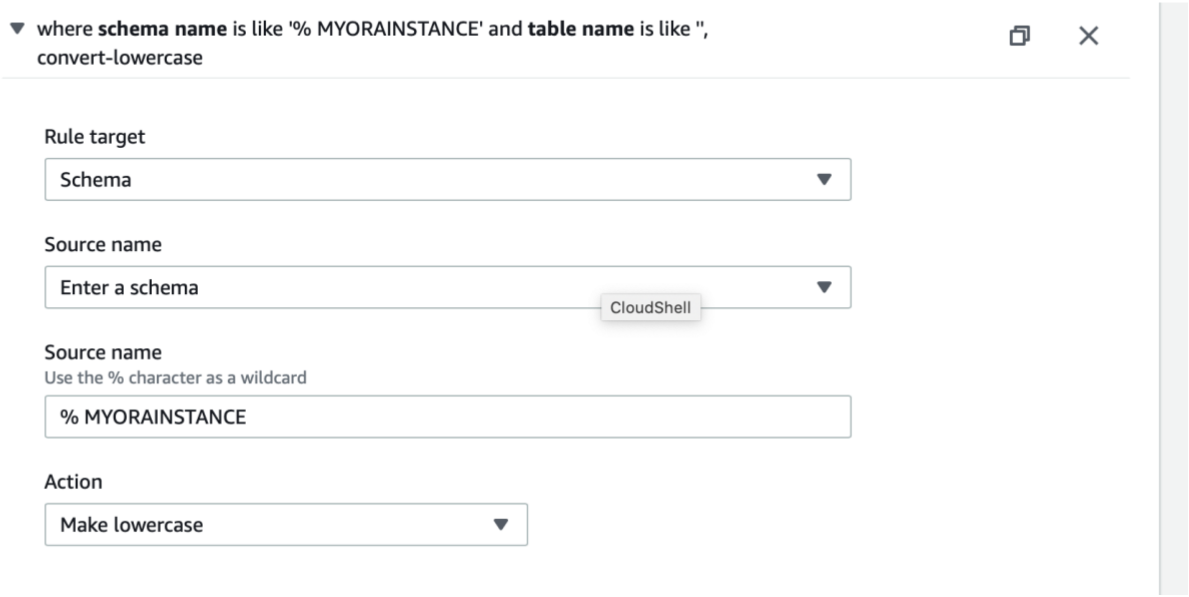
%MYORAINSTANCE

Table name
Use the % character as a wildcard

%

Action

Make lowercase ▼



▼ where **schema name** is like '% MYORAINSTANCE' and **table name** is like '', convert-lowercase

Rule target

Schema ▼

Source name

Enter a schema ▼

Source name
Use the % character as a wildcard

% MYORAINSTANCE

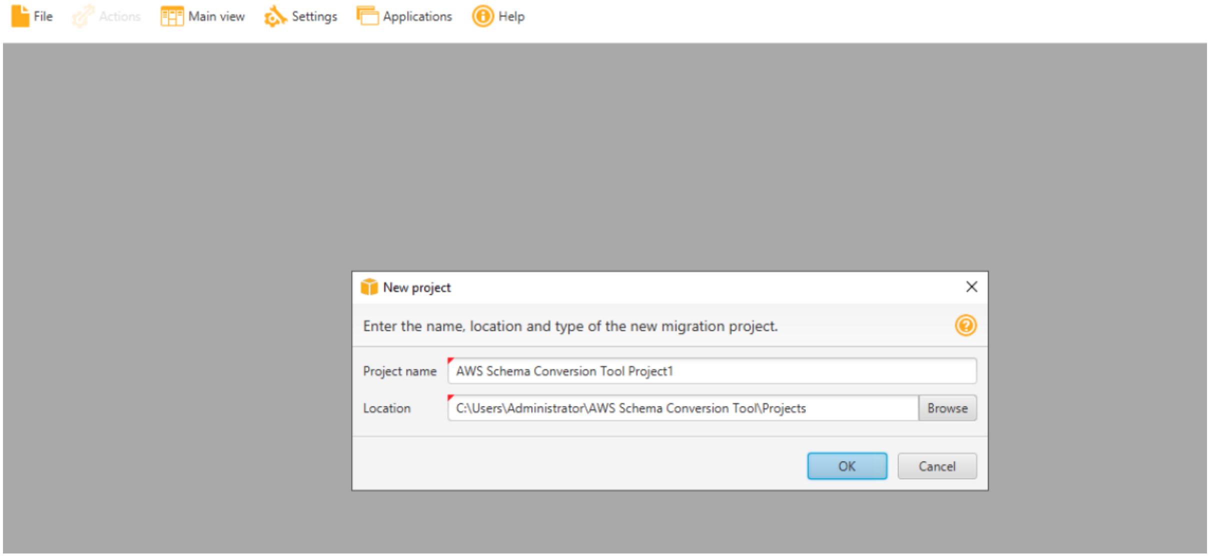
Action

Make lowercase ▼

DO NOT Click Create task.

Step 11: Create a new Windows instance with Type T3.large and install AWS Schema Conversion tool from the URL https://s3.amazonaws.com/publicsctdownload/Windows/aws-schema-conversion-tool-1.0.latest.zip.
Unzip the file and install on your windows instance.

Step 12: Open the SCT and click File and create New Project



New project

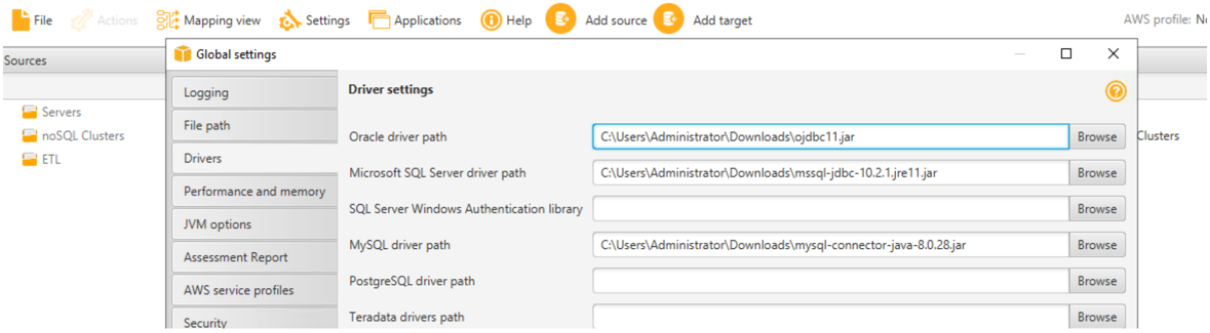
Enter the name, location and type of the new migration project.

Project name: AWS Schema Conversion Tool Project1

Location: C:\Users\Administrator\AWS Schema Conversion Tool\Projects

Step 13: Click on Settings and click Global settings. Browse the path of oracle driver and mysql driver files. (The jars are provided in the materials)

The jars can also be downloaded from the following URL https://docs.aws.amazon.com/SchemaConversionTool/latest/userguide/CHAP_Installing.html#CHAP_Installing.JDBCDrivers



Driver settings

Oracle driver path: C:\Users\Administrator\Downloads\ojdbc11.jar

Microsoft SQL Server driver path: C:\Users\Administrator\Downloads\mssql-jdbc-10.2.1.jre11.jar

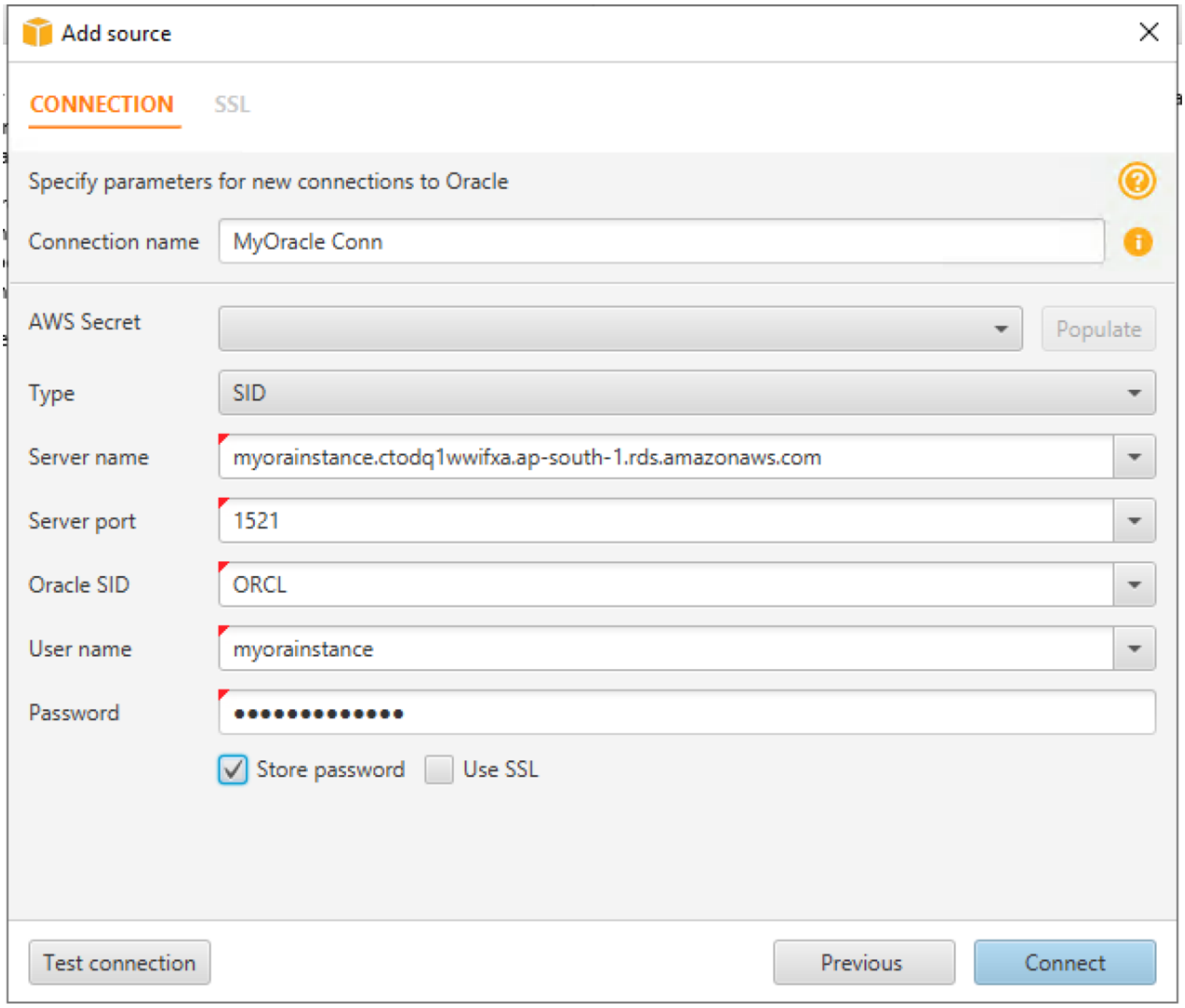
SQL Server Windows Authentication library:

MySQL driver path: C:\Users\Administrator\Downloads\mysql-connector-java-8.0.28.jar

PostgreSQL driver path:

Teradata drivers path:

Step 14: Click Add Source and select Oracle. Click Next. Enter your Oracle DB instance details and Test connection. Click Accept Risk and Test.



Specify parameters for new connections to Oracle

Connection name: MyOracle Conn

AWS Secret:

Type: SID

Server name: myorainstance.ctodq1wwifxa.ap-south-1.rds.amazonaws.com

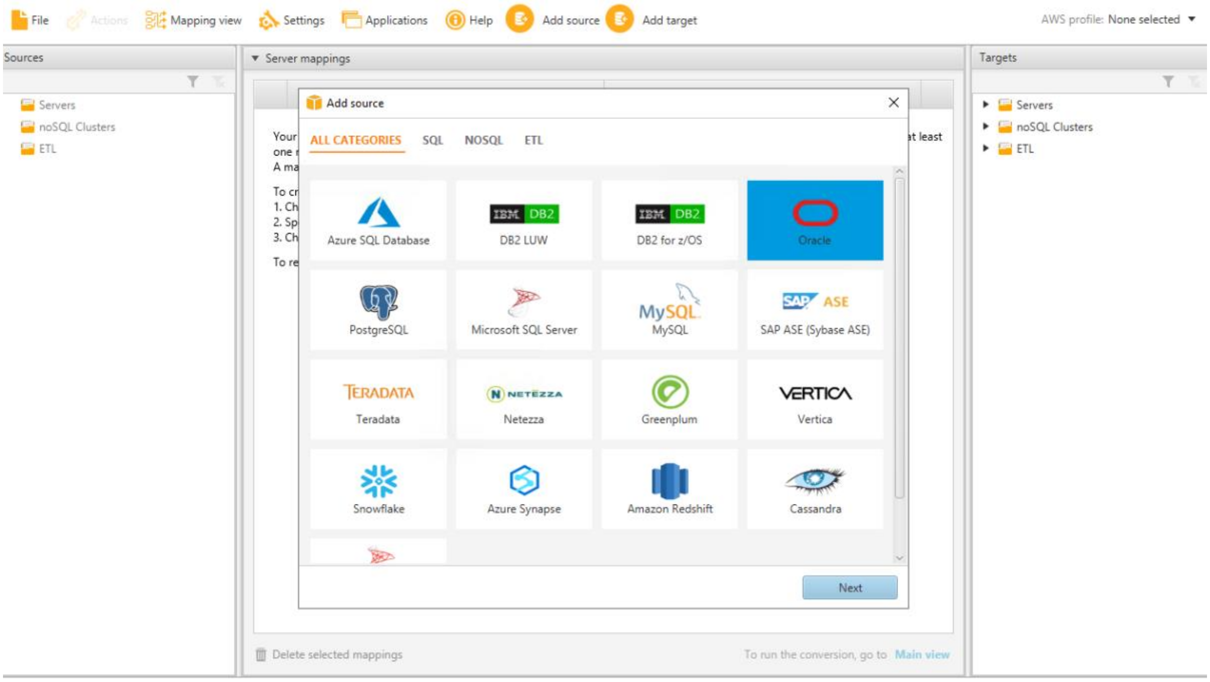
Server port: 1521

Oracle SID: ORCL

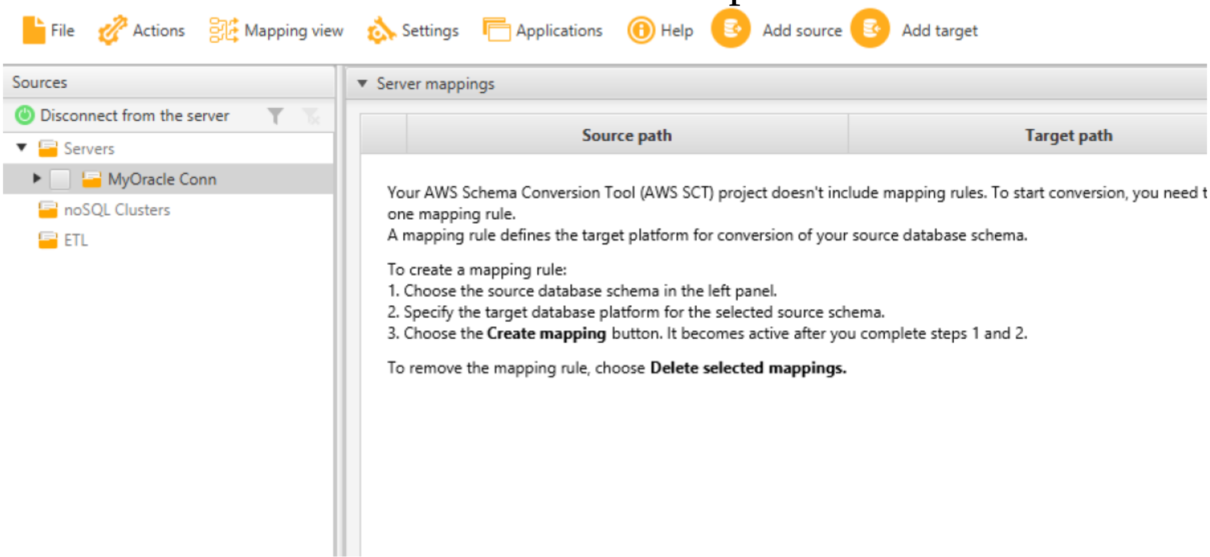
User name: myorainstance

Password: ●●●●●●●●●●●●●

☑ Store password ☐ Use SSL



Step 15: Once Connection is successful click Connect. Your Oracle DB must be shown on the left panel

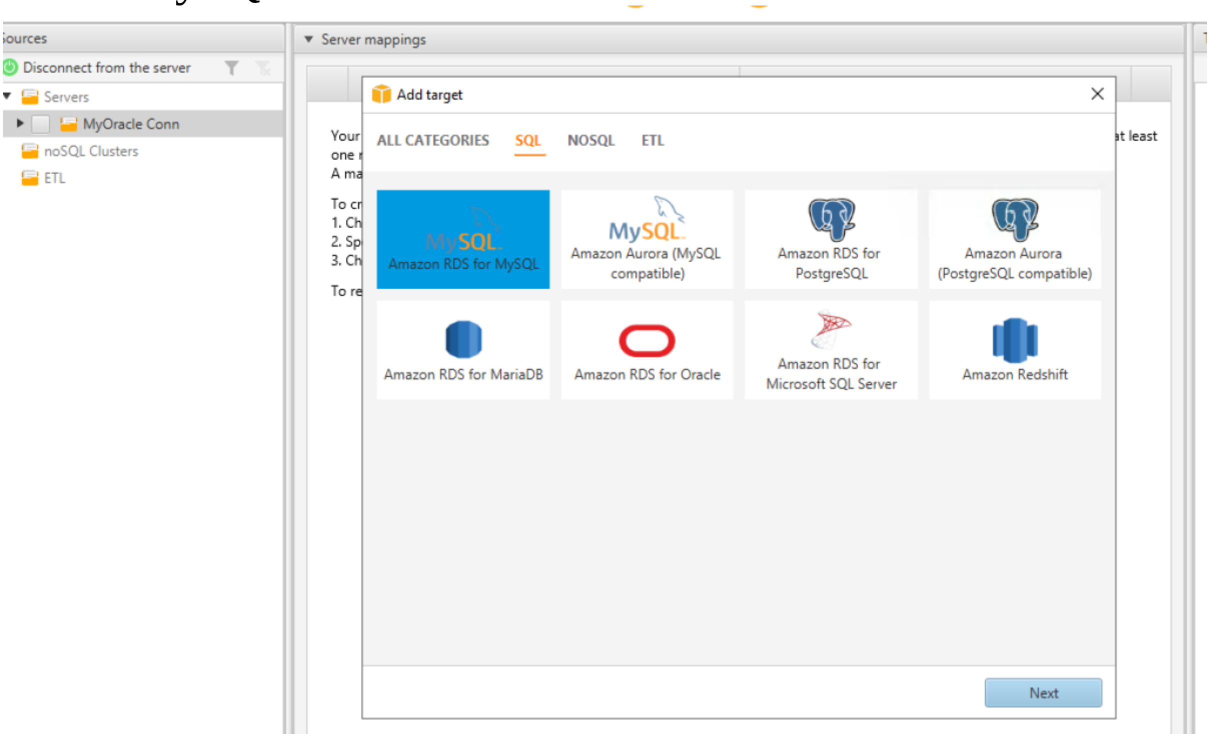


Your AWS Schema Conversion Tool (AWS SCT) project doesn't include mapping rules. To start conversion, you need t
one mapping rule.
A mapping rule defines the target platform for conversion of your source database schema.

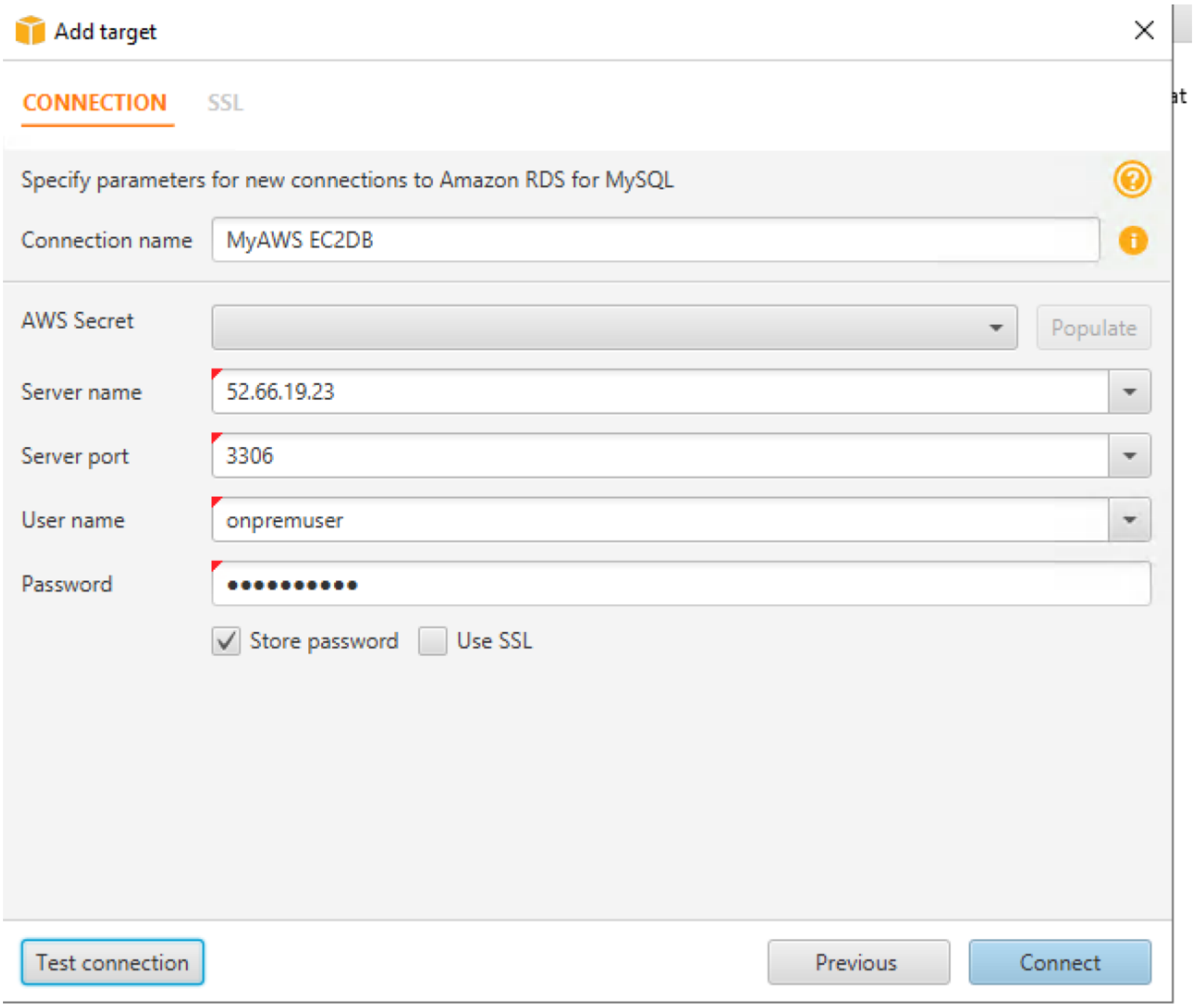
To create a mapping rule:
1. Choose the source database schema in the left panel.
2. Specify the target database platform for the selected source schema.
3. Choose the **Create mapping** button. It becomes active after you complete steps 1 and 2.

To remove the mapping rule, choose **Delete selected mappings.**

Step 16: Click On the right panel and click Add Target DB. Select MySQL RDS DB. Click Next



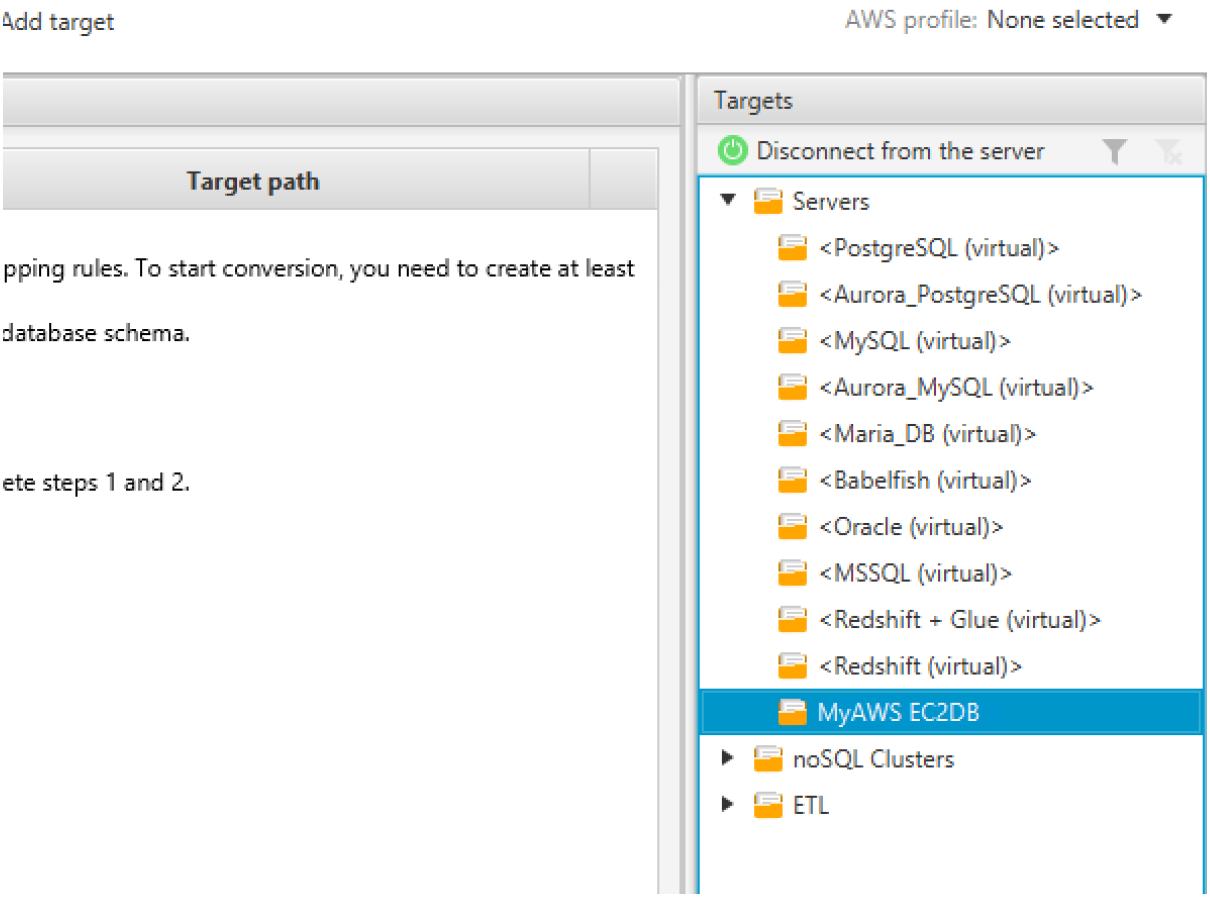
Step 17: Add details for your Mysql instance and connect. You should see your my sql on the right panel



Specify parameters for new connections to Amazon RDS for MySQL

Connection name	MyAWS EC2DB
AWS Secret	
Server name	52.66.19.23
Server port	3306
User name	onpremuser
Password	●●●●●●●●●●●

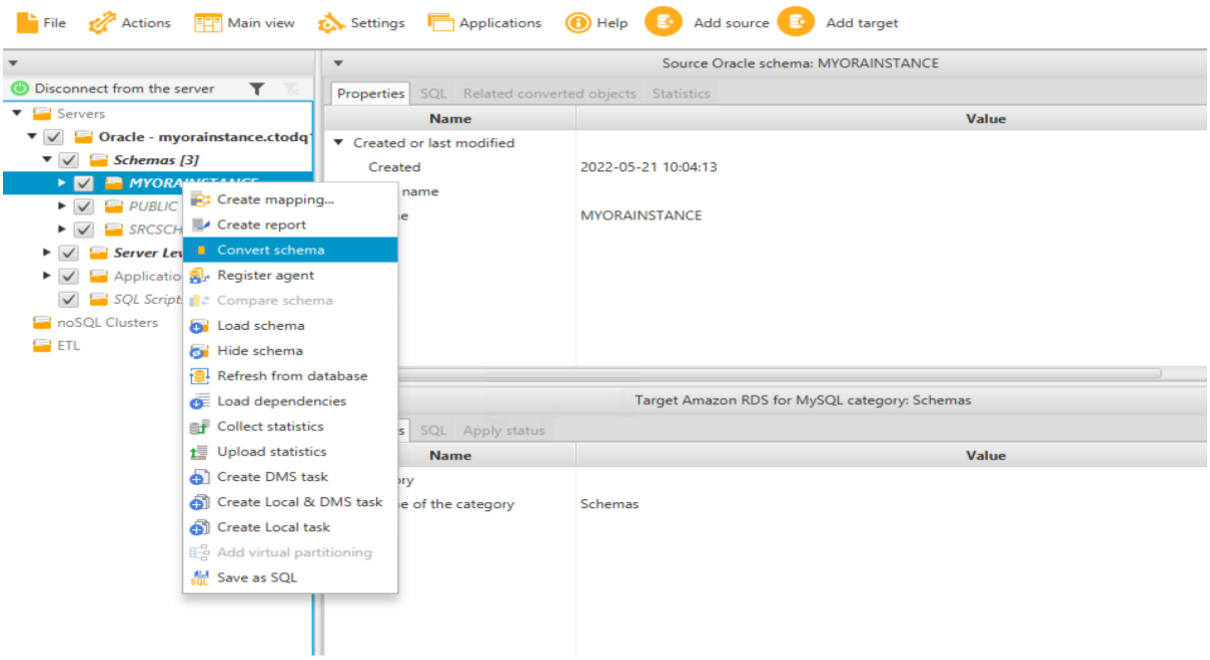
☑ Store password □ Use SSL



pping rules. To start conversion, you need to create at least database schema.

ete steps 1 and 2.

Step 18: Click Project View at the top and select Mapping view. Click Create Mapping.
Step 19: Click Project view and Select Main view.
Step 20: Select Your schema and click Convert Schema



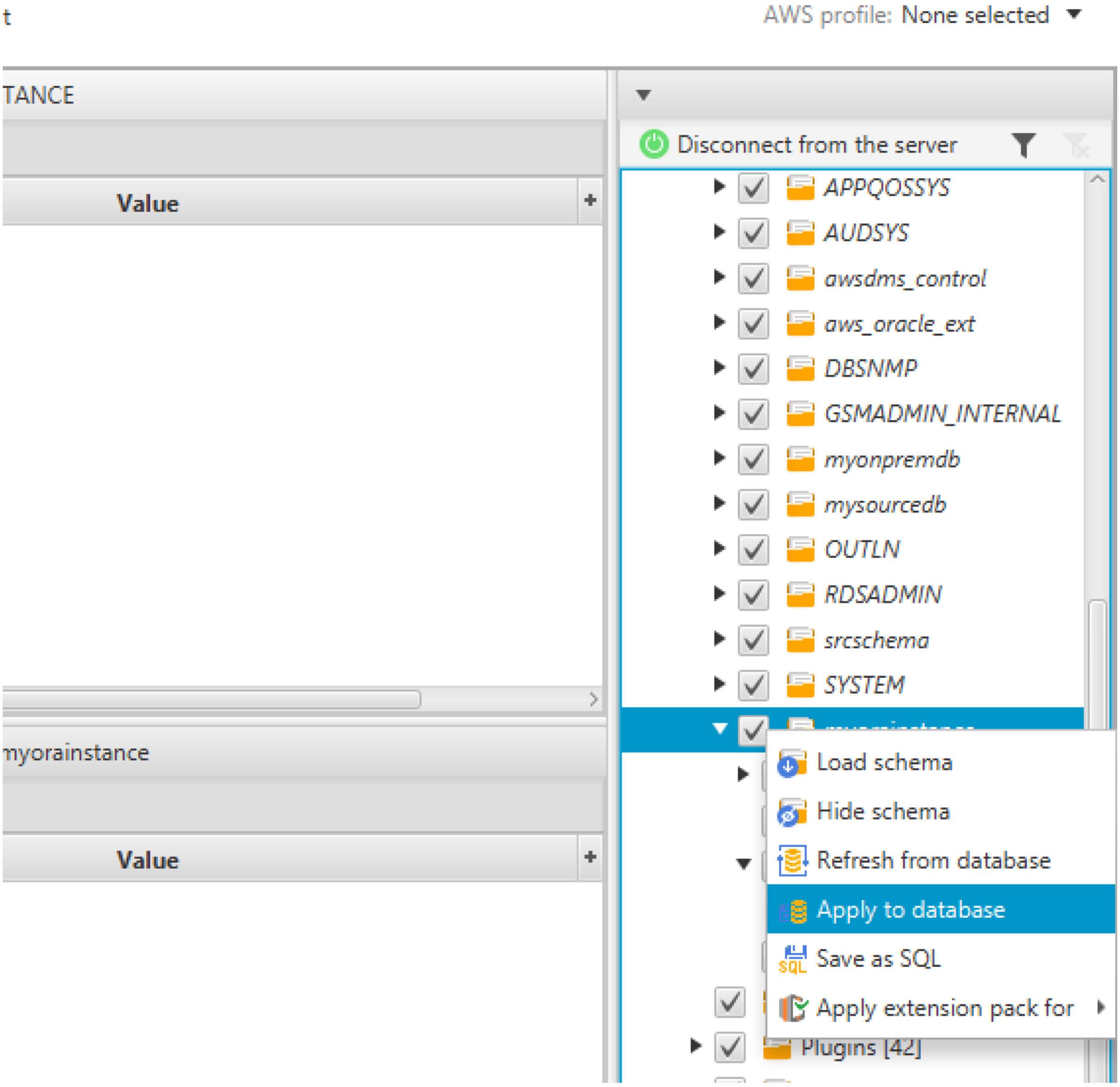
Source Oracle schema: MYORAINSTANCE

Name	Value
Created or last modified	
Created	2022-05-21 10:04:13
name	
e	MYORAINSTANCE

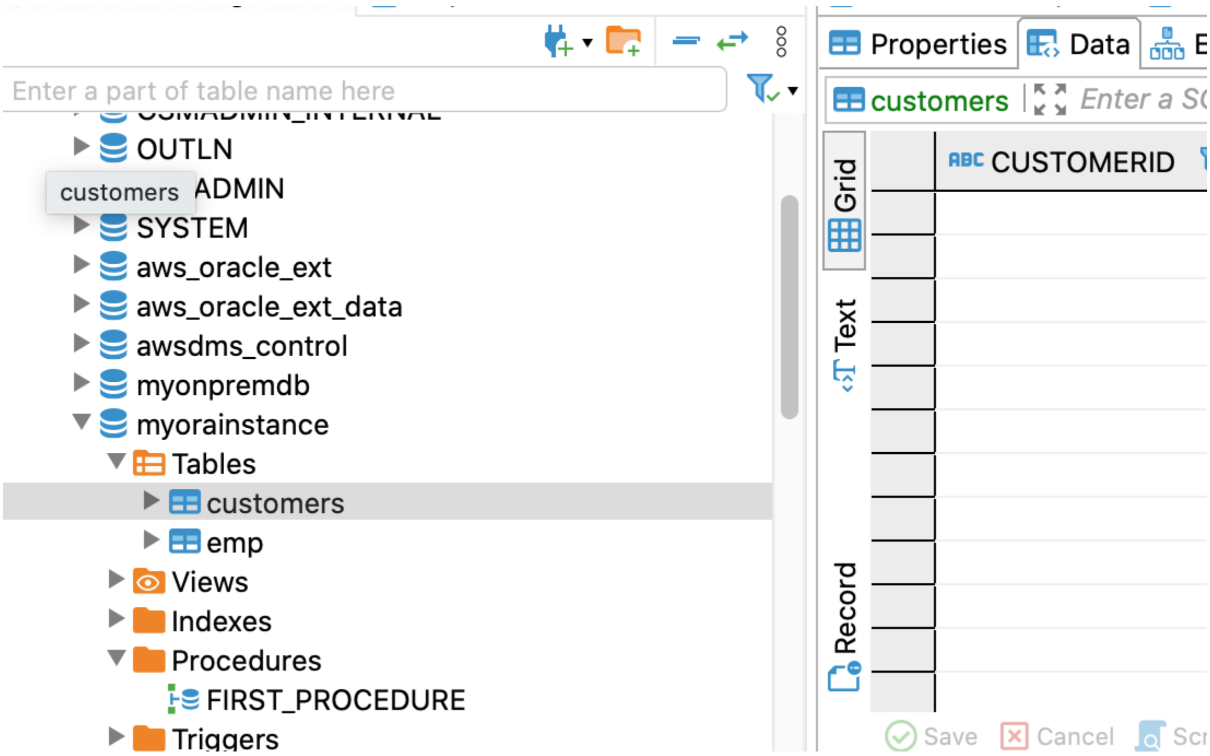
Target Amazon RDS for MySQL category: Schemas

Name	Value
ry	
e of the category	Schemas

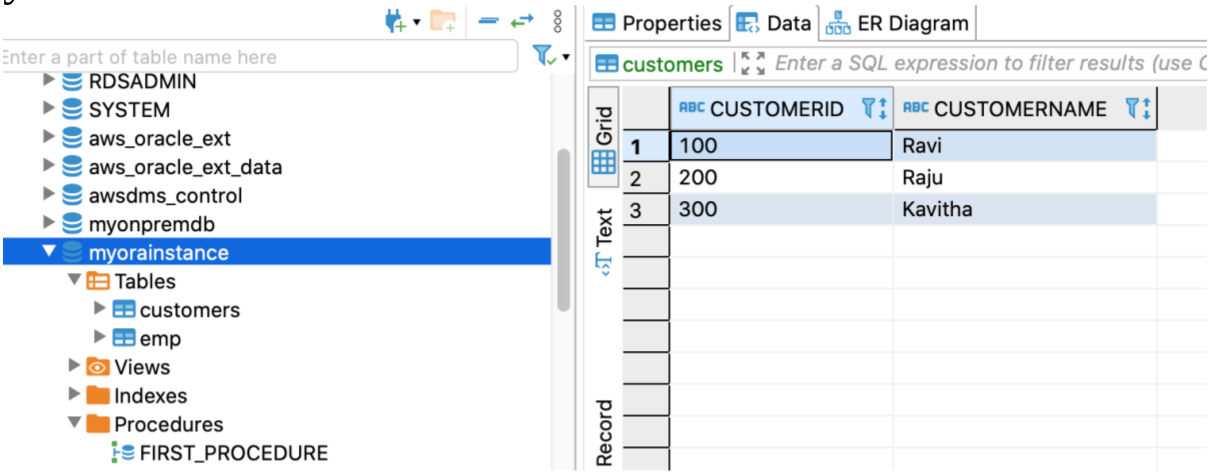
Step 21: After the conversion is complete, observe the right panel to find your database in the target DB.Right click on the schema and click Apply to Database.



Step 22: Use the DB tool of your choice and connect to your target DB(MySQL) and check if the DB tables are converted



Step 23: Go back to the Database Migration Task and click create task. You should be able to see the data migrated to your new database.



	CUSTOMERID	CUSTOMERNAME
1	100	Ravi
2	200	Raju
3	300	Kavitha