

# MSDS 7346

## Cloud Computing

### Mini Project 1 – AWS

Name: Mooyoung Lee

**Question 1 :** The objective of this lab is to gain familiarity with AWS (public cloud provider). In this course, we will primarily use AWS, but at times will work with other public providers to be able to compare and contrast.

If you do not already have AWS subscription, please sign up as a student. AWS provides you access to certain resources for free. Please be advised, NOT all of the services are free and it is your responsibility to ensure that you launch free resources and terminate them as soon as you are finished. Each instance in AWS states "free tier enabled." If you choose any other one, it could cost you money.

Once you are signed up to AWS, you will configure and launch an instance in EC2 instance. You can choose operating system of your choice. I usually work with Linux, but that does not mean you need to use that. Once an instance is launched, you need to connect to that instance from your local machine (laptop) using the secure shell.

At this time you should have an instance up and running in AWS, and you should be able to login from your laptop using SSH (putty), etc.

The next step is to download and install the MySQL Community Server database program on the EC2 instance. MySQL Community Server is a free download from <https://www.mysql.com>. Download and install MySQL Workbench on your local machine. Most of you should already have this from your previous course. MySQL Workbench is a visualization application for accessing MySQL databases.

Once you have installed MySQL, be sure to set the password for your user account on the MySQL database. And, be sure to give your account the privileges needed to create and modify databases. The MySQL reference manual, available from <https://www.mysql.com>, provides in-depth instructions on how to install and configure your MySQL software.

Once you have installed and configured MySQL, select the MySQL database by

executing the “USE MySQL” command. Then, run the query “SELECT User, Host FROM mysql.user;” from the command line.

Capture the resulting output as a screen capture or grab and turn in the resulting pdf showing both the query and the results.

This configuration is similar to what we did in the database class except that you are running MySQL on AWS instance. The next step of this lab is to create an AWS RDS instance of MySQL and connect using the MySQL workbench on your local machine.

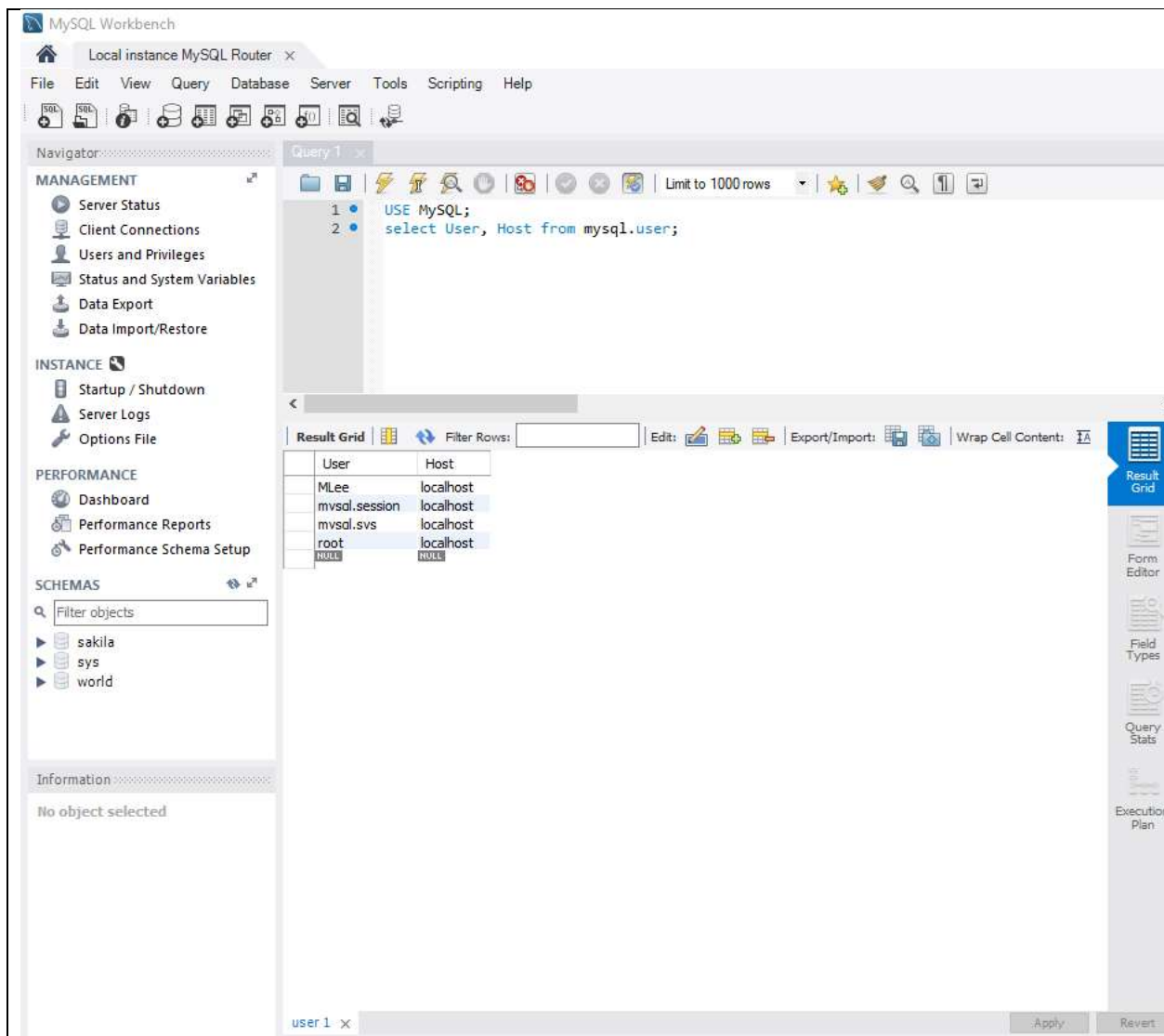
- 1) Create an instance on EC2
- 2) Download and install MySQL on EC2
- 3) Download (if you don't already have) MySQL Workbench on your local machine
- 4) Connect MySQL Workbench to EC2 instance (you will need to create keys)
- 5) Create MySQL instance on RDS
- 6) Connect your local MySQL Workbench to MySQL on RDS

Submission: Submit different screen shots to show completion of each steps

NOTE: As stated earlier, please make sure to only use free resources on AWS and **NEVER FORGET to terminate your instances**. Everything that you get from your subscription is NOT FREE.

## Answers:

### 1A. Install MySQL Workbench on local machine



The screenshot displays the MySQL Workbench interface. The left sidebar contains the 'Navigator' pane with sections for MANAGEMENT, INSTANCE, PERFORMANCE, and SCHEMAS. The 'Query 1' editor shows the following SQL code:

```
1 • USE MySQL;  
2 • select User, Host from mysql.user;
```

The 'Result Grid' pane shows the execution results of the query:

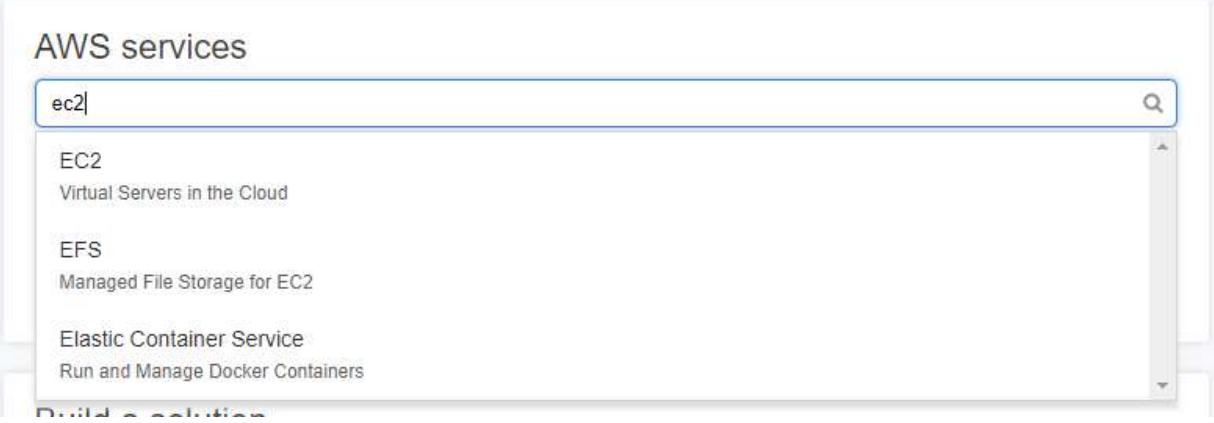
User	Host
MLee	localhost
mysql.session	localhost
mysql.svs	localhost
root	localhost
NULL	NULL

The bottom status bar indicates 'user 1' is selected, with 'Apply' and 'Revert' buttons.

## 1B. Create AWS RDS instance

### 1) Create an instance on EC2

- Search EC2 window



The screenshot shows the AWS search bar with 'ec2' entered. The search results list several services: EC2 (Virtual Servers in the Cloud), EFS (Managed File Storage for EC2), Elastic Container Service (Run and Manage Docker Containers), and Build (Build solutions).

- Click 'Launch Instance'

### Resources

You are using the following Amazon EC2 resources in the US East (Ohio) region:

0 Running Instances	0 Elastic IPs
0 Dedicated Hosts	0 Snapshots
0 Volumes	0 Load Balancers
0 Key Pairs	1 Security Groups
0 Placement Groups	

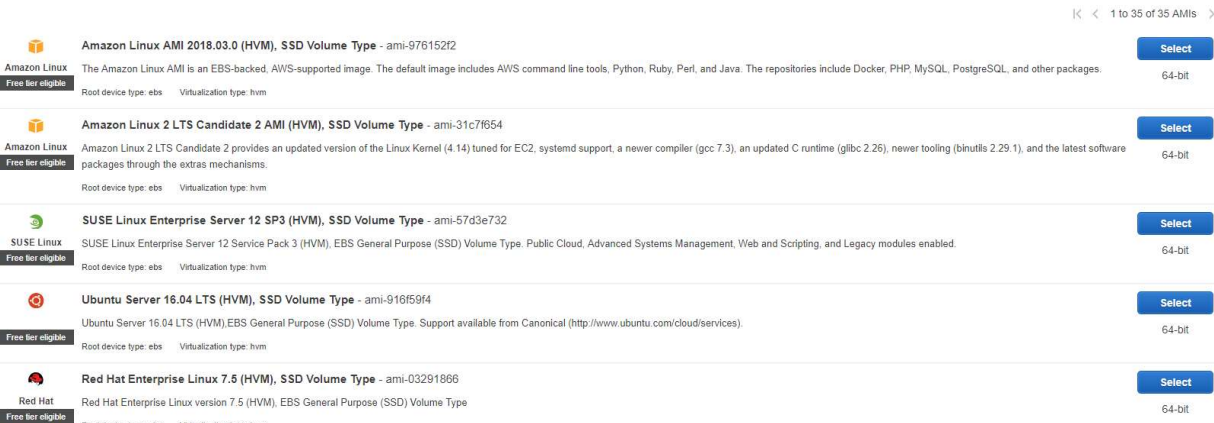
Learn more about the latest in AWS Compute from AWS re:Invent 2017 by viewing the EC2 Videos.

### Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

[Launch Instance](#)

- Select instance w/ right operating system



The screenshot shows the 'Create Instance' page with a list of AMIs. The first AMI is 'Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type' with ID 'ami-976152f2'. It is marked as 'Free tier eligible' and has a 'Select' button. The second AMI is 'Amazon Linux 2 LTS Candidate 2 AMI (HVM), SSD Volume Type' with ID 'ami-31c7f654'. It is also marked as 'Free tier eligible' and has a 'Select' button. The third AMI is 'SUSE Linux Enterprise Server 12 SP3 (HVM), SSD Volume Type' with ID 'ami-57d3e732'. It is marked as 'Free tier eligible' and has a 'Select' button. The fourth AMI is 'Ubuntu Server 16.04 LTS (HVM), SSD Volume Type' with ID 'ami-916f59f4'. It is marked as 'Free tier eligible' and has a 'Select' button. The fifth AMI is 'Red Hat Enterprise Linux 7.5 (HVM), SSD Volume Type' with ID 'ami-03291866'. It is marked as 'Free tier eligible' and has a 'Select' button.

- Only one instance type is for free

Step 2: Choose an Instance Type

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GiB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.xlarge	8	32	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	m5.large	2	8	EBS only	Yes	Up to 10 Gigabit	Yes
<input type="checkbox"/>	General purpose	m5.xlarge	4	16	EBS only	Yes	Up to 10 Gigabit	Yes
<input type="checkbox"/>	General purpose	m5.2xlarge	8	32	EBS only	Yes	Up to 10 Gigabit	Yes
<input type="checkbox"/>	General purpose	m5.4xlarge	16	64	EBS only	Yes	Up to 10 Gigabit	Yes
<input type="checkbox"/>	General purpose	m5.12xlarge	48	192	EBS only	Yes	10 Gigabit	Yes
<input type="checkbox"/>	General purpose	m5.24xlarge	96	384	EBS only	Yes	25 Gigabit	Yes
<input type="checkbox"/>	General purpose	m4.large	2	8	EBS only	Yes	Moderate	Yes
<input type="checkbox"/>	General purpose	m4.xlarge	4	16	EBS only	Yes	High	Yes
<input type="checkbox"/>	General purpose	m4.2xlarge	8	32	EBS only	Yes	High	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

- Create public/private key by following the automated steps (first time only)

- Locate the instance ID and Public DNS from description tab

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs
	i-0cc6b1876ad6cb4b7	t2.micro	us-east-2b	running	2/2 checks ...	None	ec2-18-218-201-20.us-east-2.compute.amazonaws.com	18.218.201.20	-

Instance: i-0cc6b1876ad6cb4b7 Public DNS: ec2-18-218-201-20.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID: i-0cc6b1876ad6cb4b7  
Instance state: running  
Instance type: t2.micro  
Elastic IPs: us-east-2b  
Availability zone: us-east-2b  
Security groups: launch-wizard-1, view inbound rules  
Scheduled events: No scheduled events  
Public DNS (IPv4): ec2-18-218-201-20.us-east-2.compute.amazonaws.com  
IPv4 Public IP: 18.218.201.20  
IPv6 IPs: -  
Private DNS: ip-172-31-24-146.us-east-2.compute.internal  
Private IPs: 172.31.24.146  
Secondary private IPs: -  
VPC ID: vpc-9fd9a777

## 2) Download and install MySQL on EC2

### 1. Install MySQL server

`sudo yum install mysql-server`

```

Package Arch Version Repository
-----
Installing:
mysql-server noarch 5.5-1.6.amzn1 amzn-main
Installing for dependencies:
mysql-config x86_64 5.5.59-1.20.amzn1 amzn-main
mysql55 x86_64 5.5.59-1.20.amzn1 amzn-main
mysql55-libs x86_64 5.5.59-1.20.amzn1 amzn-main
mysql55-server x86_64 5.5.59-1.20.amzn1 amzn-main
perl-Compress-Raw-Bzip2 x86_64 2.061-3.11.amzn1 amzn-main
perl-Compress-Raw-Zlib x86_64 1:2.061-4.1.amzn1 amzn-main
perl-DBD-MySQL55 x86_64 4.023-5.23.amzn1 amzn-main
perl-DBI x86_64 1.627-4.8.amzn1 amzn-main
perl-Data-Dumper x86_64 2.145-3.5.amzn1 amzn-main
perl-IO-Compress noarch 2.061-2.12.amzn1 amzn-main
perl-Net-Daemon noarch 0.48-5.5.amzn1 amzn-main
perl-PRPC noarch 0.2020-14.7.amzn1 amzn-main

Transaction Summary
-----
Install 1 Package (+12 Dependent packages)

Total download size: 23 M
Installed size: 81 M
Is this ok [y/d/N]:

Press 'y'

Installed:
mysql-server.noarch 0:5.5-1.6.amzn1

Dependency Installed:
mysql-config.x86_64 0:5.5.59-1.20.amzn1 mysql55.x86_64 0:5.5.59-1.20.amzn1 mysql55-libs.x86_64 0:5.5.59-1.20.amzn1 mysql55-server.x86_64 0:5.5.59-1.20.amzn1
perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.11.amzn1 perl-Compress-Raw-Zlib.x86_64 1:2.061-4.1.amzn1 perl-DBD-MySQL55.x86_64 0:4.023-5.23.amzn1 perl-DBI.x86_64 0:1.627-4.8.amzn1
perl-Data-Dumper.x86_64 0:2.145-3.5.amzn1 perl-IO-Compress.noarch 0:2.061-2.12.amzn1 perl-Net-Daemon.noarch 0:0.48-5.5.amzn1 perl-PRPC.noarch 0:0.2020-14.7.amzn1

Complete!
[ec2-user@ip-172-31-19-199 ~]$

```

### 2. To run MySQL

To check status:

`Sudo service mysqld status`

```

[ec2-user@ip-172-31-19-199 ~]$ sudo service mysqld status
mysqld is stopped
[ec2-user@ip-172-31-19-199 ~]$

```

To start MySQL:

`Sudo service mysqld start`

```
[ec2-user@ip-172-31-19-199 ~]$ sudo service mysqld start
Initializing MySQL database: Installing MySQL system tables...
180514 19:20:15 [Note] Ignoring --secure-file-priv value as server is running with --bootstrap.
180514 19:20:15 [Note] /usr/libexec/mysqld55/mysqld (mysqld 5.5.59) starting as process 23353 ...
OK
Filling help tables...
180514 19:20:15 [Note] Ignoring --secure-file-priv value as server is running with --bootstrap.
180514 19:20:15 [Note] /usr/libexec/mysqld55/mysqld (mysqld 5.5.59) starting as process 23360 ...
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:

/usr/libexec/mysqld55/mysqladmin -u root password 'new-password'
/usr/libexec/mysqld55/mysqladmin -u root -h ip-172-31-19-199 password 'new-password'

Alternatively you can run:
/usr/libexec/mysqld55/mysql_secure_installation

which will also give you the option of removing the test
databases and anonymous user created by default. This is
strongly recommended for production servers.

See the manual for more instructions.

You can start the MySQL daemon with:
cd /usr ; /usr/libexec/mysqld55/mysqld_safe &

You can test the MySQL daemon with mysql-test-run.pl
cd /usr/mysql-test ; perl mysql-test-run.pl

Please report any problems at http://bugs.mysql.com/

Starting mysqld: [ OK ]
```

### 3. Setup password

```
mysqladmin -u root password 'new_password'
```

### 4. Create a database

```
mysqladmin -u root -p create 'db_name'
```

```
[ec2-user@ip-172-31-19-199 ~]$ mysqladmin -u root -p create 'ec2sql'
Enter password:
[ec2-user@ip-172-31-19-199 ~]$
```

### 5. Login as a admin

```
mysql -u root -p
```

```
[ec2-user@ip-172-31-26-160 ~]$ mysql -u root -p
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)
[ec2-user@ip-172-31-26-160 ~]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 5.5.59 MySQL Community Server (GPL)
```

### 6. Give privileges

```
GRANT ALL ON *.* TO 'your_mysql_name'@'your_client_host' identified by 'password';
```



```
mysql> grant all privileges on *.* to 'paul2017'@'localhost' identified by 'paul2017';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

#### 7. Create table and insert for testing

```
mysql> use db1
Database changed
mysql> create table test (id smallint unsigned not null auto_increment, name varchar(20) not null, constraint
key (id));
Query OK, 0 rows affected (0.00 sec)

mysql> insert into example (id, name) values (null, 'test data');
ERROR 1146 (42S02): Table 'db1.example' doesn't exist
mysql> insert into test (id, name) values (null, 'test data');
Query OK, 1 row affected (0.01 sec)

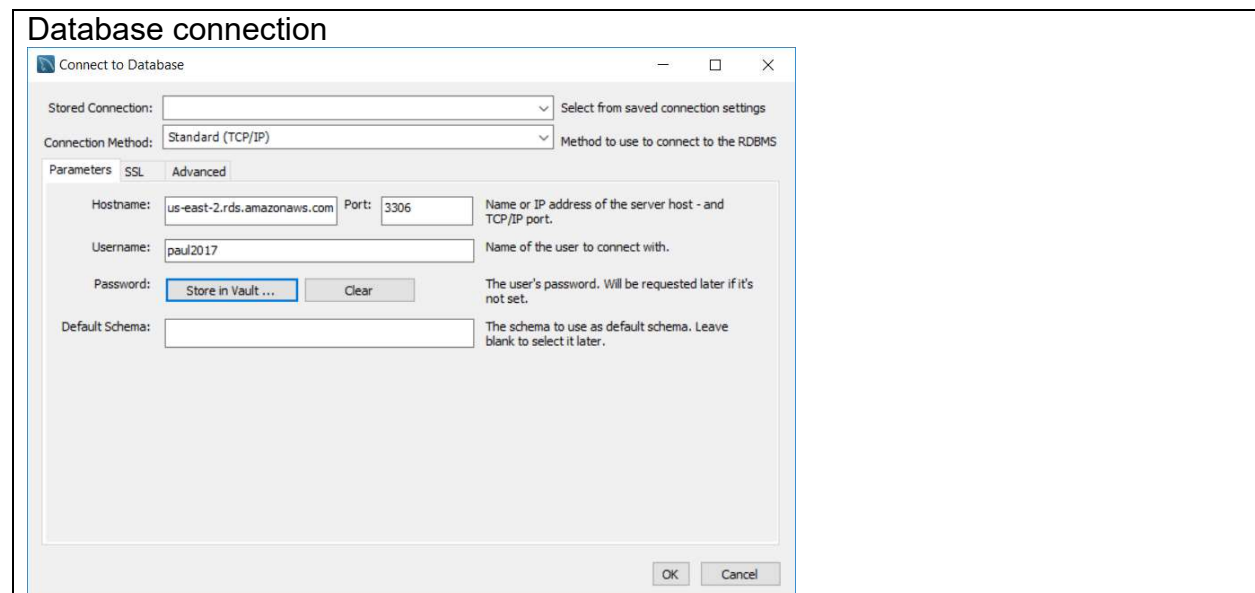
mysql> select * from test;
+----+-----+
| id | name      |
+----+-----+
| 1  | test data |
+----+-----+
1 row in set (0.00 sec)

mysql>
```

3) Download (if you don't already have) MySQL Workbench on your local machine

Please see the answer 1A above.

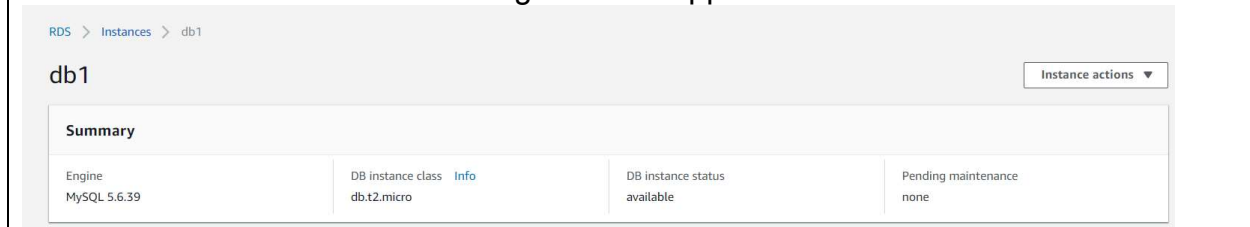
4) Connect MySQL Workbench to EC2 instance (you will need to create keys)





## 5) Create MySQL instance on RDS

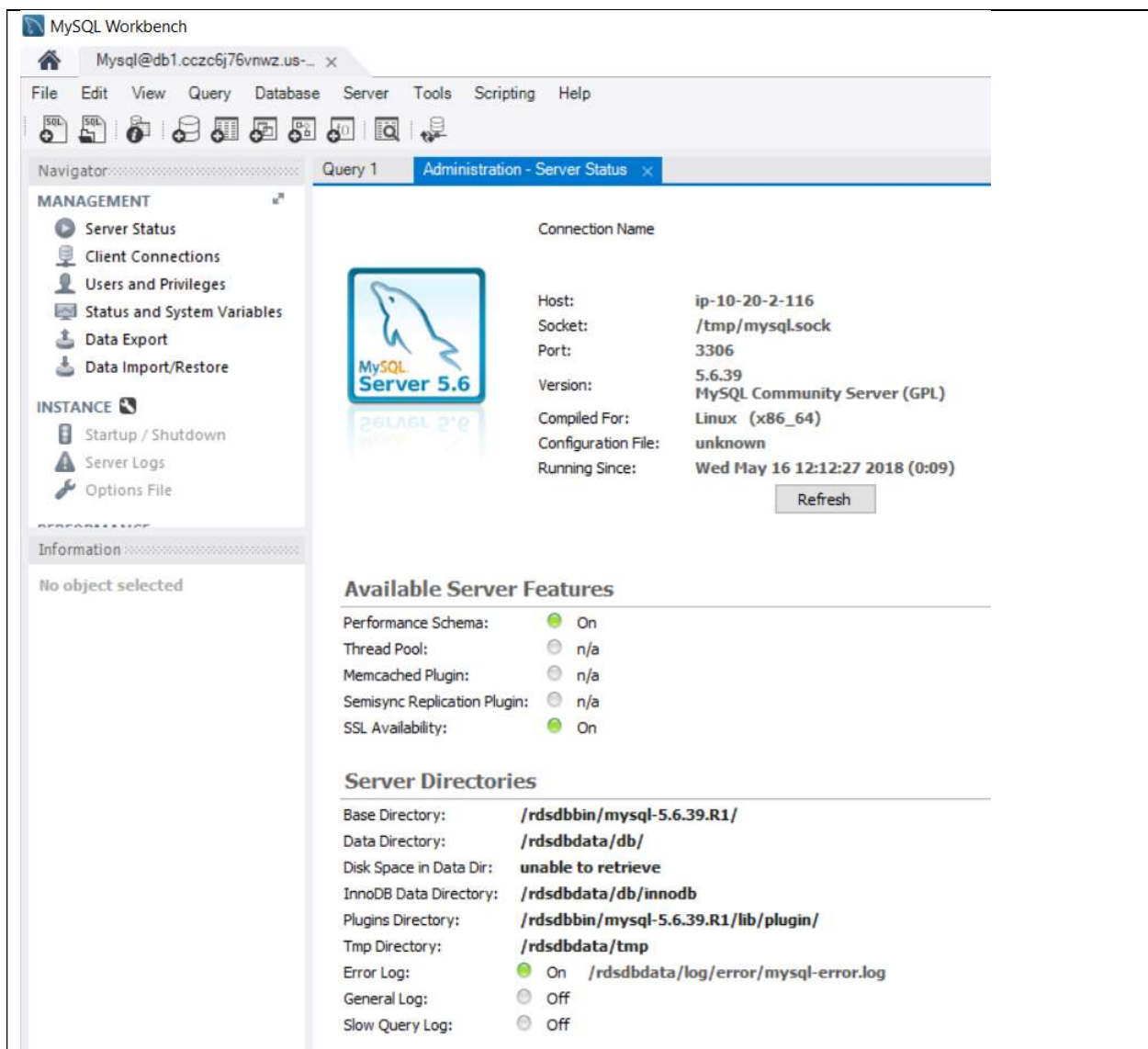
AWS RDS instance is created using the RDS app on AWS website.



The screenshot shows the AWS RDS console for an instance named 'db1'. The breadcrumb navigation is 'RDS > Instances > db1'. There is an 'Instance actions' dropdown menu. Below is a 'Summary' table with the following data:

Summary			
Engine MySQL 5.6.39	DB instance class db.t2.micro <a href="#">Info</a>	DB instance status available	Pending maintenance none

## 6) Connect your local MySQL Workbench to MySQL on RDS



The screenshot shows the MySQL Workbench 'Administration - Server Status' window. The connection name is 'Mysql@db1.cczc6j76vwnz.us-...'. The left sidebar shows the 'MANAGEMENT' and 'INSTANCE' sections. The main area displays the MySQL Server 5.6 logo and the following connection details:

Connection Name	
Host:	ip-10-20-2-116
Socket:	/tmp/mysql.sock
Port:	3306
Version:	5.6.39 MySQL Community Server (GPL)
Compiled For:	Linux (x86_64)
Configuration File:	unknown
Running Since:	Wed May 16 12:12:27 2018 (0:09)

There is a 'Refresh' button below the connection details. Below the connection details, there are two sections: 'Available Server Features' and 'Server Directories'.

**Available Server Features**

Feature	Status
Performance Schema:	On
Thread Pool:	n/a
Memcached Plugin:	n/a
Semisync Replication Plugin:	n/a
SSL Availability:	On

**Server Directories**

Directory	Path
Base Directory:	/rdsdbbin/mysql-5.6.39.R1/
Data Directory:	/rdsdbdata/db/
Disk Space in Data Dir:	unable to retrieve
InnoDB Data Directory:	/rdsdbdata/db/innodb
Plugins Directory:	/rdsdbbin/mysql-5.6.39.R1/lib/plugin/
Tmp Directory:	/rdsdbdata/tmp
Error Log:	On /rdsdbdata/log/error/mysql-error.log
General Log:	Off
Slow Query Log:	Off

Test query on Workbench from AWS RDS

The screenshot displays the MySQL Workbench interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The left sidebar contains a 'MANAGEMENT' section with options like Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, and Data Import/Restore. Below this is an 'INSTANCE' section with Startup / Shutdown, Server Logs, and Options File. The main workspace is titled 'Query 1 x Administration - Server Status' and shows the SQL query: `select User, Host from mysql.user`. The query results are displayed in a table with two columns: 'User' and 'Host'. The results are as follows:

User	Host
paul2017	%
mvsol.svs	localhost
rdsadmin	localhost
NULL	NULL

At the bottom of the interface, the 'Output' tab is selected, showing the 'Action Output' for the query execution. The output table has columns for '#', 'Time', and 'Action'. The first row indicates a successful execution at 12:21:22.

#	Time	Action
1	12:21:22	select User, Host from mysql.user LIMIT 0, 1000

Collaborators: None.

Resources:

Putty connection to EC2

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html>

MySQL install on EC2

<http://text-analytics101.rxnlp.com/2013/11/how-to-install-mysql-on-amazon-ec2.html>

<https://stackoverflow.com/questions/39025524/how-to-install-mysql-5-7-on-amazon-ec2>

AWS RDS instance creation

<https://aws.amazon.com/getting-started/tutorials/create-mysql-db/>