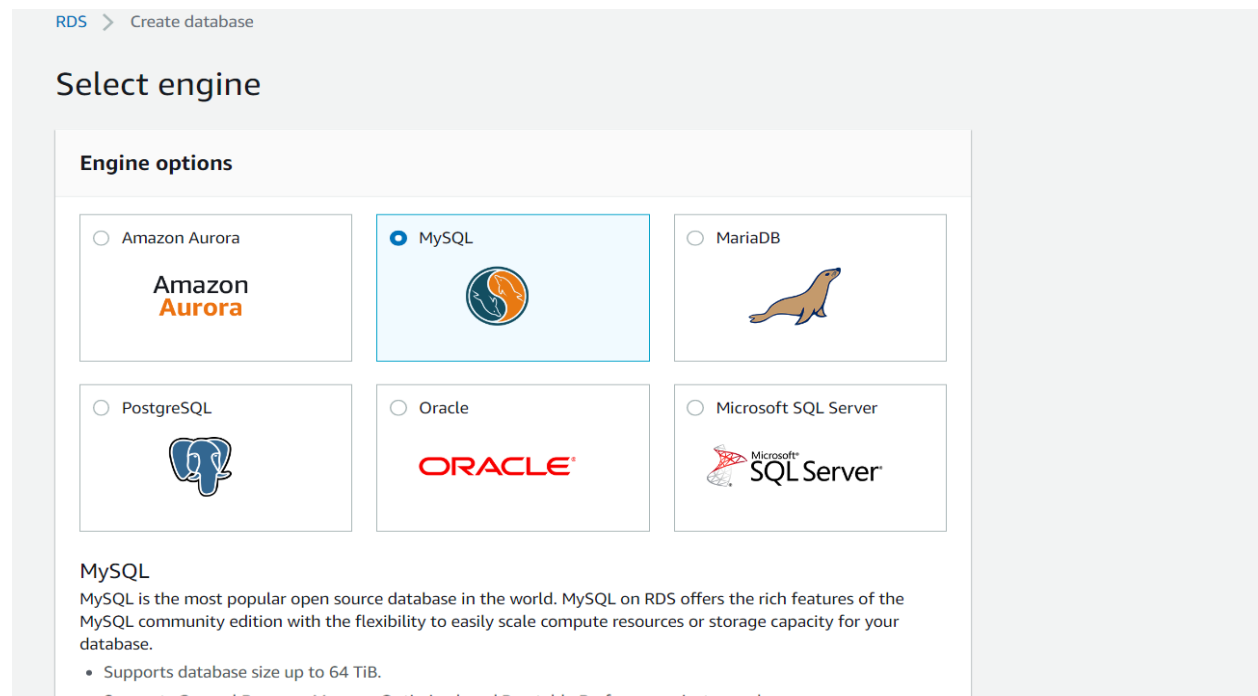


## Lab 6

### Run the PHP Code on EC2 instance that retrieve data from RDS Server

**Step 1:** Log in to AWS Console and Select RDS

**Step 2:** There are different database engines are available. check mark on free tier applicable on the bottom. Select MySQL Database Engine.



**Step 3:** Create Database Identifier

**Step 4:** Set Master username and Master Password for database

1000 GIB  
(Minimum: 21 GIB, Maximum: 16384 GIB)

### Settings

**DB instance identifier** [Info](#)  
Specify a name that is unique for all DB instances owned by your AWS account in the current region.  
  
DB instance identifier is case insensitive, but stored as all lower-case, as in "mydbinstance". Must contain from 1 to 63 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

**Master username** [Info](#)  
Specify an alphanumeric string that defines the login ID for the master user.  
  
Master Username must start with a letter. Must contain 1 to 16 alphanumeric characters.

**Master password** [Info](#) **Confirm password** [Info](#)  
   
Master Password must be at least eight characters long, as in "mypassword". Can be any printable ASCII character except "/", "", or "@".

[Cancel](#) [Previous](#) [Next](#)

**Step 5:** Enter the database name,Port in the database options panel.then click create database.

### Database options

**Database name** [Info](#)  
  
Note: if no database name is specified then no initial MySQL database will be created on the DB Instance.

**Port** [Info](#)  
TCP/IP port the DB instance will use for application connections.

**DB parameter group** [Info](#)

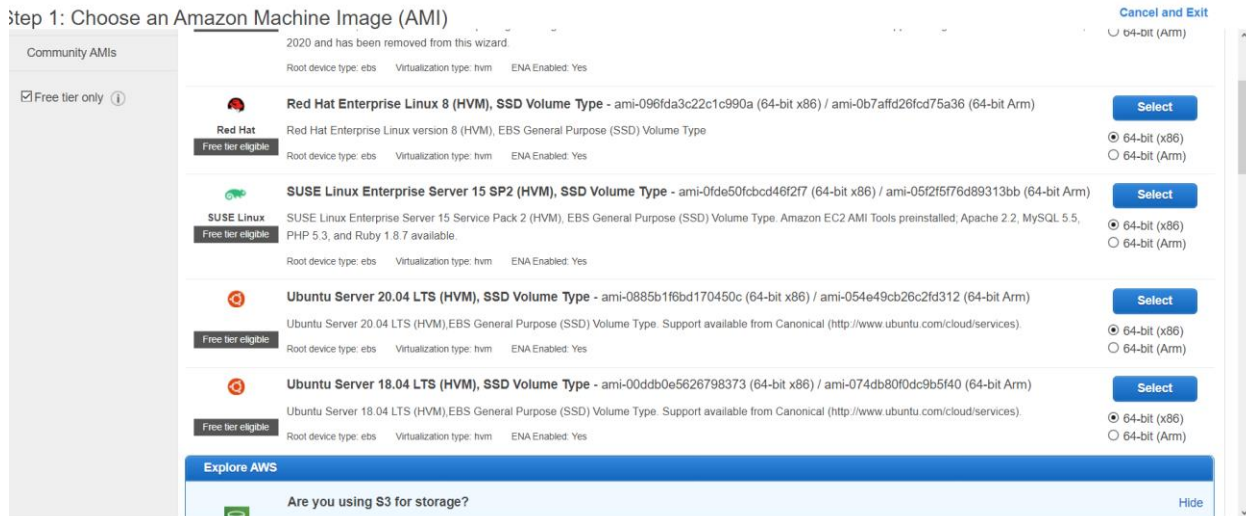
**Option group** [Info](#)

**IAM DB authentication** [Info](#)  
☐ Enable IAM DB authentication  
Manage your database user credentials through AWS IAM users and roles.  
☒ Disable

### Encryption

**Step 6:** Now go to EC2 and Select Ubuntu Server AMI

### Step 1: Choose an Amazon Machine Image (AMI)



### Step 7: Choose Instance type as t2.micro which is free tier eligible

#### Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: **All instance families** **Current generation** **Show/Hide Columns**

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes

### Step 8: In Security Group Add

- HTTP Protocol and Set Source anywhere
- HTTPS Protocol and Set Source anywhere
- MYSQL/AURORA and Set Source anywhere
- All traffic and set source anywhere

Click on review and launch

### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group

☐ Select an existing security group

Security group name: launch-wizard-3

Description: launch-wizard-3 created 2021-01-02T19:52:49.330+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere 0.0.0.0/0, :::0	e.g. SSH for Admin Desktop
All traffic	All	0 - 65535	Anywhere 0.0.0.0/0, :::0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Anywhere 0.0.0.0/0, :::0	e.g. SSH for Admin Desktop
MySQL/Aurora	TCP	3306	Anywhere 0.0.0.0/0, :::0	e.g. SSH for Admin Desktop

Add Rule



#### Warning

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Previous](#) [Review and Launch](#)

### Step 9: Create a new Key Pair and Download the Key pair

#### Select an existing key pair or create a new key pair



A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Create a new key pair

Key pair name

labtest

Download Key Pair



You have to download the **private key file** (\*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

[Cancel](#)

[Launch Instances](#)

Click on launch instances

### Step 10: Connect Ubuntu VM using ssh client using Command :

```
ssh -I <.pem file>@dns
```

```
ubuntu@ip-172-31-50-175:~$ ssh ubuntu@ec2-18-207-140-250.compute-1.amazonaws.com
Microsoft Windows [Version 10.0.19042.685]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Akash>H:\Downloads>ssh -i labtest.pem ubuntu@ec2-18-207-140-250.compute-1.amazonaws.com
The authenticity of host 'ec2-18-207-140-250.compute-1.amazonaws.com (18.207.140.250)' can't be established.
ECDSA key fingerprint is SHA256:rwM0JUBZMB08mcuPukfIUz2C4AfOcm1627/T5hBzKB.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-18-207-140-250.compute-1.amazonaws.com,18.207.140.250' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-1029-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Sat Jan  2 14:26:15 UTC 2021

System load:  0.59          Processes:           101
Usage of /:   16.8% of 7.69GB Users logged in:       0
Memory usage: 20%          IPv4 address for eth0: 172.31.50.175
Swap usage:   0%

1 update can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-50-175:~$
```

**Step 11:** Now install apache2 software in Ubuntu VM using Command :

`sudo apt install apache2 -y`

```
ubuntu@ip-172-31-50-175:~$ sudo apt install apache2 -y
Get:13 http://security.ubuntu.com/ubuntu focal-security/universe amd64 c-n-f Metadata [9364 B]
Get:16 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 Packages [1256 B]
Get:17 http://security.ubuntu.com/ubuntu focal-security/multiverse Translation-en [540 B]
Get:18 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 c-n-f Metadata [116 B]
Fetched 17.7 kB in 3s (5047 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
66 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-50-175:~$ sudo apt install apache2 -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.2-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.2-0 ssl-cert
0 upgraded, 11 newly installed, 0 to remove and 66 not upgraded.
Need to get 1865 kB of archives.
After this operation, 8080 kB of additional disk space will be used.
Get:1 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal/main amd64 libapr1 amd64 1.6.5-1ubuntu1 [91.4 kB]
Get:2 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal/main amd64 libaprutil1 amd64 1.6.1-4ubuntu2 [84.7 kB]
Get:3 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.1-4ubuntu2 [10.5 kB]
Get:4 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal/main amd64 libaprutil1-ldap amd64 1.6.1-4ubuntu2 [8736 B]
Get:5 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal/main amd64 libjansson4 amd64 2.12-1build1 [28.9 kB]
Get:6 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal/main amd64 liblua5.2-0 amd64 5.2.4-1.1build3 [106 kB]
Get:7 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2-bin amd64 2.4.41-4ubuntu3.1 [1180 kB]
Get:8 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2-data all 2.4.41-4ubuntu3.1 [158 kB]
Get:9 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2-utils amd64 2.4.41-4ubuntu3.1 [83.8 kB]
Get:10 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2 amd64 2.4.41-4ubuntu3.1 [95.5 kB]
Get:11 http://us-east-1-ec2.archive.ubuntu.com/ubuntu focal/main amd64 ssl-cert all 1.0.39 [17.0 kB]
Fetched 1865 kB in 8s (15.2 MB/s)
Preconfiguring packages ...
Selecting previously unselected package libapr1:amd64.
(Reading database ... 59940 files and directories currently installed.)
Preparing to unpack .../00-libapr1_1.6.5-1ubuntu1_amd64.deb ...
Unpacking libapr1:amd64 (1.6.5-1ubuntu1) ...
Selecting previously unselected package libaprutil1:amd64.
Preparing to unpack .../01-libaprutil1_1.6.1-4ubuntu2_amd64.deb ...
Unpacking libaprutil1:amd64 (1.6.1-4ubuntu2) ...
```

**Step 12:** Now install mysql-server in Ubuntu VM using Command:

`sudo apt install mysql-server -y`

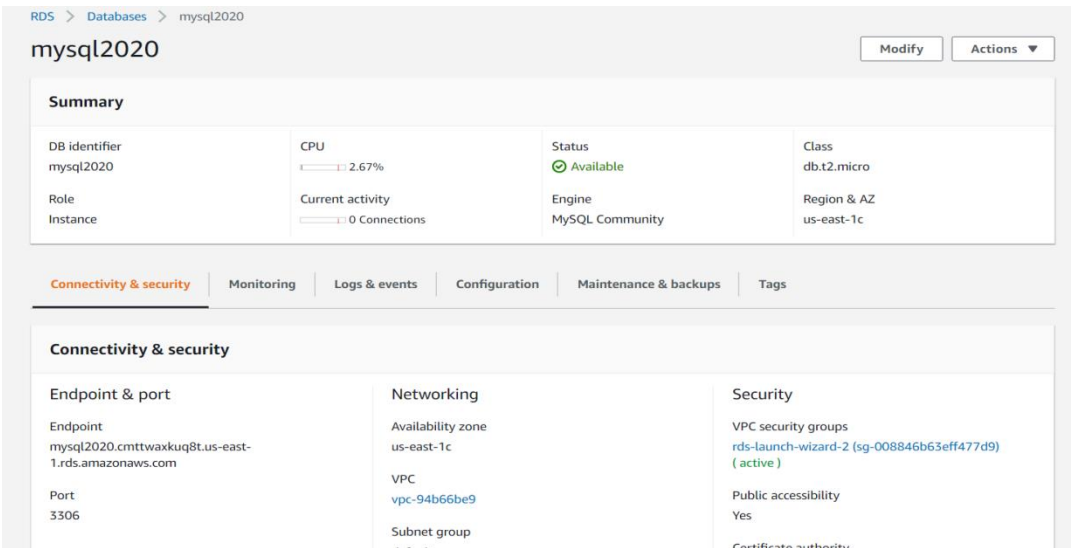
**Step 13:** Install php in Ubuntu VM using Command :

`sudo apt-get install php libapache2-mod-php php-mysql php-curl php-gd php-json php-zip php-mbstring`

```
Select ubuntu@ip-172-31-50-175: ~
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for ufw (0.36-0) ...
Processing triggers for systemd (245.4-0ubuntu3.2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.1) ...
ubuntu@ip-172-31-50-175:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2021-01-02 14:27:59 UTC; 54s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2660 (apache2)
     Tasks: 55 (limit: 1164)
    Memory: 2.3M
    CGroup: /system.slice/apache2.service
            └─2660 /usr/sbin/apache2 -k start
              2662 /usr/sbin/apache2 -k start
              2663 /usr/sbin/apache2 -k start

Jan 02 14:27:59 ip-172-31-50-175 systemd[1]: Starting The Apache HTTP Server...
Jan 02 14:27:59 ip-172-31-50-175 systemd[1]: Started The Apache HTTP Server.
ubuntu@ip-172-31-50-175:~$ sudo apt install; mysql-server -y
-bash: syntax error near unexpected token `;'
ubuntu@ip-172-31-50-175:~$ sudo apt install mysql-server -y
```

**Step 14:** Now go to connectivity and security panel in Credet database and copy the Endpoint and port number .



**Step 15:** Now connect Mysql database using following command

```
mysql -h <End point> -P 3306 -u <UserName> -p
```

It will ask for Master Password , enter the master password.

**Step 16:** Now Mysql database which we created inside AWS is connected to Mysql database

```
ubuntu@ip-172-31-38-93: ~  
ubuntu@ip-172-31-38-93:~$ mysql -h mysql2020.cmttwaxkuq8t.us-east-1.rds.amazonaws.com -P 3306  
-u mysql2020 -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 21  
Server version: 8.0.20 Source distribution  
  
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.  
  
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> create database bookdb;  
Query OK, 1 row affected (0.02 sec)  
  
mysql> use b_
```

### Step 17: Create table in mysql database

```
ubuntu@ip-172-31-38-93: ~  
-u mysql2020 -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 21  
Server version: 8.0.20 Source distribution  
  
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.  
  
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> create database bookdb;  
Query OK, 1 row affected (0.02 sec)  
  
mysql> use bookdb;  
Database changed  
mysql> create table books(  
-> id int AUTO_INCREMENT primary key,  
-> bookname varchar(255),  
-> price varchar(255)  
-> );  
Query OK, 0 rows affected (0.03 sec)  
  
mysql>
```

### Step 18: Now go to /var/www/html folder and create new sample.php file inside ubuntu VM.

```
ubuntu@ip-172-31-38-93: /var/www/html  
ubuntu@ip-172-31-38-93:/var/www/html$ sudo nano sample.php
```

### Step 19: Write the php code inside sample.php to take input and store in database

```
ubuntu@ip-172-31-38-93: /var/www/html
GNU nano 4.8 sample.php Modified
<html>
<head>
  <title>Cloud Computing Lab</title>
</head>
<body>

<form method="post">
<p>Name: <input type="text" name="bname"></p>
<p>Price: <input type="text" name="bprice"></p>
<p><input type="submit" name="sub"></p>
</form>
<?php
$script = new script();
if(isset($_POST['sub'])){
    //$script = new script();
    $script->add($_POST['bname'],$_POST['bprice']);
}
$script->getdata();

?>

</body>
</html>
```

**Step 20:** Now copy the public ip of ubuntu instance and paste in address bar.php application is live and it stores the details in mysql database .

The screenshot shows a web browser window with the address bar displaying "54.167.245.184/sample.php". The page content includes a form with two input fields labeled "ID:" and "NAME:", followed by a "Submit" button. Below the form, a message "Data added!" is displayed, followed by a table with two columns: "id" and "name". The table contains two rows of data: "100 Ajith" and "200 Clark kent".

id	name
100	Ajith
200	Clark kent