

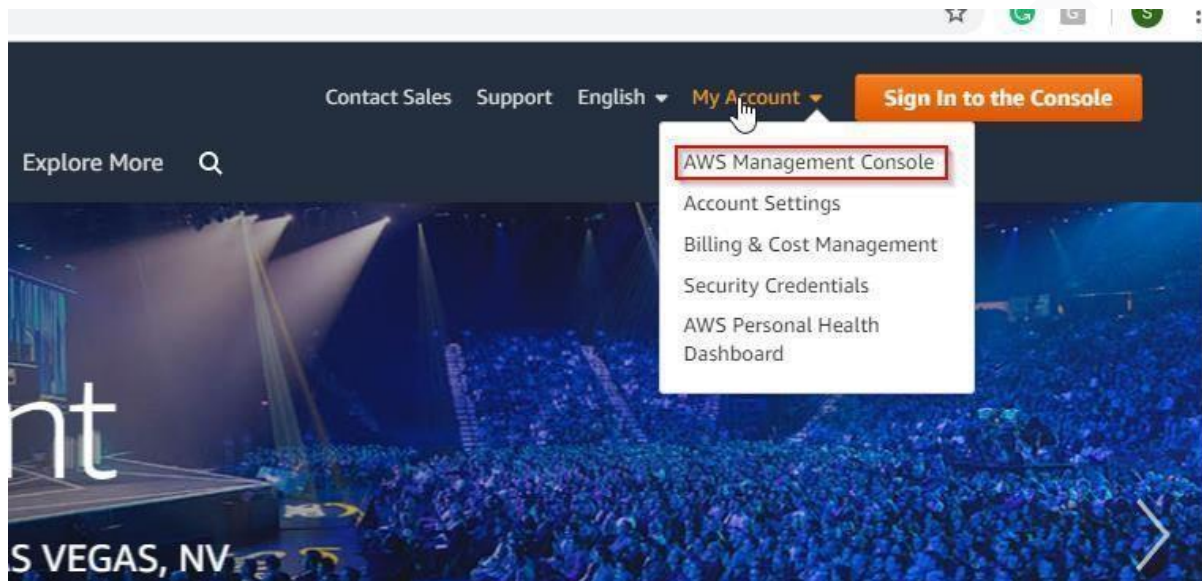


Project- 1: Solution

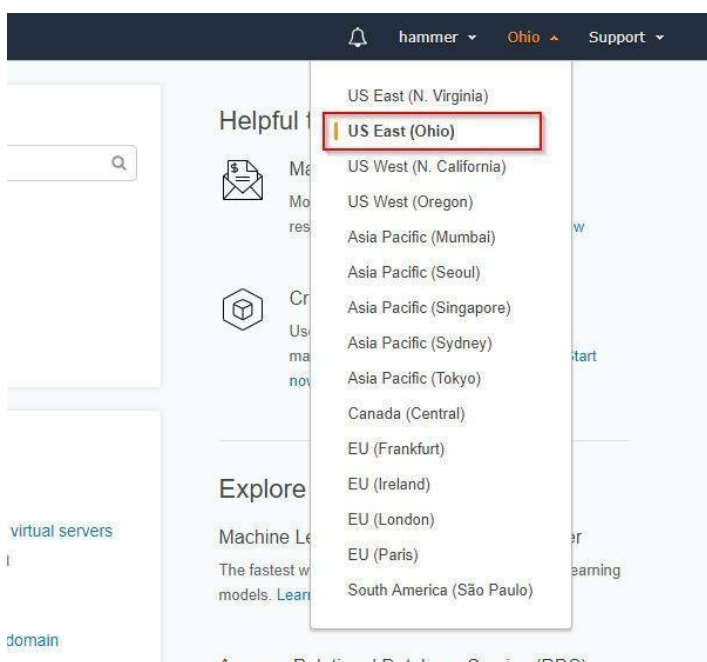
Connect your system with your EC2 instance:

1. First you need to install PuTTY on your system and then connect it with your EC2 instance.
2. Below are the steps for it:

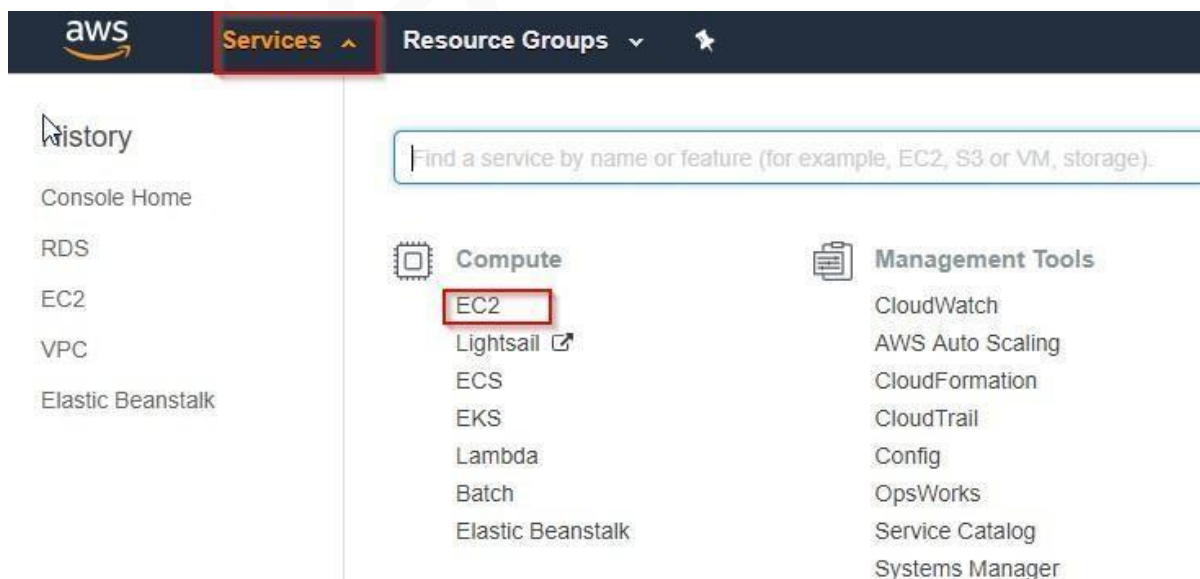
a. First sign into the AWS Management Console



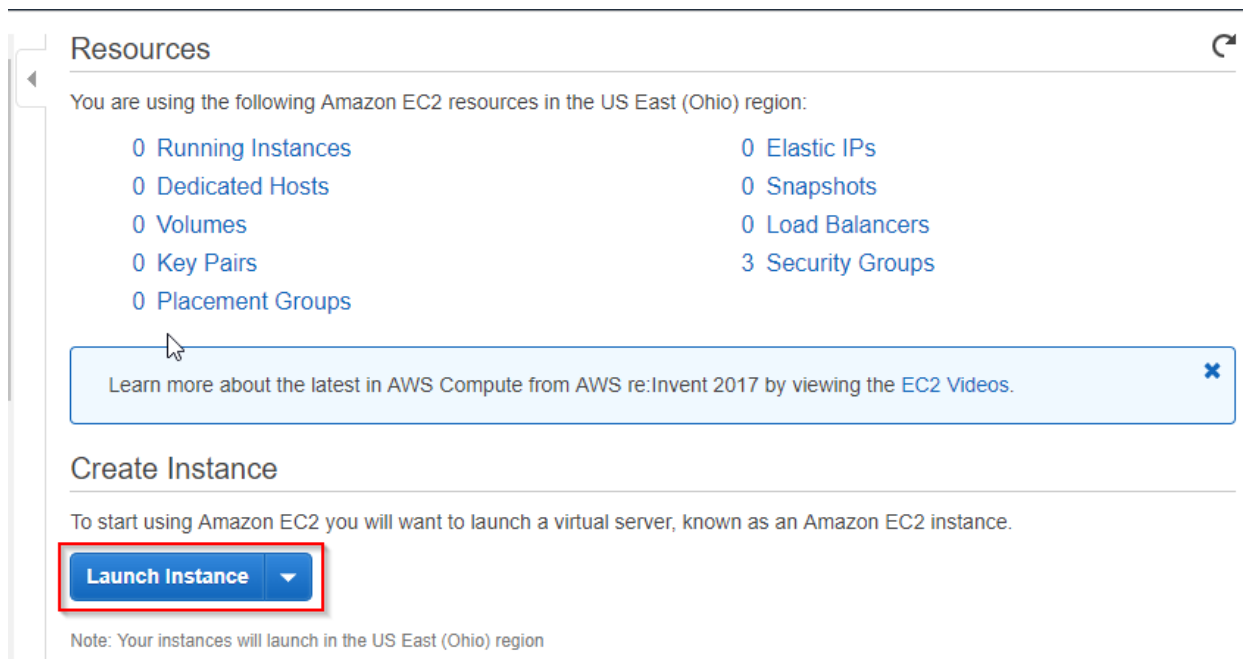
b. Select any region you want. We've selected Ohio here



c. In the **Services** section, you'll see **Compute** where you need to choose **EC2**



- d. Then in the Create Instance section, select the **Launch Instance** option



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
- 3 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

Note: Your instances will launch in the US East (Ohio) region

- e. Then Select an **AMI** or **Amazon Machine Image**



Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0f65671a86f061fcd

Free tier eligible

Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Root device type: ebs Virtualization type: hvm

Select

64-bit (x86)

f. Choose your instance type. We're choosing Free tier for demo purposes

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 types Current generation Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	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 Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

g. Configure your instance details and then select the **Add storage** option

aws Services Resource Groups

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: Launch into Auto Scaling Group

Purchasing option: ☐ Request Spot instances

Network: vpc-7f040317 (default) Create new VPC

Subnet: No preference (default subnet in any Availability Zone) Create new subnet

Auto-assign Public IP: Use subnet setting (Enable)

Placement group: ☐ Add instance to placement group.

Capacity Reservation: Open Create new Capacity Reservation

IAM role: None Create new IAM role

Shutdown behavior: Stop

Cancel Previous Review and Launch Next: Add Storage

h. Then click on Add Tags

Volume Type <small>i</small>	Device <small>i</small>	Snapshot <small>i</small>	Size (GiB) <small>i</small>	Volume Type <small>i</small>	IOPS <small>i</small>	Throughput (MB/s) <small>i</small>	Delete on Termination <small>i</small>	Encrypted <small>i</small>
Root	/dev/sda1	snap-0474571d378f0fac2	<input type="text" value="8"/>	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

i. Add Tags, name the Key and Value then click Configure Security Group

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key <small>(127 characters maximum)</small>	Value <small>(255 characters maximum)</small>	Instances <small>i</small>	Volumes <small>i</small>
intellipaat	ec2-demo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

j. Keep the configuration of security group as it

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:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

k. Then click Review & Launch

Cancel

Previous

Review and Launch

l. Then directly Launch it

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 7: Review Instance Launch

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

[Edit security groups](#)

Security group name: launch-wizard-1
Description: launch-wizard-1 created 2018-11-16T14:27:39.538+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	

Instance Details

[Edit instance details](#)

Storage

[Edit storage](#)

Tags

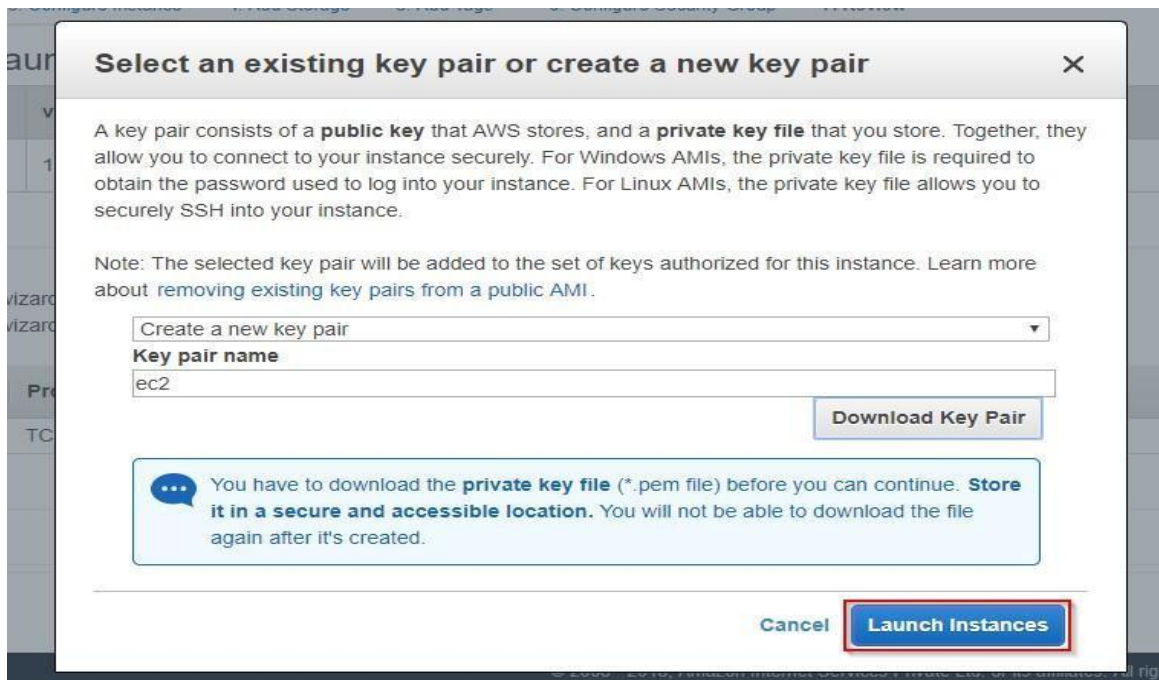
[Edit tags](#)

Cancel

Previous

Launch

m. Then Create a key pair, download it and then Launch your instance

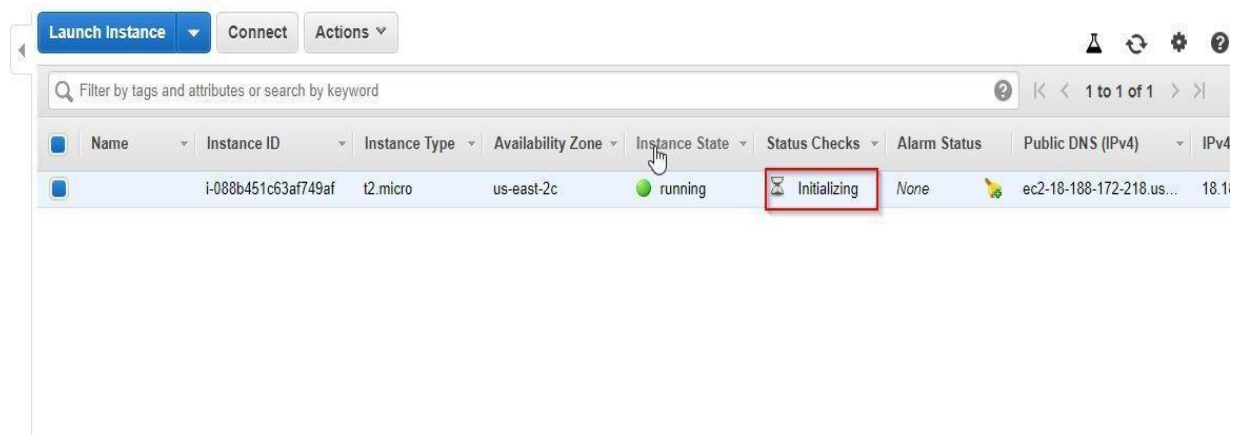


n. Status

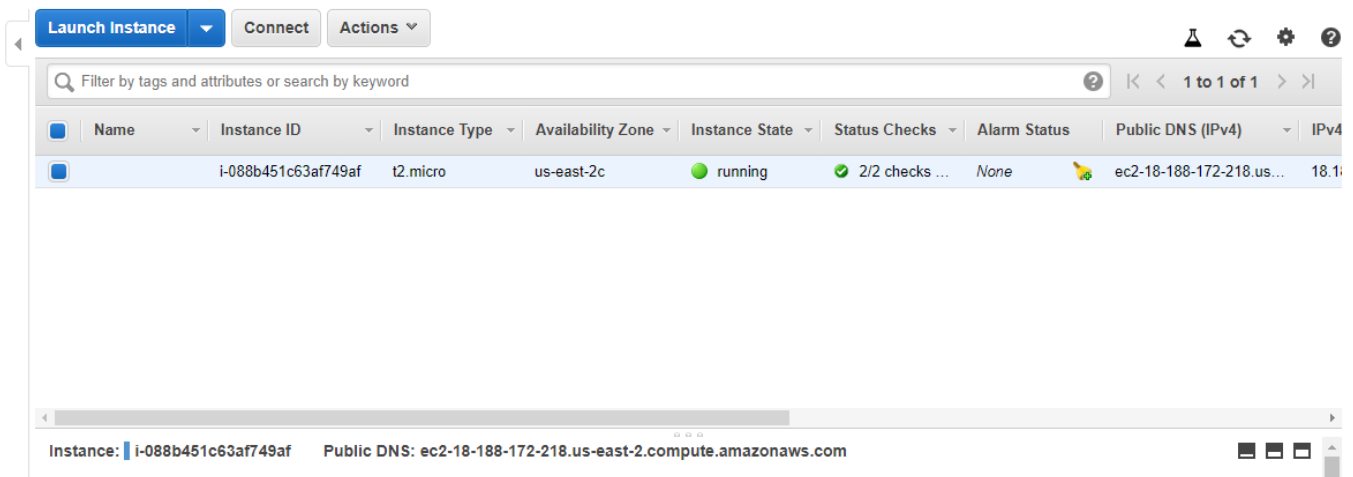
Launch Status



o. You will be able to see in your status that your Instance is on Initializing stage



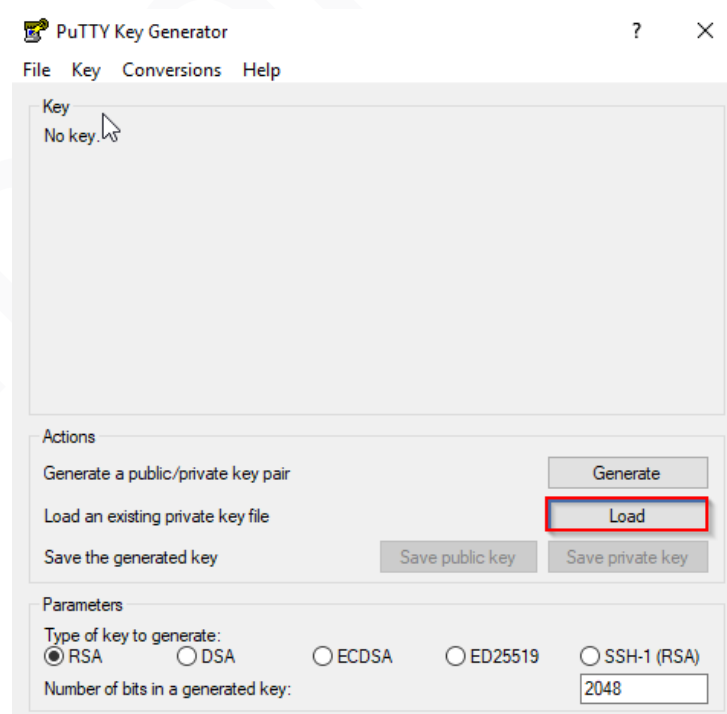
p. Then after few minutes, you will see that now your instance is in running stage



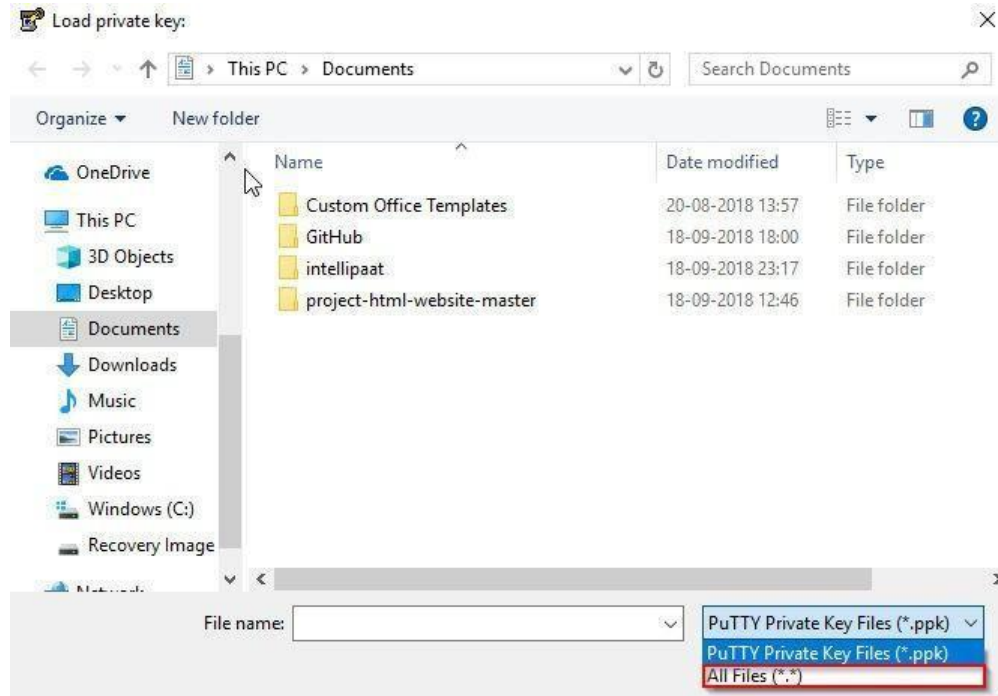
q. Now it's time to convert your private key using PuTTYgen

PuTTY won't be able to support this .pem file, so you'd require a PuTTY gen tool which can convert your .pem file into .ppk format, because you need a .ppk file in order to connect it with your instance.

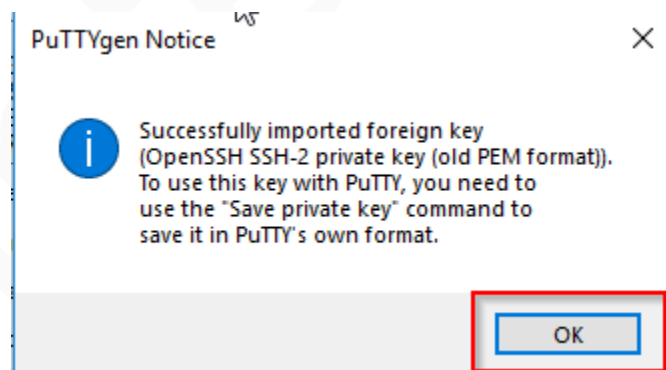
r. Click Load in your PuTTY gen



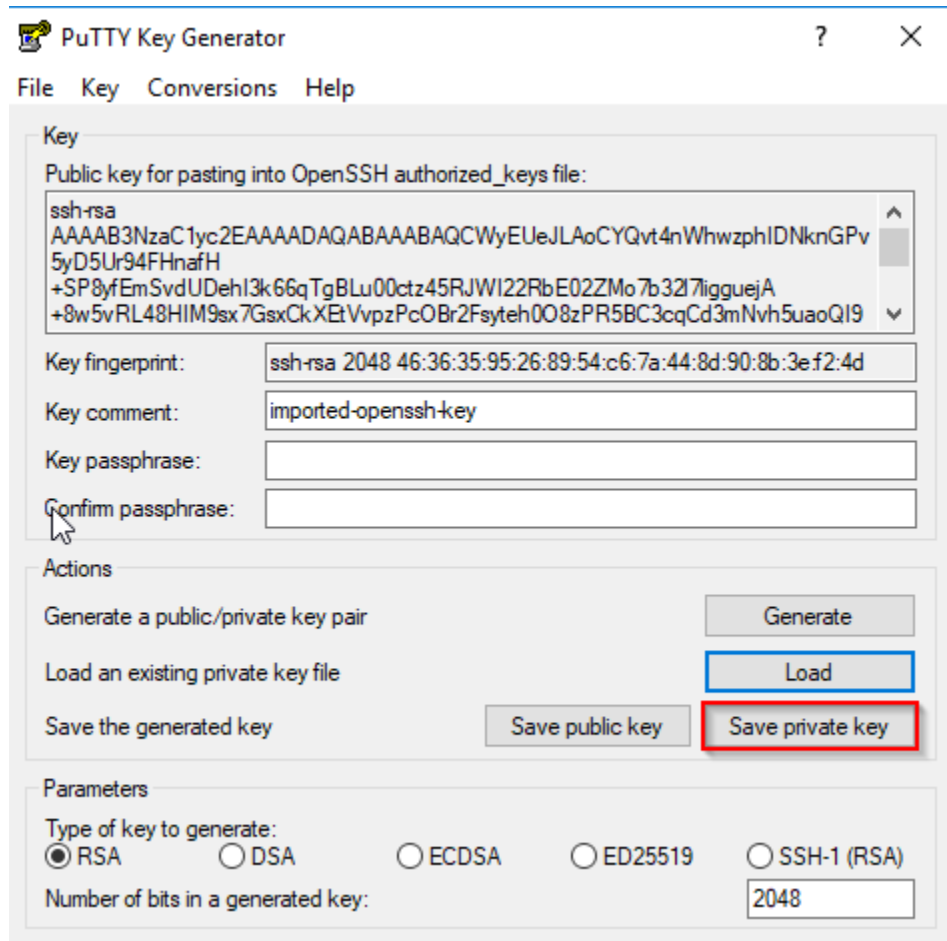
- s. PuTTYgen key always shows the .ppk format file, so go to the right bottom bar and select the All files option as shown below



- t. Then select the folder where you downloaded this keypair and load it there
- u. You will see this option then click OK



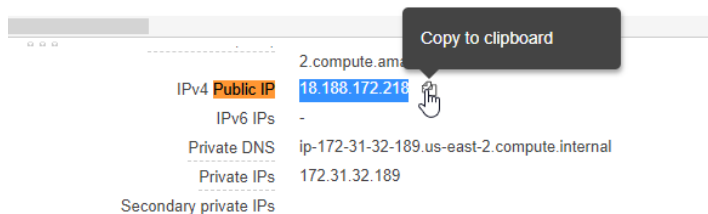
- v. Then click on Save the Private key, PuTTY gen will give a warning about saving the key without Key passphrase, click Yes and specify the same name for your file that you gave it in the key pair



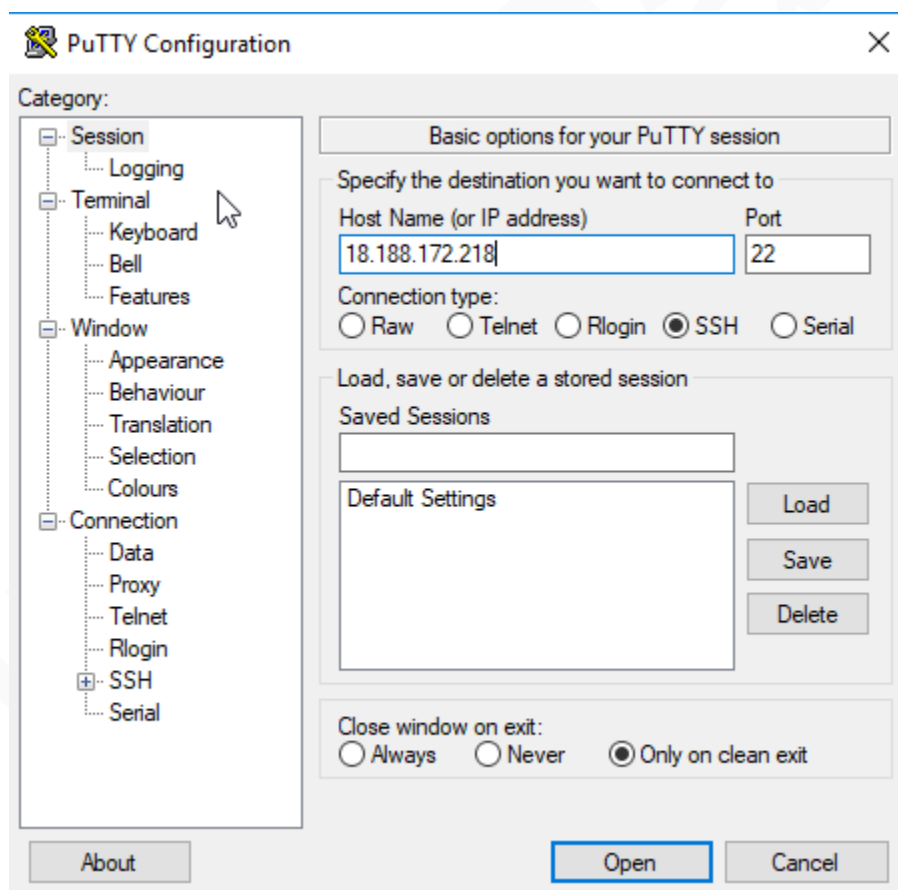
- w. Now you will see that in your folder, the .ppk file is already added with that name you had given (in our case, it's ec2)

Connecting to your EC2 Instance using SSH & PuTTY:

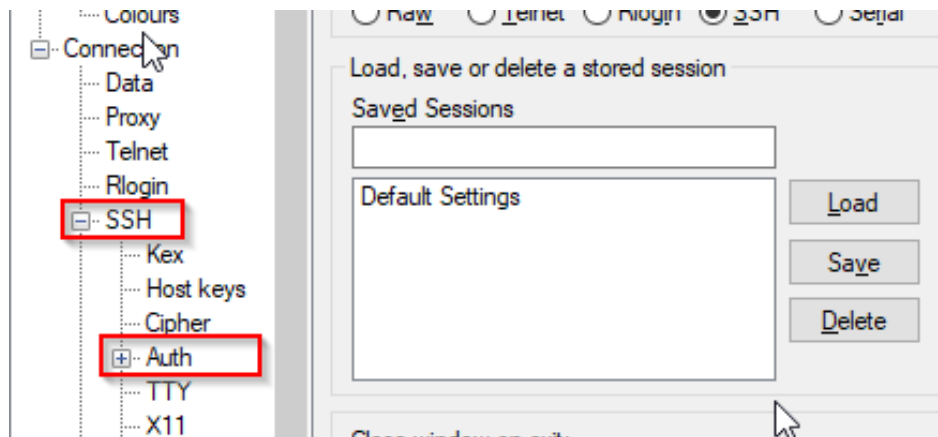
- First open PuTTY.exe then in the Host Name box, add the Public IP of your Instance



- Copy paste this Public IP in your PuTTY Hostname



- Then in the category list, expand the SSH and Click on AUTH (but don't expand it)



- Then Click Open
 - Login as per your OS, in our case it is ubuntu, so we will Login as: Ubuntu

```
ubuntu@ip-172-31-32-189: ~  
Here's a step-by-step tutorial for a rainy weekend, or a startup.  
- https://bit.ly/secure-kiosk  
  
Get cloud support with Ubuntu Advantage CloudGuest:  
http://www.ubuntu.com/business/services/cloud  
  
0 packages can be updated.  
0 updates are security updates.  
  
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-172-31-32-189:~$
```

- First Update your system using the command
sudo apt-get update
- Then use this command in PuTTY to install Apache2
sudo apt-get install apache2
- Then install php-mysql using the following command
sudo add-apt-repository -y ppa:ondrej/php
sudo apt install php5.6 mysql-client php5.6-mysqli

Now everything is updated in your system

```
Creating config file /etc/php/5.6/mods-available/pdo_mysql.ini with new version
Creating config file /etc/php/5.6/mods-available/mysql.ini with new version
Setting up php5.6-json (5.6.38-3+ubuntu18.04.1+deb.sury.org+1) ...

Creating config file /etc/php/5.6/mods-available/json.ini with new version
Setting up mysql-client (5.7.24-0ubuntu0.18.04.1) ...
Setting up php5.6-cli (5.6.38-3+ubuntu18.04.1+deb.sury.org+1) ...
update-alternatives: using /usr/bin/php5.6 to provide /usr/bin/php (php) in auto mode
update-alternatives: using /usr/bin/phar5.6 to provide /usr/bin/phar (phar) in auto mode
update-alternatives: using /usr/bin/phar.phar5.6 to provide /usr/bin/phar.phar (phar.phar) in auto mode

Creating config file /etc/php/5.6/cli/php.ini with new version
Setting up libapache2-mod-php5.6 (5.6.38-3+ubuntu18.04.1+deb.sury.org+1) ...

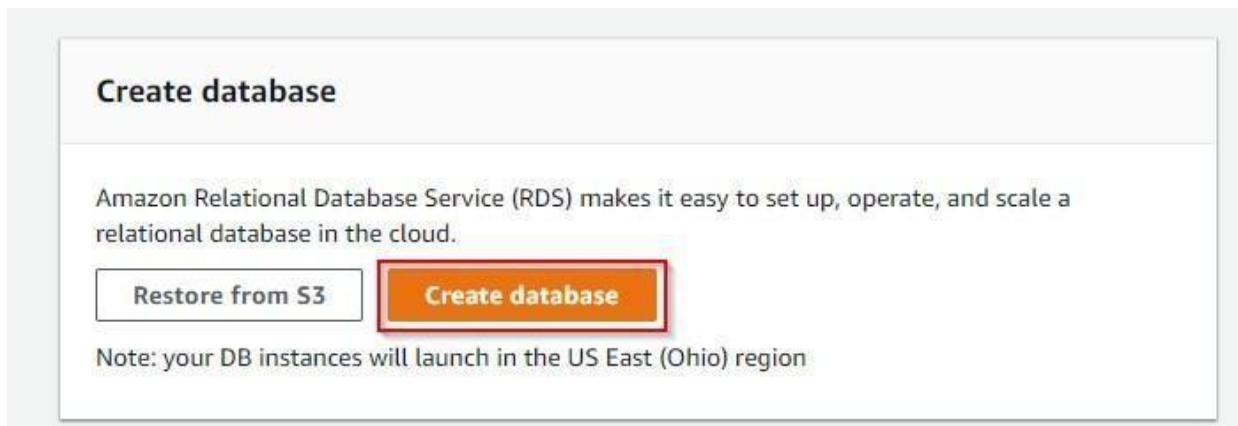
Creating config file /etc/php/5.6/apache2/php.ini with new version
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
apache2_invoke: Enable module php5.6
Setting up php5.6 (5.6.38-3+ubuntu18.04.1+deb.sury.org+1) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
ubuntu@ip-172-31-32-189:~$
```

Now we connect mysql with the RDS:

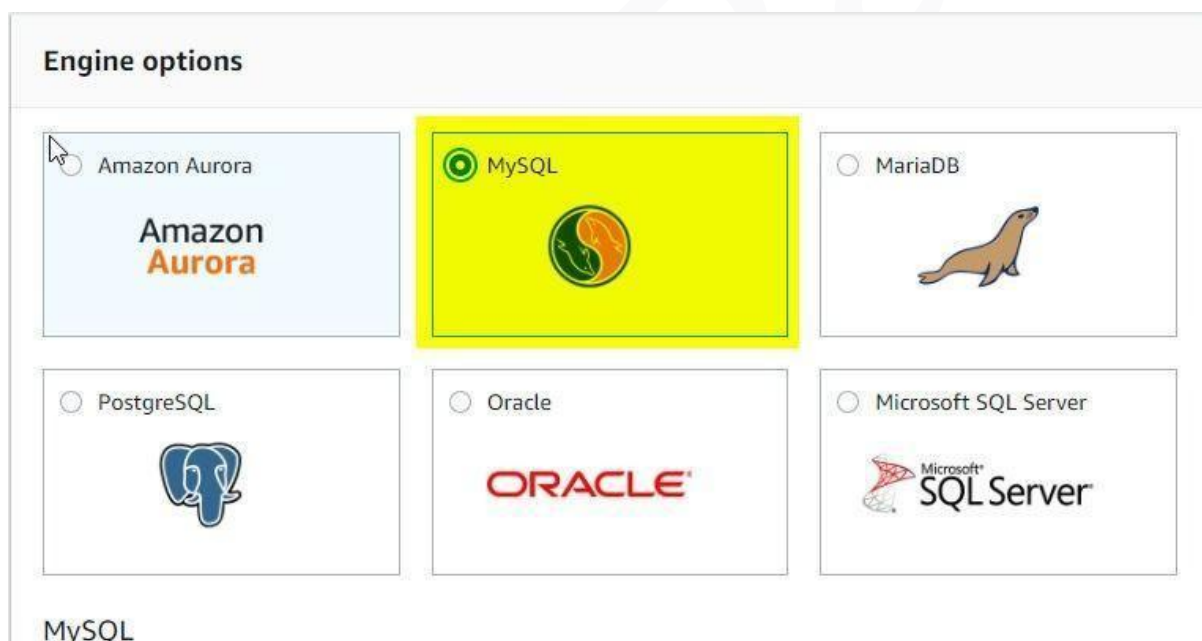
- Go to your AWS Management Console
- Select RDS



- Then click on Create Database



- Select the MySQL Engine and click Next



- Since we're using it for the demo purpose, so we'll choose the Dev/Test -MySQL option only and then click Next

Choose use case

Use case
Do you plan to use this database for production purposes?

Use case

☐ **Production - Amazon Aurora** Recommended
MySQL-compatible, enterprise-class database at 1/10th the cost of commercial databases.

☐ **Production - MySQL**
Use Multi-AZ Deployment and Provisioned IOPS Storage as defaults for high availability and fast, consistent performance.

☒ **Dev/Test - MySQL**
This instance is intended for use outside of production or under the RDS Free Usage Tier.

Billing is based on [RDS pricing](#).

Cancel Previous **Next**

- Specify DB Details, make sure to choose only db.t2.micro in DB Instance Class

DB instance class [Info](#)

db.t2.micro — 1 vCPU, 1 GiB RAM ▼

Multi-AZ deployment [Info](#)

☐ Create replica in different zone
Creates a replica in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.

☒ No

Storage type [Info](#)

General Purpose (SSD) ▼

Allocated storage

- Enter these credentials (Note: Make sure you remember these credentials, as they will be required for connecting the RDS with your PuTTY)

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](#)

- Then in the Configure Advanced Option, make sure to keep the VPC as default, along with the Public Accessibility as Yes

aws Services Resource Groups

RDS > Create database

Step 1 Select engine

Step 2 Choose use case

Step 3 Specify DB details

Step 4 **Configure advanced settings**

Configure advanced settings

Network & Security

Virtual Private Cloud (VPC) [Info](#)
VPC defines the virtual networking environment for this DB instance.

[Refresh](#)

Only VPCs with a corresponding DB subnet group are listed.

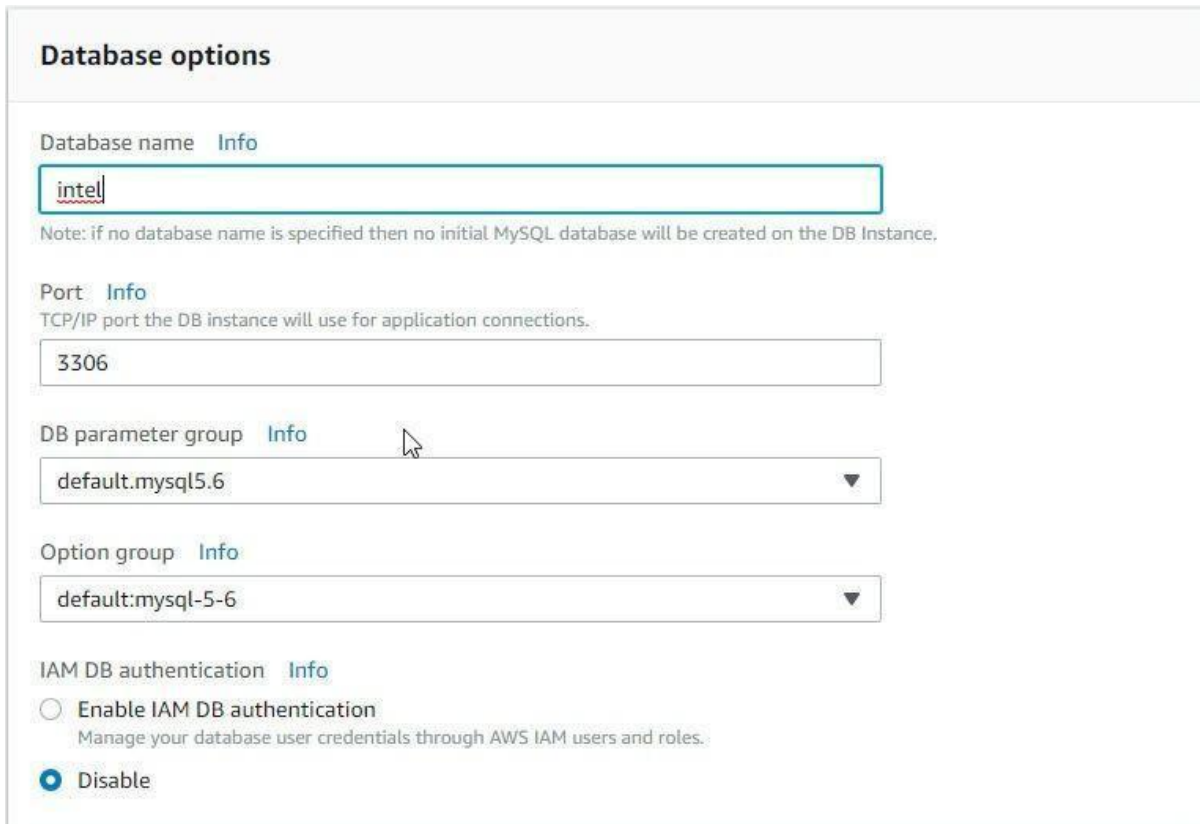
Subnet group [Info](#)
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

Public accessibility [Info](#)

☒ **Yes**
EC2 instances and devices outside of the VPC hosting the DB instance will connect to the DB instances. You must also select one or more VPC security groups that specify which EC2 instances and devices can connect to the DB instance.

☐ **No**
DB instance will not have a public IP address assigned. No EC2 instance or devices outside of the VPC will be able to connect.

- In the Database Options, name the Database and keep the other artifacts as it is



Database options

Database name [Info](#)

Note: If no database name is specified then no initial MySQL database will be created on the DB Instance.

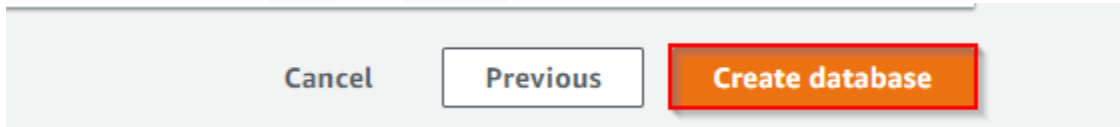
Port [Info](#)
TCP/IP port the DB instance will use for application connections.

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


- Then click on Create database



[Cancel](#) [Previous](#) [Create database](#)

- Then you can check your instance status

RDS > Create database

 **Your DB instance is being created.**
Note: Your instance may take a few minutes to launch.


Connecting to your DB instance

Once Amazon RDS finishes provisioning your DB instance, you can use a SQL client application or utility to connect to the instance.
[Learn about connecting to your DB instance](#)

[All DB instances](#) [View DB instance details](#)

- It may take few minutes for RDS to go from Initial to Running stage, you will observe that Endpoint and Port are not yet available (wait for few minutes)

RDS > Create database

 **Your DB instance is being created.**
Note: Your instance may take a few minutes to launch.

Connecting to your DB instance

Once Amazon RDS finishes provisioning your DB instance, you can use a SQL client application or utility to connect to the instance.
[Learn about connecting to your DB instance](#)

[All DB instances](#) [View DB instance details](#)

- In few minutes, you will be able to see the Endpoint and Port

Connect

Endpoint intel.cqvpjg4mk8sa.us-east-2.rds.amazonaws.com	Port 3306	Publicly accessible Yes
--	--------------	----------------------------

- Also, make sure to change some security configuration in the RDS
- Go to your EC2 Instance Security Groups and select your group ID

Security Group: sg-0e70f9429f934a81e

Description Inbound Outbound Tags

Group name launch-wizard-1

Group ID sg-0e70f9429f934a81e

Group description launch-wizard-1 created 2018-11-16T14:27:39.538+05:30

VPC ID vpc-7f040317

- Then go to RDS Security groups and select the Inbound rules panel there and click on Add Rule

Edit inbound rules

Type	Protocol	Port Range	Source	Description
MySQL/Aurora	TCP	3306	Custom 182.75.139.2/32	e.g. SSH for Admin Desktop

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

- Then paste the EC2 Security ID in Source> Custom > Security Group by keeping the Type as MySQL/Aurora

Edit inbound rules
✕

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
MYSQL/Auror ▾	TCP	3306	Custom ▾ 182.75.139.2/32	e.g. SSH for Admin Desktop ✕
MYSQL/Auror ▾	TCP	3306	Custom ▾ sg-0e70f9429f934a81e	e.g. SSH for Admin Desktop ✕

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel
Save

- Now go back to your PuTTY and use this command as shown below
mysql -h hostname -u username -p

NOTE: In place of hostname, make sure to use your Endpoint from RDS
Username which you created

Here, we're using our own Endpoint and username and password used

```
ubuntu@ip-172-31-32-189:~$ mysql -h intel.cqvpjg4mk8sa.us-east-2.rds.amazonaws.com -u intel -p
```

Use the command as shown below:

- After this, it will ask for your password, in our case, password is: intel123
- Then it will show that you're connected to the mysql

```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 19
Server version: 5.6.41-log Source distribution

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

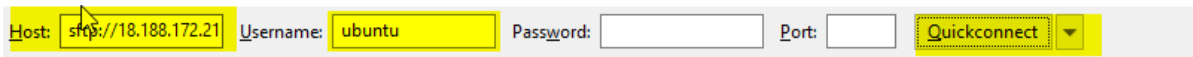
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>
```

Filezilla

- Now install Filezilla
- In order to connect it, enter hostname as the Endpoint of EC2 and Username as Ubuntu and no need to keep the password, then quickconnect



The image shows the Filezilla Quickconnect interface. The 'Host' field contains 'sftp://18.188.172.21', the 'Username' field contains 'ubuntu', and the 'Quickconnect' button is highlighted in yellow. The 'Password' and 'Port' fields are empty.

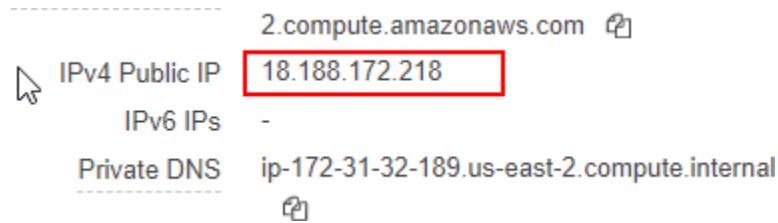
- Now your Filezilla is connected with your EC2 instance
- Create a 'New Folder' of your website in your Desktop
- And copy paste it in your Filezilla Remote Site path: /home/ubuntu

```
ubuntu@ip-172-31-32-189:~$ sudo cp -r New\ folder/ /var/www/html
ubuntu@ip-172-31-32-189:~$ cd /var/www/html
ubuntu@ip-172-31-32-189:/var/www/html$ ls
'New folder'  index.html
```

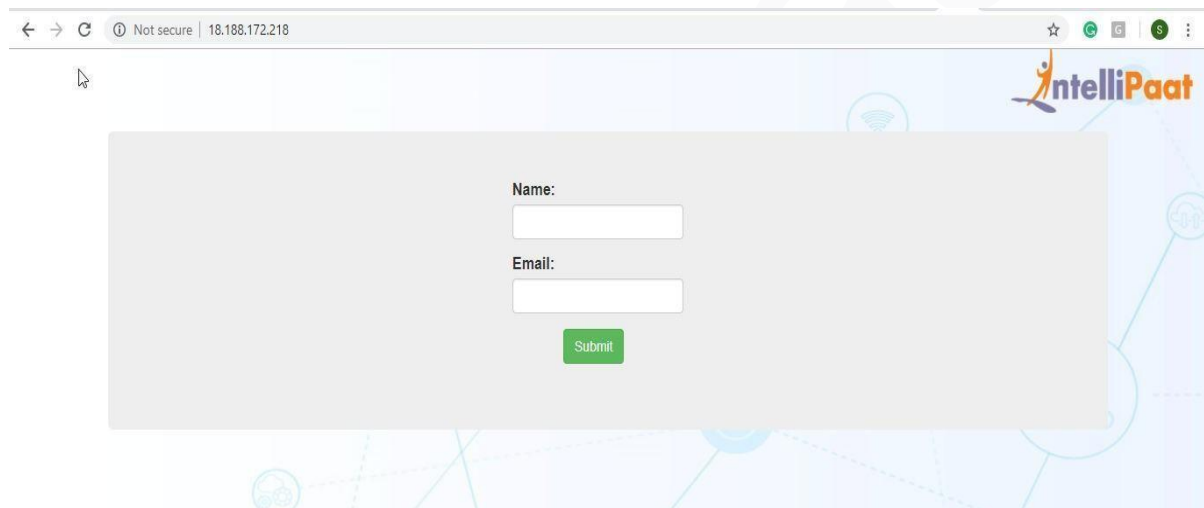
- Now go back to your PuTTY, where you will see that it contains the index.html file
- Now you need to remove this 'index.html' file and add 'index.php' in its place
- For that you need to use "sudo su" and remove this file using remove command

```
ubuntu@ip-172-31-32-189:/var/www/html$ sudo su
[root@ip-172-31-32-189:/var/www/html# rm index.html
root@ip-172-31-32-189:/var/www/html# cd New\ folder/
root@ip-172-31-32-189:/var/www/html/New folder# ls
images  index.php
```

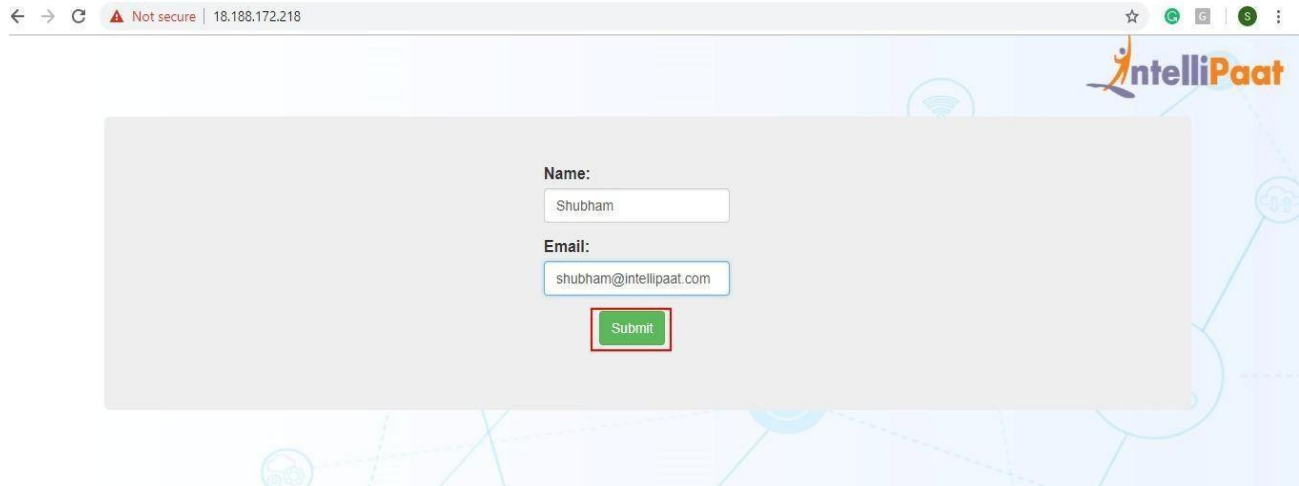

- Now when you will try, and copy paste the Public IP of your EC2 Instance



- After copying this IP to your browser, you will observe that your website is working on it



- Now when you enter these details in this website, you will see the following result



← → ↻ ⚠ Not secure | 18.188.172.218

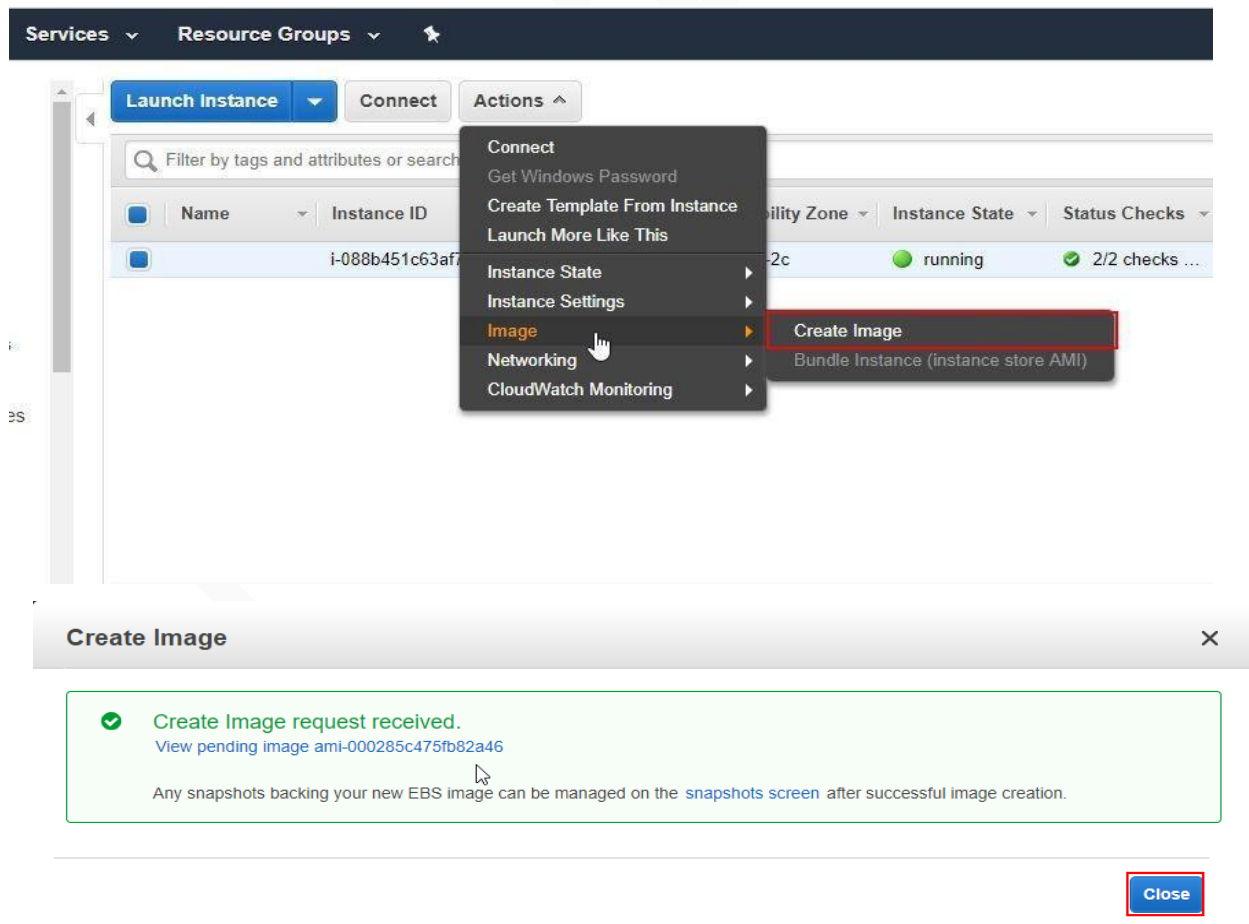
IntelliPaat

Name:

Email:

Auto Scaling:

Now, we'll do the autoscaling of our website by going to our EC2 Instance and then click on Actions and Create Image



Services ▾ Resource Groups ▾ ⚡

Launch Instance ▾ Connect Actions ▴

Filter by tags and attributes or search

Name	Instance ID	Availability Zone	Instance State	Status Checks
	i-088b451c63af...	us-east-2c	running	2/2 checks ...

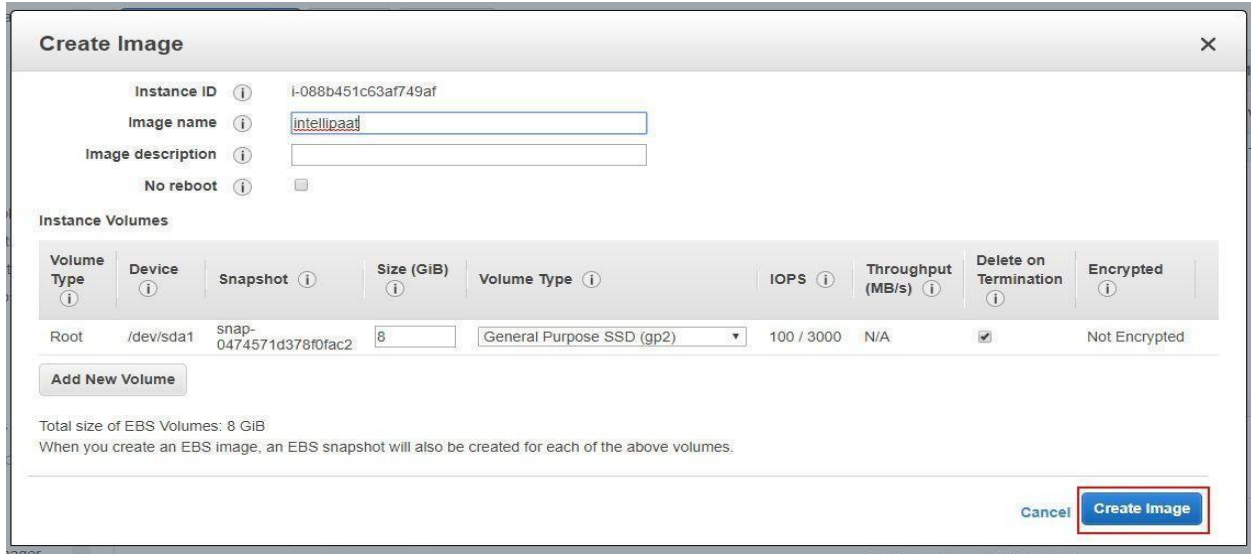
Connect
Get Windows Password
Create Template From Instance
Launch More Like This
Instance State
Instance Settings
Image
Networking
CloudWatch Monitoring

Create Image
Bundle Instance (instance store AMI)

Create Image

✓ Create Image request received.
[View pending image ami-000285c475fb82a46](#)
Any snapshots backing your new EBS image can be managed on the [snapshots screen](#) after successful image creation.

Close



Create Image

Instance ID ⓘ i-088b451c63af749af

Image name ⓘ

Image description ⓘ

No reboot ⓘ ☒

Instance Volumes

Volume Type ⓘ	Device ⓘ	Snapshot ⓘ	Size (GiB) ⓘ	Volume Type ⓘ	IOPS ⓘ	Throughput (MB/s) ⓘ	Delete on Termination ⓘ	Encrypted ⓘ
Root	/dev/sda1	snap-0474571d378f0fac2	<input type="text" value="8"/>	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Total size of EBS Volumes: 8 GiB
When you create an EBS image, an EBS snapshot will also be created for each of the above volumes.

Then further, activate its autoscaling and then its classic load balancer which directs the traffic to your website directly.