

I used MYSQL workbench to connect to AWS RDS.

1. Proof of Active Connection:

The screenshot displays the MySQL Workbench interface with the 'mysql_aws' connection selected. The left sidebar shows the 'MANAGEMENT' tab with options like Server Status, Client Connections, and Users and Privileges. The 'SCHEMAS' tab is active, showing a list of databases including 'devdb' and 'innodb'. The main query editor shows a query: `select * from emp;`. The 'Result Grid' is visible, showing a table with columns: emp_id, emp_name, dept_id, sal, date_of_joining. The 'Query Output' tab at the bottom shows the execution log with the following entries:

Action	Time	Action	Response	Duration / Fetch Time
5	13:51:37	select * from all_tables LIMIT 0, 1000	Error Code: 1146. Table 'devdb.all_tables' doesn't exist	0.044 sec
6	13:56:39	DROP DATABASE 'innodb'	Error Code: 1010. Error dropping database (can't rmdi...	0.045 sec
7	13:56:58	DROP DATABASE 'innodb'	Error Code: 1010. Error dropping database (can't rmdi...	0.050 sec
8	13:58:19	select * from emp LIMIT 0, 1000	0 row(s) returned	0.044 sec / 0.000013...
9	14:00:59	select * from emp LIMIT 0, 1000	0 row(s) returned	0.051 sec / 0.000013...

The 'Query Completed' status is shown at the bottom left.

2. Proof of AWS RDS instance:

The screenshot displays the AWS Management Console interface for an Amazon RDS instance. The browser address bar shows the URL: <https://us-west-1.console.aws.amazon.com/rds/home?region=us-west-1#dbinstances:>. The console header includes navigation tabs for 'Launch DB Instance', 'Show Monitoring', and 'Instance Actions'. A sidebar on the left lists various RDS-related services like 'Instances', 'Reserved Purchases', 'Snapshots', etc. The main content area shows a table with one instance, 'dev', of type 'MySQL' and status 'available'. Below the table, the 'Endpoint' is listed as 'dev.cv5ofpddpvr.us-west-1.rds.amazonaws.com:3306 (authorized)'. The instance details are organized into several sections: 'Configuration Details' (Engine: MySQL 5.6.23, License: General Public License, Created Time: January 7, 2016 at 12:15:25 PM UTC-8, DB Name: devdb, Username: admin, Option Group: default:mysql-5-6 (in-sync), Parameter Group: default:mysql5.6 (in-sync), Copy Tags To Snapshots: No), 'Security and Network' (Availability Zone: us-west-1a, VPC: vpc-773b8612, Subnet Group: default (Complete), Subnets: subnet-cff396aa, subnet-ac38e2f5, Security Groups: rds-launch-wizard (sg-98f9c5fd) (active), Publicly Accessible: Yes, Endpoint: dev.cv5ofpddpvr.us-west-1.rds.amazonaws.com, Port: 3306, Certificate Authority: rds-ca-2015 (Mar 5, 2020)), 'Instance and IOPS' (Instance Class: db.t2.micro, Storage Type: General Purpose (SSD), IOPS: disabled, Storage: 5 GB), 'Encryption Details' (Encryption Enabled: No), 'Availability and Durability' (DB Instance Status: available, Multi AZ: No, Automated Backups: Enabled (7 Days), Latest Restore Time: January 7, 2016 at 2:00:00 PM UTC-), 'Maintenance Details' (Auto Minor Version Updates: Enabled, Pending Maintenance: None, Backup window: 00:35-01:05), and a warning about the certificate expiration date of March 5, 2020. The footer of the console shows the copyright notice: '© 2008 - 2016, Amazon Web Services, Inc. or its affiliates. All rights reserved.' and links to 'Privacy Policy' and 'Terms of Use'.

RDS Dashboard

Launch DB Instance Show Monitoring Instance Actions

Filter: All Instances Search DB Instances... Viewing 1 of 1 DB Instances

Engine	DB Instance	Status	CPU	Current Activity	Maintenance	Class	VPC	Multi-AZ	Replication Role
MySQL	dev	available	0.83%	4 Connections	None	db.t2.micro	vpc-773b8612	No	

Endpoint: dev.cv5ofpddpvr.us-west-1.rds.amazonaws.com:3306 (authorized)

Configuration Details

- Engine: MySQL 5.6.23
- License Model: General Public License
- Created Time: January 7, 2016 at 12:15:25 PM UTC-8
- DB Name: devdb
- Username: admin
- Option Group: default:mysql-5-6 (in-sync)
- Parameter Group: default:mysql5.6 (in-sync)
- Copy Tags To Snapshots: No

Security and Network

- Availability Zone: us-west-1a
- VPC: vpc-773b8612
- Subnet Group: default (Complete)
- Subnets: subnet-cff396aa, subnet-ac38e2f5
- Security Groups: rds-launch-wizard (sg-98f9c5fd) (active)
- Publicly Accessible: Yes
- Endpoint: dev.cv5ofpddpvr.us-west-1.rds.amazonaws.com
- Port: 3306
- Certificate Authority: rds-ca-2015 (Mar 5, 2020)

Instance and IOPS

- Instance Class: db.t2.micro
- Storage Type: General Purpose (SSD)
- IOPS: disabled
- Storage: 5 GB

Encryption Details

- Encryption Enabled: No

Availability and Durability

- DB Instance Status: available
- Multi AZ: No
- Automated Backups: Enabled (7 Days)
- Latest Restore Time: January 7, 2016 at 2:00:00 PM UTC-

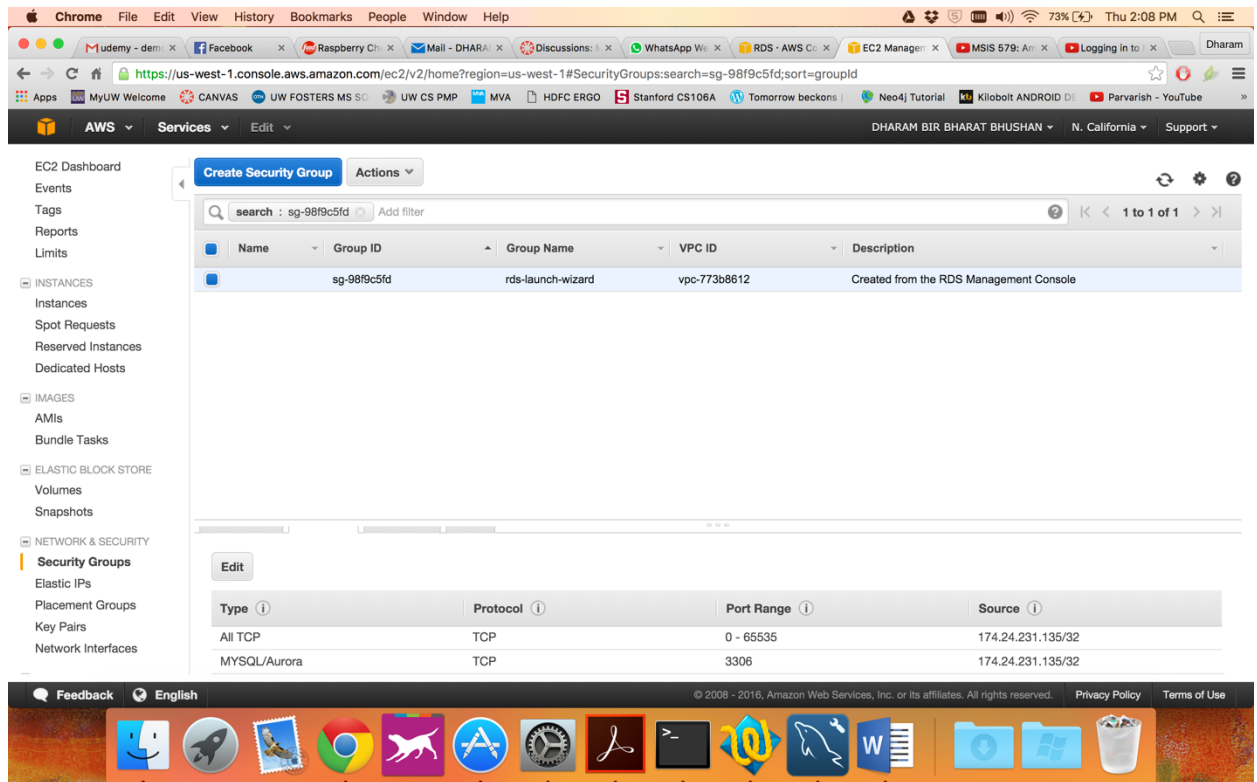
Maintenance Details

- Auto Minor Version Updates: Enabled
- Pending Maintenance: None
- Backup window: 00:35-01:05

Date the certificate will expire. If your application uses SSL to connect to your RDS instance you must update your certificate bundle before this date or your applications may fail to connect to your RDS instances after that time.

Feedback English © 2008 - 2016, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

3. Proof of Security Setting:



4. EXTRA: Using python program to fetch data from AWS RDS

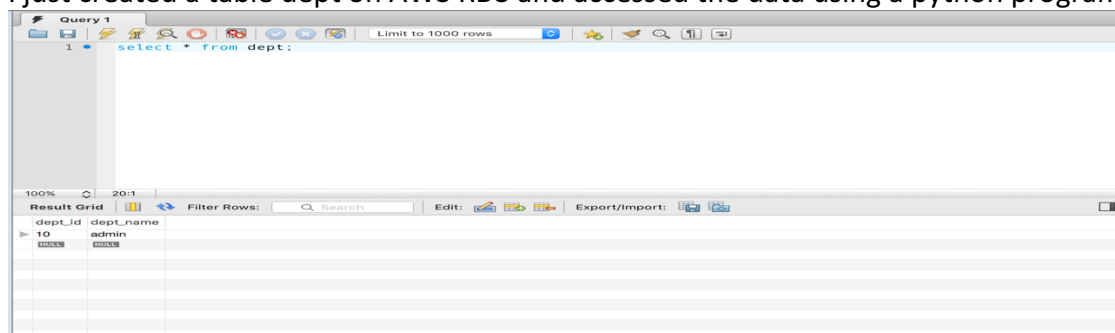
Installed PyMySQL using the below command.

```
pip install PyMySQL
```

Created a login file to hide the credentials which contains the following separated by a space.

EC2_endpoint user_name password port_number db_name

I just created a table dept on AWS RDS and accessed the data using a python program.



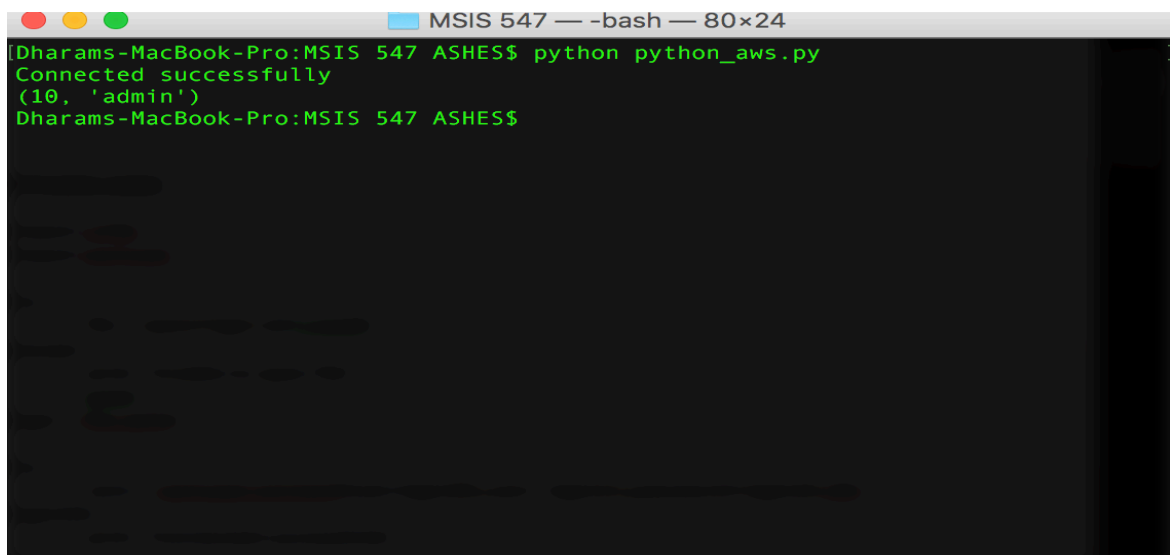
PYTHON CODE

```
import urllib
import pymysql

try:
    file = open('login.txt').read()
except:
    print "Unable to open file"
    exit()
login = file.split()

try:
    conn = pymysql.connect(host=login[0],user= login[1],password=login[2],db=
login[4])
    print "Connected successfully"
except:
    print "Unable to connect"
cur = conn.cursor()
sql = "select * from dept"
cur.execute(sql)
result = cur.fetchone()
print (result)
conn.close()
```

EXECUTE THE CODE (HOLLAAAA !!!)

A screenshot of a terminal window on a Mac. The window title is "MSIS 547 — -bash — 80x24". The prompt is "Dharams-MacBook-Pro:MSIS 547 ASHES\$". The user has entered "python python_aws.py". The output shows "Connected successfully" and "(10, 'admin')". The prompt is now "Dharams-MacBook-Pro:MSIS 547 ASHES\$".

```
MSIS 547 — -bash — 80x24
Dharams-MacBook-Pro:MSIS 547 ASHES$ python python_aws.py
Connected successfully
(10, 'admin')
Dharams-MacBook-Pro:MSIS 547 ASHES$
```